UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Development of a data acquisition architecture with distributed synchronization for a Positron Emission Tomography system with integrated front-end

Ph.D. dissertation

December 2015

Author:
Ramón José Aliaga Varea

Supervisors:
Dr. Rafael Gadea Gironés
Dr. Ricardo José Colom Palero

Abstract

Positron Emission Tomography (PET) is a non-invasive nuclear medical imaging modality that makes it possible to observe the distribution of metabolic substances within a patient's body after marking them with radioactive isotopes and arranging an annular scanner around him in order to detect their decays. The main applications of this technique are the detection and tracing of tumors in cancer patients and metabolic studies with small animals.

The Electronic Design for Nuclear Applications (EDNA) research group within the Instituto de Instrumentación para Imagen Molecular (I3M) has been involved in the study of high performance PET systems and maintains a small experimental setup with two detector modules. This thesis is framed within the necessity of developing a new data acquisition system (DAQ) for the aforementioned setup that corrects the drawbacks of the existing one. The main objective is to define a DAQ architecture that is completely scalable, modular, and guarantees the mobility and the possibility of reusing its components, so that it admits any extension of modification of the setup and it is possible to export it directly to the configurations used by other groups or experiments. At the same time, this architecture should be compatible with the best possible resolutions attainable at the present instead of imposing artificial limits on system performance. In particular, the new DAQ system should outperform the previous one.

As a first step, a general study of DAQ architectures is carried out in the context of experimental setups for PET and other high energy physics applications. On one hand, the conclusion is reached that the desired specifications require early digitization of detector signals, exclusively digital communication between modules, and the absence of a centralized trigger. On the other hand, the necessity of a very precise distributed synchronization scheme between modules becomes apparent, with errors in the order of 100 ps, and operating directly over the data links. A study of the existing methods reveals their severe limitations in terms of achievable precision. A theoretical analysis of the situation is carried out with the goal of overcoming them, and a new synchronization algorithm is proposed that is able to reach the desired resolution while getting rid of the restrictions on clock

alignment that are imposed by virtually all usual schemes. Since the measurement of clock phase difference plays a crucial role in the proposed algorithm, extensions to the existing methods are defined and analyzed that improve them significantly. The proposed scheme for synchronism is validated using commercial evaluation boards.

Taking the proposed synchronization method as a starting point, a DAQ architecture for PET is defined that is composed of two types of module (acquisition and concentration) whose replication makes it possible to arrange a hierarchic system of arbitrary size, and circuit boards are designed and commissioned that implement a realization of the architecture for the particular case of two detectors. This DAQ is finally installed at the experimental setup, where their synchronization properties and resolution as a PET system are characterized and its performance is verified to have improved with respect to the previous system.

Resumen

La Tomografía por Emisión de Positrones (PET) es una modalidad de imagen médica nuclear no invasiva que permite observar la distribución de sustancias metabólicas en el interior del cuerpo de un paciente tras marcarlas con isótopos radioactivos y disponer después un escáner anular a su alrededor para detectar su desintegración. Las principales aplicaciones de esta técnica son la detección y seguimiento de tumores en pacientes con cáncer y los estudios metabólicos en animales pequeños.

El grupo de investigación Electronic Design for Nuclear Applications (EDNA) del Instituto de Instrumentación para Imagen Molecular (I3M) ha estado involucrado en el estudio de sistemas PET de alto rendimiento y mantiene un pequeño *setup* experimental con dos módulos detectores. La presente tesis se enmarca dentro de la necesidad de desarrollar un nuevo sistema de adquisición de datos (DAQ) para dicho *setup* que corrija los inconvenientes del ya existente. En particular, el objetivo es definir una arquitectura de DAQ que sea totalmente escalable, modular, y que asegure la movilidad y la posibilidad de reutilización de sus componentes, de manera que admita cualquier ampliación o alteración del *setup* y pueda exportarse directamente a los de otros grupos o experimentos. Al mismo tiempo, se desea que dicha arquitectura no limite artificialmente el rendimiento del sistema sino que sea compatible con las mejores resoluciones disponibles en la actualidad, y en particular que sus prestaciones superen a las del DAQ instalado previamente.

En primer lugar, se lleva a cabo un estudio general de las arquitecturas de DAQ para *setups* experimentales para PET y otras aplicaciones de física de altas energías. Por un lado, se determina que las características deseadas implican la digitalización temprana de las señales del detector, la comunicación exclusivamente digital entre módulos, y la ausencia de *trigger* centralizado. Por otro lado, se hace patente la necesidad de un esquema de sincronización distribuida muy preciso entre módulos, con errores del orden de 100 ps, que opere directamente sobre los enlaces de datos. Un estudio de los métodos ya existentes revela sus graves limitaciones a la hora de alcanzar esas precisiones. Con el fin de paliarlos, se lleva a cabo un análisis teórico de la situación y se propone un nuevo algoritmo de sincronización que es capaz de alcanzar la resolución deseada y elimina las restricciones de alineamiento de reloj impuestas por casi todos los esquemas usuales. Dado que la medida de desfase entre relojes juega un papel crucial en el algoritmo propuesto, se definen y analizan extensiones a los métodos ya existentes que suponen una mejora sustancial. El esquema de sincronismo propuesto se valida utilizando placas de evaluación comerciales.

Partiendo del método de sincronismo propuesto, se define una arquitectura de DAQ para PET compuesta de dos tipos de módulos (adquisición y concentración) cuya replicación permite construir un sistema jerárquico de tamaño arbitrario, y se diseñan e implementan placas de circuito basadas en dicha arquitectura para el caso particular de dos detectores. El DAQ así construído se instala finalmente en el *setup* experimental, donde se caracterizan tanto sus propiedades de sincronización como su resolución como sistema PET y se comprueba que sus prestaciones son superiores a las del sistema previo.

Resum

La Tomografia per Emissió de Positrons (PET) és una modalitat d'imatge mèdica nuclear no invasiva que permet observar la distribució de substàncies metabòliques a l'interior del cos d'un pacient després d'haver-les marcat amb isòtops radioactius disposant un escàner anular al seu voltant per a detectar la seua desintegració. Aquesta tècnica troba les seues principals aplicacions a la detecció i seguiment de tumors a pacients amb càncer i als estudis metabòlics en animals petits.

El grup d'investigació Electronic Design for Nuclear Applications (EDNA) de l'Instituto de Instrumentación para Imagen Molecular (I3M) ha estat involucrat en l'estudi de sistemes PET d'alt rendiment i manté un petit *setup* experimental amb dos mòduls detectors. Aquesta tesi neix de la necessitat de desenvolupar un nou sistema d'adquisició de dades (DAQ) per al *setup* esmentat que corregisca els inconvenients de l'anterior. En particular, l'objectiu és definir una arquitectura de DAQ que sigui totalment escalable, modular, i que asseguri la mobilitat i la possibilitat de reutilització dels seus components, de tal manera que admeta qualsevol ampliació o alteració del *setup* i pugui exportar-se directament a aquells d'altres grups o experiments. Al mateix temps, es desitja que aquesta arquitectura no introduisca límits artificials al rendiment del sistema sinó que sigui compatible amb les millors resolucions disponibles a l'actualitat, i en particular que les seues prestacions siguin superiors a les del DAQ instal·lat amb anterioritat.

En primer lloc, es porta a terme un estudi general de les arquitectures de DAQ per a *setups* experimentals per a PET i altres aplicacions de física d'altes energies. Per una banda, s'arriba a la conclusió que les característiques desitjades impliquen la digitalització dels senyals del detector el més aviat possible, la comunicació exclusivament digital entre mòduls, i l'absència de *trigger* centralitzat. D'altra banda, es fa palesa la necessitat d'un mecanisme de sincronització distribuïda molt precís entre mòduls, amb errors de l'ordre de 100 ps, que treballi directament sobre els enllaços de dades. Un estudi dels mètodes ja existents revela les seues greus limitacions a l'hora d'assolir aquest nivell de precisió. Amb l'objectiu de pal·liar-les, es duu a terme una anàlisi teòrica de la situació i es proposa un nou algoritme de sincronització que és capaç d'obtindre la resolució desitjada i es desfà de les restriccions d'alineament de rellotges imposades per gairebé tots els esquemes usuals. Atès que la mesura del desfasament entre rellotges juga un paper cabdal a l'algoritme proposat, es defineixen i analitzen extensions als mètodes ja existents que suposen una millora substancial. L'esquema de sincronisme proposat es valida mitjançant plaques d'avaluació comercials.

Prenent el mètode proposat com a punt de partida, es defineix una arquitectura de DAQ per a PET composta de dos tipus de mòduls (d'adquisició i de concentració) tals que la replicació d'aquests elements permet construir un sistema jeràrquic de mida arbitrària, i es dissenyen i implementen plaques de circuit basades en aquesta arquitectura per al cas particular de dos detectors. L'electrònica desenvolupada s'instal·la finalment al *setup* experimental, on es caracteritzen tant les seues propietats de sincronització com la seua resolució com a sistema PET i es comprova que les seues prestacions són superiors a les del sistema previ.

Contents

Chapter 1

Introduction

Positron Emission Tomography (PET) is a non-invasive nuclear medical imaging modality that provides maps of the distribution of particular substances within a patient's body by marking them with certain radioactive isotopes and subsequently detecting their decay using rings of detectors located around it. It is particularly well suited for the detection and tracing of tumors in cancer patients and for metabolic studies with small animals. PET shares many of its working principles, required instrumentation, and measurement and processing techniques with those employed in nuclear and particle physics experiments.

The Electronic Design for Nuclear Applications (EDNA) research group within the Instituto de Instrumentación para Imagen Molecular (I3M), located at the Universidad Politécnica de Valencia (UPV), was involved in projects for the development of high performance PET systems from 2004 to 2013. In particular, its tasks included research on detector readout and data processing methods, the development of data acquisition (DAQ) systems, and the maintenance of a small experimental setup with two detector modules and a radioactive test source. This thesis is framed within EDNA's research activities at the time and involves the inception, design and development of a scalable DAQ system for PET scanners that is compatible with the available setup and synergizes with the rest of the group's developments.

Motivation

The title of the thesis specifically mentions the synchronization aspect of the PET system. In order to see how this becomes a concern, it is probably best to describe the initial motivation behind the line of work followed in this thesis.

The original problem is set within the development of the previous-generation DAQ system in the group, which was commissioned by a partner company for the readout and processing of a 16-detector scanner [1]. Very late in the development process, after the system had already been planned and designed and the circuit boards had been fabricated, the partner company decided to change specifications to those of an 18-detector scanner. Unfortunately, the system was not able to be redesigned to allow for an expansion in its number of detectors at this stage so the new specifications could not be fulfilled; instead, it was adapted for its internal use in the group's experimental PET setup for research purposes.

The starting point of this thesis arises in this context, as an attempt to come up with a DAQ architecture and implementation that is configurable enough to avoid these kinds of organizational drawbacks by supporting variable, expandable detector topologies. The obvious problem in this case is that the system is not scalable; however, stating it in these terms is just a matter of naming and does not provide a real insight into the limitations behind it. One should instead look for the root problem that makes the 16-detector DAQ system design unable to handle 18 detectors. After all, there is the immediate question: if the 16-detector DAQ has already been designed and built, why not just use two full DAQs to handle 16 detectors each and merge the results from each one externally in order to obtain a 32-detector DAQ? This would allow the reuse of the system that had already been designed. Unfortunately, this turns out to be impossible.

A detailed analysis of the situation reveals that the root of the problem lies in the synchronization between DAQ components. To be more precise, a DAQ system for PET with reasonable specifications (particularly time resolution), requires the set of all its sensing electronics to be synchronized with an accuracy well below 1 ns. The only external connection of the 16-detector DAQ consisted in a data link that was more than capable of handling the data rate required by the application but it was not designed in a way that could provide this kind of synchronization accuracy between its internal components and an external reference. Typically, when a data link is used for synchronization, the best resolution that it can provide is in the order of the period of the local logic clock frequency, i.e. around 5 ns to 10 ns.

It should be mentioned that this situation is hardly exclusive of the PET electronics designed at I3M, and similar examples are provided by commercial modular evaluation kits with data acquisition for photodetectors, such as SensL's Matrix system [2] and later Philips' Technology Evaluation Kits (TEK) for digital silicon photomultipliers [3, 4]. These systems consist of sensor array modules with a central communications board to which all modules are connected, that has a digital data output link and supports a maximum number of sensor modules (16 and 4, respectively). It is not possible to extend the system size by using several communication boards even if all events are timestamped because the outgoing data link does not provide precise synchronization capability between them. Instead, the only possibility is system redesign with support for more modules (e.g. the Philips TEK was eventually extended to a Module TEK supporting 8 modules), but this is just an increase of the hard limit and not real scalability.

To recap, this thesis starts with the observation that external links with precise synchronization capability are mandatory for the development of truly scalable, modular data acquisition systems for PET. A simple bibliographic search reveals that these considerations arise also in the more general context of experimental physics applications and particularly in large scale installations for high energy physics research, although they do for different reasons such as the operational difficulties introduced by calibration processes and the effect of variations in environmental conditions. It therefore makes sense to study these and other problems, existing or potential, that are present in both situations and try to come up with a common solution. Hence, while synchronization is the main focus of the thesis, the topics of programmability, versatility and simplified calibration will also be emphasized.

Hypothesis and objectives

The starting hypothesis of this dissertation is that it is possible to define a DAQ system architecture for PET that is arbitrarily scalable, configurable, mobile and consisting of completely reusable electronic modules, yet still compatible with the highest possible performance offered by available detectors i.e. such that it does not hinder accuracy by introducing an artificial resolution limit. This hypothesis, together with the goal of upgrading the existing setup at the EDNA facilities, makes it possible to break the thesis work down into the following main objectives:

- To carry out a study of DAQ architectures, including PET systems and more general experimental physics setups, in order to determine the key features that are needed to fulfill the stated conditions. In other words, to translate the requirements stated in the starting hypothesis into more specific technical terms.

- To propose a specific PET DAQ architecture that satisfies these conditions, i.e. to define its building blocks and functional requirements.

- To develop and implement a functional demonstrator of the proposed architecture that is compatible with and able to replace the existing DAQ system already in place at the experimental setup.
- To validate and characterize the new electronics.

As was already pointed out above, it becomes eventually clear during the course of this thesis that synchronization between all electronic modules within the system is one of the key issues behind the scalability of DAQs with precise timing capability. Hence, the following list of objectives must be added to the previous one:

- To study the existing methods for scalable, precise synchronization in DAQ systems and determine their limitations.
- To propose a synchronization scheme that is compatible with the architectural and numerical requirements stated above.
- To implement and validate the proposed synchronization method.

Outline of the dissertation

The analytic process followed during the inception of the thesis has already been described, from the detection of a problem to the exposure of its root causes. In contrast, this dissertation is structured in a synthetic way, starting with basic definitions and facts and progressively building up on them until the problem and its proposed solution can be described rigorously. In accordance with this approach, the text is divided in two parts. The first part comprises chapters §2 and §3 and is dedicated to data acquisition systems and synchronization in generic high energy physics settings, whereas the second part spans through chapters §4, §5 and §6 and particularizes the general discussion in the early chapters to the special case of PET systems.

Chapter §2 starts by describing the structure of experimental setups and DAQ systems. Their past and current trends are described, and two main conclusions are extracted: the optimality of triggerless systems with early digitization of sensor signals for reduced signal distortion, and the existence of strong restrictions on module synchronization in order to satisfy increasingly stringent time resolution requirements. These conditions imply the need to implement schemes for precise front-end synchronization over cables or fibers. Traditional realizations of such schemes have several drawbacks, mainly related to calibration procedures and sensitivity to temperature variations.

Connection to modules through self-calibrated, self-synchronizing data links is proposed as a better alternative and analyzed extensively in chapter §3. The fundamental reasons behind the usual accuracy limits are exposed, together with the conditions that make it possible to overcome them. A review of the available technology reveals that most implementations of precise synchronization force systemwide alignment of clocks in all sampling nodes. It is argued here that this is not necessary, and it is in fact more efficient to allow for phase differences by adopting the concept of fractional timestamps for synchronization algorithms. A proof-of-concept implementation of the proposed synchronization method is described and early experimental results are reported that prove its validity.

The second part of this thesis begins with chapter §4, dedicated to a detailed description of the PET technique, from the generation of events inside the patient's body until the computation of the medical image, including all aspects related to the acquisition and estimation of physical data. After that, chapter §5 focuses on the PET DAQ systems that handle all these steps and their synchronization as particular cases of the DAQs for experimental physics studied before, and a specific DAQ architecture is proposed based on the results obtained in the first part.

Finally, chapter §6 describes a partial implementation of the proposed architecture for the specific situation of the experimental PET setup located within the EDNA facilities, consisting of only two detectors. While this chapter only fills a relatively small fraction of the total text length, the work it describes corresponds to the largest part of the total time devoted to this thesis, as it involved the development, fabrication and debugging of complex circuit boards together with all associated firmware and acquisition software. Most details concerning the development process itself are skipped, however; only the final, working result is described instead.

Publications

Different parts of the work described in this thesis were presented at the Real Time Conference in 2010 [5] and the Nuclear Science Symposium in 2012 [6], and subsequently published as peer-reviewed journal papers in IEEE Transactions on Nuclear Science describing the generic synchronization scheme [7] and the DAQ circuit boards [8]; they correspond roughly to the first and second parts of this text, respectively.

In parallel with the development of this thesis, the author has also taken part in the research of many other technological advances within the field of PET instrumentation at EDNA. Most of them are mentioned in §4.3 where appropriate, together with the corresponding paper references.

Acknowledgments

The development of this thesis was supported in part by the Spanish Ministry of Science and Innovation under FPU Grant AP2006-04275 and CICYT Grant FIS2010-21216-C02-02. The author would also like to thank the Altera University Program for their generous donation of two Stratix IV GX devices.

Special thanks go to Paola Ferrario, Rafael Gadea, Gessamí Montalbán, Uwe Pietrzyk, Ana Ros and Eduardo Ros for proofreading the document, and to Vicente Herrero, Christoph Lerche, Carles Montoliu and José María Monzó for providing graphic material.

Chapter 2

Data acquisition for experimental physics

Any system such as PET that aims at obtaining a logical representation of a real-world object or phenomenon needs a way to detect and interact with said phenomenon and generate enough data to describe it adequately. This generic process of measurement and recording of physical quantities is referred to as *data acquisition* and is carried out by a *data acquisition system*, usually abbreviated as DAQ.[1] The DAQ itself consists of electronic subsystems including electrical transducers, signal conditioning circuitry, analog-to-digital converters and digital signal processing firmware or software.

This chapter serves as an introduction to the most important aspects of electronic DAQ systems, not just for PET but rather for more general experimental physics applications. In particular, a generic high energy physics experiment and its associated trigger and data acquisition system will be described, highlighting the specific DAQ requirements for these kinds of setup, as well as their evolution over the last few decades in order to justify the current state of the art. The text will focus especially on the DAQ components closer to the physical detectors and less on data storage and offline processing aspects. The fundamental issue of DAQ synchronization will be separately and thoroughly discussed in chapter §3, as it is central to this dissertation.

It may strike the reader as odd to find a thesis dedicated to PET systems focusing on DAQ for high energy physics. There are two reasons for this approach. On one hand, there is considerable overlap between the specific techniques and instru-

[1] The acronym DAQ is typically used to refer to the electronic acquisition system but not to the general concept of data acquisition; such will be the convention followed in this text.

mentation used by PET scanners and those employed in generic particle physics setups, and studying the more generic setting can provide a better overview of the reasons behind the current state of the art, as well as hint at its possible evolution. On the other hand, the problem of the synchronization of distributed systems, which is at the core of this thesis, is applicable to the generic setting and particularly important for large-scale installations, so it makes little sense to limit the discussion to PET systems. In later chapters, PET imaging and its associated data acquisition systems will be regarded as a particular case of the physics setups described here in which the variable of interest is the spatial distribution of positrons within a given sample.

2.1 High Energy Physics experiments

Particle physics is the branch of physics that studies the states of matter and their interaction at the subatomic scale. Nowadays, it is usually divided into *nuclear physics*, dealing with the structure of atomic nuclei, and *elementary particle physics* (or simply *particle physics*), which studies the fundamental particles themselves, particularly those described by the Standard Model ([9], pp. 1–5).

Typical particle physics experiments consist in bringing accelerated particles or ions to collision with other accelerated particles or with fixed targets and then recording and cataloguing the new, usually short-lived particles created by the resulting reactions [10]; as these collisions have to take place at very high energies in order to generate interesting interactions, the name *high energy physics* (HEP) is commonly used to refer to this branch of physics. The difference between all three terms (particle physics, nuclear physics, HEP) is subtle and goes beyond the scope of this dissertation, so they will be used indistinctly.

2.1.1 Structure of a HEP setup

The goal of any HEP setup is to characterize a specific subset of particle interactions occurring within a well-defined space region. Instances of the interactions under study are called *events*. An event starts with a primary interaction that may spawn secondary particles, and includes all subsequent interactions by any generated particles, which may be referred to as *subevents*. An immediate but very important consequence of this definition is that different events in an experiment can always be assumed to be independent of each other [10].

The intended characterization of events may be complete, as is the case with large-scale experiments looking for new physics where ideally all generated particles, both temporary and final, are identified and their tracks are reconstructed; or it may be partial, usually in smaller-scale setups like a gamma camera, where the necessary data for each interaction is just the linear projection of the primary

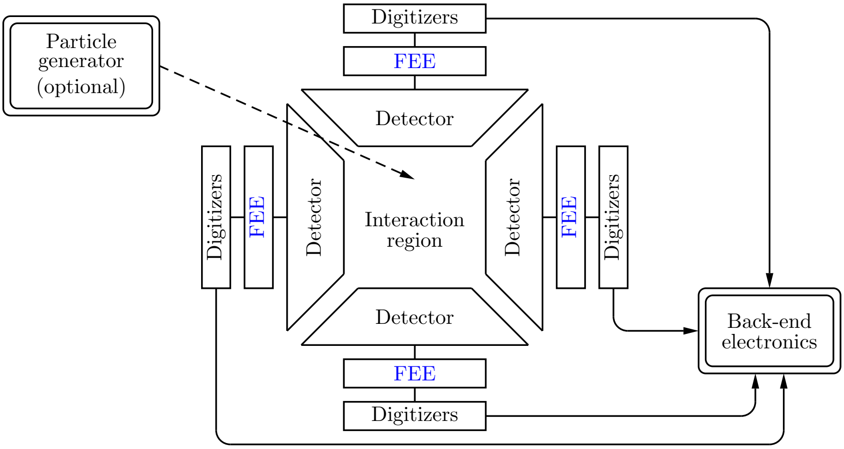


Figure 2.1: Block diagram of a generic HEP setup front-end with a particle accelerator. FEE stands for *front-end electronics*.

interaction location onto a detector plane. Additionally, events under study may take place spontaneously, e.g. at cosmic ray observatories or in setups that measure radioactive decay like PET scanners; or they may need to be forced, especially when the desired interactions need to happen at very high energies. In the latter case, particle accelerators are used in order to create the conditions for the desired interactions.

Figure 2.1 outlines the typical block diagram of an experimental HEP setup. An *interaction region* is defined where the events under study will be detected; beams from particle generators need to be directed to that region if used. The interaction region is surrounded by one or more *particle detectors* with associated readout and digitization electronics. These elements are referred to collectively as the *front-end*.[2] Outputs from the front-end are transmitted to the *back-end* section for further processing and storage, which may be at a long physical distance from the front-end electronics and the detectors themselves.

[2] Terminology is somewhat loose in that some authors consider the term *front-end* to be a functional term and refer to all electronics until the digitization step, regardless of their physical location. In this text, *front-end* will be used as a positional term and denote the electronics close to the detector, regardless of their function.

Particle accelerators

A *particle accelerator* is a device that generates beams of subatomic charged particles moving at speeds comparable to the speed of light in order to cause high energy collisions. The first accelerators were invented in the 1920s by Widerøe and Lawrence [11], but their principle of operation has barely changed since then: particles are accelerated by electric fields while their movement is controlled by magnetic fields ([12], pp. 8–9).

Accelerators may be classified into two basic types according to particle movement. In a linear accelerator, or *linac*, particles are accelerated in a straight line using an electric field. Cyclic accelerators, sometimes called *synchrotrons*,[3] apply an additional magnetic field that induces a Lorentz force on the particles, causing their trajectories to bend into arcs. This allows devices of equivalent size to accelerate the particles at higher energies by having them perform several rotations.

An accelerated particle beam can be brought to collision with either a fixed target or another accelerated beam. In the second case, interactions with higher energy can be achieved. A single circular accelerator may be used to generate two colliding beams of particles of opposite charge. In order to reach the highest energies, accelerator fields have to be modulated to compensate for relativistic effects. This makes it impossible for modern accelerators to generate continuous beams. Instead, particles are supplied in periodic bunches, synchronized with a *bunch frequency* clock ([13], pp. 6–7).

Two key concepts are used for the design and assessment of HEP experiments and beam generators [14]. The *luminosity* $\mathcal{L}$ of an experimental setup is a measure of its ability to generate the required number of interactions, and is defined as the number of incident particles per unit area and unit time produced by the accelerator that get to interact with the target ([15], pp. 66–67). The *cross section* σ of a particular particle interaction is a measure of the probability that said interaction will occur between an incident particle beam and its target per unit area, and is expressed in cm^2 ([15], pp. 20–22). Cross sections are additive, in the sense that partial cross sections representing the probabilities of the different possible reactions can be added to obtain the total cross section representing the probability that there is any interaction at all. Together, these values yield the expected rate of events per second

$$\frac{dN}{dt} = \mathcal{L}\,\sigma \tag{2.1}$$

which can be integrated to obtain the expected total number of events in a given time interval [16]. The value of σ is a physical constant that depends on the particles and reaction type only, while $\mathcal{L}$ is a parameter of the particle accelerator setup.

[3] Incorrectly, as a synchrotron is just a particular type of cyclic accelerator.

Detector type	Interaction	Detector examples
Tracking	Ionization	Gaseous detectors: Multiwire Chamber Drift Chamber Time Projection Chamber Hodoscopes
	Pair production	Semiconductor detectors
Calorimetry	Bremsstrahlung	EM calorimeters Hadronic calorimeters
	Scintillation	Scintillation counters
Particle ID	Čerenkov radiation	Ring Imaging Čerenkov counters
	Transition radiation	TR detectors

Table 2.1: Common particle detection methods in HEP and corresponding detectors. Data collected from [18]

For a given collision, several interaction outcomes are usually possible with different cross sections, but only some of them are actually of interest. Increasing the accelerator luminosity results in a higher rate of interesting events, but it also increases the rate of *background events*, i.e. uninteresting ones; the "signal-to-noise ratio" is given by the ratio of cross sections and therefore an invariant.

Particle detectors

A reaction caused by particle collision may generate a large number of new subatomic particles that need to be identified to obtain a complete description of the event ([17], p. xxi). This is accomplished by deploying several particle detectors around the interaction region. Various detector elements are employed to interact with different types of particles and measure different physical properties such as position, momentum, energy or charge. A complex detector system can thus consist of separate subdetectors whose information is later aggregated in order to completely identify particles and their trajectories.

Table 2.1 outlines the most common types of particle detector and the underlying physical phenomena. Detectors are typically classified according to their specific function within the system as follows ([12], pp. 9–10):

- *Track detectors:* These devices contain materials that react with electrically charged particles that traverse them and large arrays of wires or stripes that detect and collect the nearby results of said reaction. The signals produced by these detectors thus codify particle position, and their evolution over time yields the particle trajectories, or *tracks*. If a magnetic field is applied, the trajectory is forced to bend and particle momentum can be computed from its radius of curvature, as well as the sign of its charge. Particle velocity may also be obtained in some cases by measuring the time of flight between different points along the track.

The most common types of track detectors include gaseous detectors like proportional chambers or drift chambers ([9], pp. 205–258), where the gas gets ionized as charged particles traverse it and emits electrons which are then collected by electrodes, and semiconductor detectors where electron-hole pairs are produced. Usually, these devices feature a high number of readout channels, only a few of which are active at any given time. The most important characteristic of track detectors is their spatial resolution; current detectors can determine particle position with an accuracy better than 100 µm [19].

- *Calorimeters:* These detectors feature massive blocks of material designed to absorb particles, forcing them to deposit all of their energy in their interactions and generating showers of secondary particles which are then detected. Particle position and total deposited energy may be measured. There are two basic types of calorimeter: electromagnetic, where electrons and photons are detected as they interact with charged particles in the material, and hadronic, where hadrons are absorbed by interaction with atomic nuclei. This is the only kind of device that is able to detect chargeless particles ([9], pp. 259–284).

  Large calorimeters usually consist of alternate layers of absorber and sensitive material; the first type stops the particles while the second one collects the deposited energy. This kind of detector provides a spatial sampling of the energy loss induced by the particle shower. As a more simple example, crystal scintillators are a type of calorimeter (sometimes called *crystal calorimeters*) designed to absorb incoming photons and ultimately generate a scintillation, i.e. a shower of optical photons that can be detected with a photodetector and whose combined energy equals that of the original impinging photon. Scintillators are the preferred detector in the vast majority of PET scanners and will be described in greater detail in §4.2.2.

- *Particle identifiers:* A third type of detector is used to measure a particle's velocity by detecting the radiation it emits under certain conditions; combining this information with the momentum obtained in a track detector yields the mass and allows full identification of the particle. The two main physical phenomena underlying the operation of these devices are Čerenkov radiation, emitted by particles moving through a medium at a speed higher than the speed of light in said medium ([9], pp. 178–183), and transition radiation, which is generated whenever particles cross the boundary between mediums with different dielectric constants ([9], pp- 294–296).

In a fully fleshed out HEP experiment setup, the complex detector is layered in such a way that the position of individual detectors matches the expected travel range of particles before they completely lose energy [18]. High-resolution track detectors are placed closest to the interaction region and trace all charged particles. They are surrounded by particle identifiers and calorimeters, electromagnetic on the innermost side and hadronic on the outermost side. Finally, very large, dense calorimetric blocks cover the previous layers in order to detect heavy charged particles like muons, which interact very weakly with matter and are likely to escape the inner layers ([9], p. 14). The only particle escaping this kind of setup is the virtually undetectable neutrino, but its presence can be inferred from the law of conservation of energy if all other particles are detected and the detector is hermetic ([12], p. 10).

Detector readout

Practically all detectors currently in use have an electronic readout [18]. For most of them, the result of a particle interaction is the appearance of an electric charge Q either in the active volume of a direct detector (e.g. in gas chambers) or indirectly in a secondary transducer (e.g. in PMTs coupled to scintillation crystals). This charge is then transported through electric fields and collected by electrodes, and an electrical current signal $i\,(t)$ is induced whose integral over the pulse duration T equals Q ([20], pp. 103–104).

An extremely common misconception is that the detected pulse does not start until the first electric charges are deposited on the electrodes. However, the current signal is actually induced by the motion of the charges through the detector volume, not by their collection, and thus starts as soon as the charge is generated; this is a consequence of the Shockley-Ramo theorem [21]. Hence, the information about relative timing of different interactions within a given event is preserved in the detected pulses as there is no delay between the particle interaction and the generated current in a direct detector.[4]

The basic electrical model for the detector, from the point of view of the readout electronics, is a current source or sink with an equivalent output capacitance C ([22], pp. 125–132).[5] Additionally, many detectors require their electrodes to be biased at relatively high voltage values, and thus a DC blocking capacitor may need to be added in series at the output ([24], pp. 12–13). The equivalent circuit is shown in Fig. 2.2, together with the first front-end electronics stages.

[4] This does not apply to indirect detectors, of course, but is valid for the last detector in the chain.

[5] Detector models for very accurate simulations may require more sophisticated equivalent circuits; see [23] for examples.

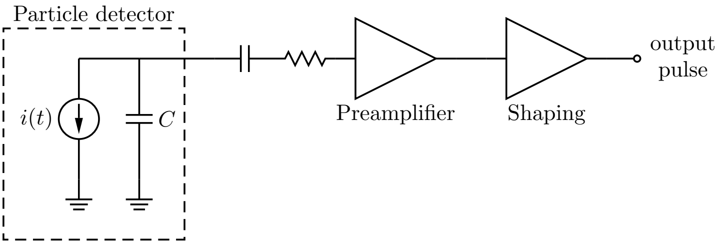


Figure 2.2: Particle detector electrical model and typical front-end circuitry.

Usually, the only interesting data to be extracted from the pulses are the magnitude Q and the timing information.[6] The signals generated by detectors are typically very weak and often masked by high levels of noise, and so they cannot be used directly, but need to be amplified instead. It is therefore desirable to preamplify and process them as soon as possible, hopefully before they traverse transmission lines that further degrade signal quality ([20], pp. 578–583) and timing [27].

Two different operation modes can be distinguished, depending on the relationship between the time constant τ of the preamplifier circuit and the original pulse duration. If $\tau \ll T$, then the output pulse will be fast and have the same shape as $i\,(t)$, thus preserving timing information. On the other hand, if $\tau \gg T$, the preamplifier will essentially integrate the input pulse current until a maximum value is reached at time T and then slowly discharge with time constant τ; magnitude information is thus preserved, as the output amplitude is directly proportional to Q ([20], pp. 107–110). In this case, the output signal presents a long tail given by time constant τ, which may cause the output pulses from separate events to overlap, resulting in corrupted amplitude values. This phenomenon is referred to as *event pile-up* and can be mitigated by employing an additional *pulse shaping* circuit that cancels the tail and reduces pulse duration ([12], pp. 10–12).

Digitization

Modern data acquisition systems usually require the conversion of all important detector-generated data to the digital domain at some point, so that further processing may be performed on them using digital circuitry. Digitization is typically carried out by one of two types of circuits: analog-to-digital and time-to-digital converters (ADCs and TDCs, respectively). An ADC provides a digital representation of the value attained by a certain voltage signal,[7] whereas a TDC is

[6] In detectors where multiple interaction types are possible, e.g. with different particles, additional information may be extracted from the shape of the rising edge by virtue of the Shockley-Ramo theorem. Pulse shape analysis may then be performed to distinguish between them [25, 26].

[7] This definition includes comparators as particular cases of ADCs with continuous 1-bit outputs.

employed to obtain the duration of a signal or the time difference between two signal pulses.

ADC values are typically not output continuously but rather sampled at given time instants, and ADCs may be further classified into two operating modes depending on the way that the conversion instants are specified. Some ADCs, particularly slower ones, only perform conversion on demand as requested by an external control signal. Faster ADCs, on the other hand, may function in *free-running sampling* (FRS) mode and provide a steady stream of waveform samples that are captured at a regular pace given by a sampling clock [28]. The main difference between both modes is in the volume of generated data, which is much higher in the second case as it includes information corresponding to detector outputs at time intervals when no event is active.

Digitizers based on on-demand conversion need a control signal that codifies the decision about what to convert and when. In the most simple case, a local fast trigger signal may be generated whenever a detector signal crosses a certain threshold value by way of a comparator, in order to determine whether said signal contains a pulse or is pure noise. The generated local trigger can be used to drive conversion circuitry, e.g. to specify the optimal sampling instant in integrating front-ends.

In many experiments, this fast trigger is modified by external control signals. For example, composite triggers that depend on time coincidence between subdetectors can be obtained by performing a logical AND operation on their individual triggers. Alternatively, a global *trigger veto* signal may be distributed to all subdetectors that inhibits digitization system-wide when it is known that new events cannot be accepted. There are various possible reasons for this, such as synchronization with the bunch frequency when interactions are only allowed at periodic beam crossings or spill times in accelerator-based setups, recovery time of subdetectors since the last event, or overload in the DAQ system. In general, the trigger is a complex DAQ subsystem with the goal of filtering invalid events and thus reducing the output data rate; it will be described in the next section.

Digitization of detector signals may take place either in the front-end or in the back-end sections of the data acquisition system. Both approaches have their advantages and disadvantages:

- In the former case, the conversion electronics are subject to large radiation doses in most experimental setups involving high energies and must therefore be based on radiation-hardened devices; moreover, in setups where all digitization electronics must be synchronized, the synchronization has to be realized using the links between front-end and back-end.
- In the latter case, conversion electronics are located in the *counting room*, i.e. the site close to the actual detector but protected against radiation

where the back-end components are located. Radiation hardening is not necessary in that case and synchronization is easier because electronics may be assembled in racks. However, analog signals from the detector have to be transmitted to the back-end using long cables which typically results in severe signal distortion, attenuation and loss of timing information.

2.1.2 DAQ and trigger systems

In the last few pages, the typical structure of a HEP setup has been presented along with the associated readout electronics up until the digitization step. Processing in these early stages is scarce, analog for the most part, and mainly limited to the adaptation of certain signal parameters and simple discrimination [18]. In this section, the later stages of the data acquisition system that deal with digital data are considered.[8]

Overview of the DAQ

Traditionally, the primary mission of the DAQ system is to gather event-related data from all detectors, aggregate them into data formats that fully describe events, and permanently store them for later analysis ([12], pp. 12–13). Nowadays, DAQ systems also assume the additional responsibility of selecting candidate events of interest to be stored and rejecting the rest, thus reducing the output data rate [18]. This is an increasingly important function as the rate of invalid events scales with higher luminosities and lower cross sections.

A DAQ system performs a number of functions that may be split into three categories representing separate functionalities with near-independent datapaths and entirely different timing requirements [29]:

- *High-speed data acquisition:* These are the functions that typically come to mind when considering a data acquisition system, and represent the main datapath for measured quantities from the detector up until permanent storage. This subsystem has stringent real-time and throughput requirements associated with the expected event rate in the experiment.
- *Slow control:* A complex DAQ system requires a dedicated control subsystem for the centralized programming of individual parameters of any component in the system, as well as starting and stopping runs, rebooting or updating firmware of single components, etc. These tasks may usually be carried out with relatively long latencies (hence the name), but require real-time

[8] Some authors actually consider the DAQ to be comprised of these stages only and separate from the front-end electronics. That distinction will not be made in this dissertation.

implementations, i.e. command execution needs to be guaranteed and its response time needs to have an absolute upper bound.

- *Monitoring and diagnostics:* The tasks under this category include the acquisition of super-event statistics like trigger rates, or hardware-specific test parameters and operating conditions in the detectors like temperatures and voltages. The collected statistics are used for the monitoring of correct experiment operation and possibly for self-adjustment or calibration; it must be noted that many shortcomings of a complex DAQ system cannot be predicted during design, but require system debugging under real working conditions [30]. These items are closely related with the previous category and often included in it; in fact, control processes for regulation of operating conditions make use of both monitoring and slow control functions.

The high speed data acquisition subsystem makes up the bulk of the DAQ electronics. The following functions are typically associated to it [29]:

- Trigger issuing, i.e. determining whether detected interactions should be recorded or ignored.
- Digitization of analog detector data from channels where actual data is present, and *zero suppression*, i.e. discarding data from inactive channels.
- Collecting subevent data from each detector subsystem in parallel and relaying it to a common higher level.
- *Event building*, i.e. collecting all subevent data from different detectors belonging to the same event and formatting it in an appropriate way.
- Optionally, online, high level analysis of the event, typically using software, in order to obtain a preliminary classification or to implement an additional event filtering stage.
- Transfer of the accepted event data to a computing center for permanent storage.

This complex subsystem may be studied as a distributed sensor network with the following characteristics ([31], pp. 1.5–1.19 and 3.5–3.13):

- The system contains few different sensor types, but they are replicated and distributed spatially in order to form large networks.
- Single sensors are intelligent in the sense that they do not merely output the measured electrical information, but rather elaborate it locally and extract

abstract information as a data compression step. The resulting output carries a semantic load and has reduced bandwidth requirements.

- Sensors are organized along hierarchic trees according to a predefined division of the event acceptance problem into smaller subproblems involving subsets of sensors. Each stage implies additional bandwidth reduction both through data integration and compression and through rejection decisions.

These conditions imply that the ideal DAQ system is both modular and scalable, in order to seamlessly support variations in the number of sensors and replacement of faulty nodes.

Communication of subevent data between hierarchy levels is routed through buffer queues. A decision from the trigger subsystem is required to determine whether to accept or reject the current data, i.e. transmit it upwards or discard it. Because of this, the generation of data from a subdetector is a random process even in accelerator installations with steady time distributions for the collisions. The buffer queues thus act as *derandomizers* or *traffic shapers* in order to isolate the next processing levels from the randomness of data generation rates ([31], pp. 2.49–2.51). Queue sizes can be designed to satisfy a target specification on event loss probability using techniques from queueing theory ([32], pp. 497–578). In any case, buffers must maintain the time correlation between successive events at all stages [18].

The data connections between subsequent processing levels may follow two different communication paradigms, known as *pull architectures* and *push architectures* respectively [33]. In the former, intermediate nodes are responsible for determining whether lower level nodes have new subevent data, pulling it, processing it, and waiting for communication requests from the higher level. In the latter, it is lower level nodes that make the decision of pushing their data upwards and higher level nodes act as passive acceptors. Note that detectors in push architectures may still need a trigger signal to enable transfers; the important distinction is that the decision to transmit data does not stem from the receiver, but rather from a separate trigger subsystem. Push architectures have been shown to achieve higher throughput [34] but require flow control mechanisms in order to avoid buffer overflows, as well as additional buffering capabilities in intermediate nodes if present [35].

The trigger subsystem

The trends in HEP experiments toward higher luminosities, large detector channel counts, and searches for interactions with lower cross sections imply an increase in the data rate generated by digitizers that contains background event noise. Although the available throughput scales with advances in communication technology, in the present day it is neither possible nor sensible to try to retransmit all captured data to the highest processing level [35]. The goal of the *trigger subsystem* is to detect and discard data belonging to background events as early as possible, in order to reduce the transfer and processing requirements of the DAQ and keep the data rates at reasonable levels. Triggering and filtering have become essential functions of data acquisition systems; although there is a clear conceptual divide between trigger and data acquisition, both are so strongly connected that they cannot be considered separately anymore [18].

Trigger is implemented in several levels [36]. The lowest level includes the local fast triggers that control the amount of data that are digitized. Subsequent trigger levels make decisions using increasingly complex algorithms based on event features extracted from digital data. In the usual jargon employed in many physics experiments, the lowest level trigger is referred to as the *level 0 trigger* or simply *L0 trigger*, whereas the next trigger levels are denoted *L1*, *L2*, etc. Typically, the L1 trigger is used for the transfer of data between the front-end and back-end electronics, and represents the earliest point where digitized data are discarded [37]. L2 usually refers to the decisions made by the DAQ components in the counting room. Trigger hierarchies rarely go higher than L2; additional filtering levels may be present that are sometimes referred to as L3 trigger and higher, but they are usually implemented as software routines.

In general, the design of the trigger subsystem and its partition into hierarchic levels is determined by a trade-off between efficiency and latency: increasing the complexity of trigger algorithms results in a larger fraction of unwanted events being discarded and thus in less strict specifications for throughput and computational load in the next stage, but it also increases processing requirements in the current stage, leading to higher latency and buffer sizes.

It is possible to design readouts without global trigger signals, or missing some trigger levels, particularly ones affecting the front-end modules, i.e. systems where all pulses are always digitized independently in subdetectors and transferred upwards to concentrator nodes. Such acquisition schemes are usually called *triggerless*. This does not mean that no trigger subsystem is present, but rather that the filtering decisions are postponed; in particular, all captured data is timestamped so that temporal coincidence can be resolved in the higher trigger levels. Triggerless hierarchy levels thus have increased output bandwidth requirements in exchange for lower latency or reduced output buffers.

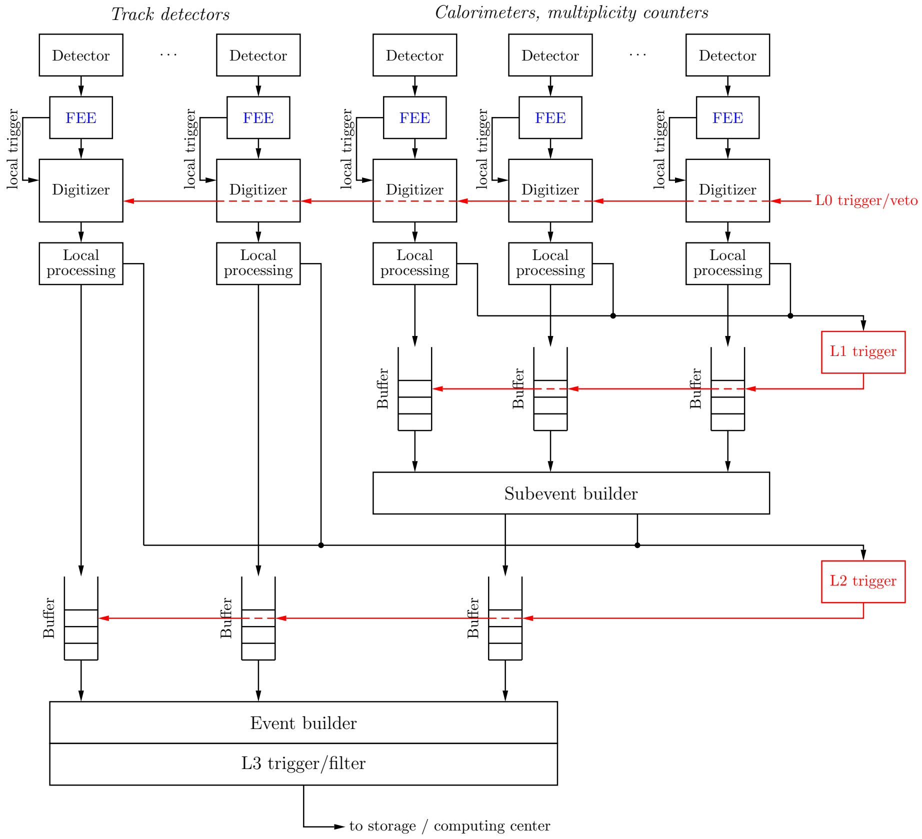


Figure 2.3: Example architecture of a DAQ system with multi-level trigger. The elements of the trigger subsystem are highlighted in red.

In a triggerless system, event information generated by each front-end module is always sent in chronological order and so the data received by back-end integrators from each node maintains temporal coherence regardless of variable latency. However, there may be large differences in the timestamps corresponding to subevent data arriving simultaneously at a concentrator node from different detectors. It is thus the responsibility of the back-end receivers to provide buffers that are large enough to accommodate incoming data from later events until the information from the slowest channel is received and all information from that event can be integrated. In essence, by eliminating the latency constraints, the requirements on buffer size are increased.

A prototypical DAQ and trigger example

Figure 2.3 shows a fairly typical architecture of a DAQ system for a complex HEP detector with multiple trigger levels. Each subdetector is attached to its own subset of analog readout electronics, capable of generating fast trigger signals on pulse detection that act as zero suppressors ([31], p. 3.22). The actual local trigger decision is modified by a global level 0 trigger which, in a collider experiment, may be associated with the temporal structure of particle collisions, i.e. the bunch frequency clock. Alternatively, a trigger veto signal may be employed that prevents triggers for a fixed amount of time after an event is detected, e.g. if detectors have a long recovery time after firing.

Digitized signals undergo a local processing stage before the result is sent to the next hierarchy level or rejected. Subdetectors can be divided into two groups according to the amount of signal processing needed to obtain useful data from them that can be used for trigger and filter decisions. On one hand, detectors like particle counters and calorimeters are used to generate a small amount of scalar data such as multiplicity i.e. amount of detected particles, energy and charge sign, enough to identify some particles ([13], pp. 38–43). Some examples of early particle identification criteria can be found in ([13], pp. 307–316). The local processing requirements are light, possibly including linearity corrections using LUTs, filtering, window discrimination, and a few elementary logic and arithmetic operations. The results on amount and identity of detected particles are thus available with a short, fixed latency and can be used to generate an early L1 trigger that filters uninteresting events out and dramatically reduces system bandwidth ([13], pp. 49–52).

On the other hand, track detectors feature many more signal channels and require much more intensive digital processing in order to fully extract the information that yields the complete event reconstruction. Nonetheless, it is possible to obtain coarse descriptions relatively fast by recognizing tracks using brute-force combinatorial processing of hit records corresponding to whole wire clusters rather than single wires, as long as the total number of tracks is higher than 2 ([13], pp. 52–62). This allows coarse tracking information to be merged with calorimeter and particle identifier data in order to obtain an early classification of the event that is enough for a L2 trigger. This trigger decision can be very selective and has the main purpose of discarding data from tracking detectors, which make up the bulk of the event data ([31], pp. 3.22–3.25).

Data accepted by the L2 trigger are formatted by an event builder subsystem and may as well be stored permanently for further analysis, however many experiments choose to apply an additional filtering level in order to reduce the final data volume. By analogy, this processing stage is sometimes called a L3 trigger, or *high-level trigger*. It usually involves almost full event reconstruction in order to derive subtle properties that may be used for discrimination. The tasks involved in event reconstruction are ([9], pp- 18–26)

- *Tracking:* Individual hits are grouped into tracks corresponding to the passage and detection of the same particle at different subdetector components. This is a refined version of the coarse estimates obtained for the hardware triggers.
- *Geometrical fitting:* Once all hits from a given particle are identified, the exact track parameters are obtained by fitting to the detection points.
- *Vertexing:* Separate tracks originating from a common point or *vertex* are grouped into a single particle decay. Decay positions are thus detected and the complete event history is reconstructed.
- *Kinematic fitting:* The mass of the particles corresponding to each track is obtained with information from particle identifier detectors.

All of these tasks are computationally intensive and naturally data-driven, based on algorithms with variable execution times, so they are usually implemented in software, possibly in dedicated computer farms ([24], pp. 18–20).

It should be noted that there are large variations between the structure of DAQ and trigger systems for different experiments; in fact, it is pointed out in [29] that all DAQ systems are state of the art and therefore different, due to the cost of developing a new one from scratch as compared to adapting an older, working architecture. In particular, the number of hardware and software trigger levels may vary, as well as their latency requirements and their input and output scopes. Nevertheless, the described architecture is representative enough and many well known experiments adhere to it with few variations, such as the four main experiments at the LHC: almost exactly for ATLAS, while CMS, ALICE and LHCb also use coarse tracking information for L1; the two latter also feature a previous calorimeter-based L0 trigger [37].

Dead time

The most important parameter determining the performance of a data acquisition system is the *dead time*. It is defined as the time after each event during which the system is not able to record another event ([38], pp. 122–127). This may take the form of a non-sensitive time period where any new event signal is not detected and therefore rejected, or of an event-defining time window that treats any incoming signals as part of the same event even if they actually belong to separate, independent events; in the latter case, it may be possible to separate piled up events at a higher processing level, but the lack of specific triggering for the second event may lead to incomplete event capture.

Global system dead time is caused by finite response times of individual components. Most elements in the system may contribute to dead time:

- Detectors often have an intrinsic recovery time due to their physical working principles, e.g. transit time of charges, shower duration, or relaxation times.
- Analog front-end with pulse shapers may introduce a dead time due to the shaping time if pile-ups cannot be resolved.
- Trigger logic may also have a reset time given by the length of the trigger pulse; if the trigger signal is sampled synchronously, then the clock period also imposes a restriction. In particular, spurious triggers due to noise may mask real events.
- Digitizers, particularly TDCs and slow on-demand ADCs, may become unresponsive during conversion.
- The signal processing stages may cause additional dead time if they contain sequential, non-pipelined online operations. Their effect on dead time can be minimized or negated using techniques such as parallelism, pipelining and buffering.

In an accelerator-based setup, the time windows for event capture are well-defined; often, a single interaction may be detected per bunch, and no interaction is expected until the next bunch. Hence, a dead time lower than the bunch period guarantees that no event will be lost [29], i.e.

$$T_d < \frac{1}{f_b} \tag{2.2}$$

where T_d is the dead time and f_b is the bunch frequency. It is therefore possible to implement slow blocking software routines as long as their execution time keeps dead time below that value reliably.

On the contrary, in setups with spontaneous events, dead time will always result in an event loss rate due to the probability of separate events taking place with a small time difference. In such a system, where the arrival of events may be described by a Poisson process with event rate λ, the expected fraction of detected events is [29]

$$\eta = \frac{1}{1 + \lambda T_d} \tag{2.3}$$

while the rest are lost. The implementation of multi-level triggers may increase this efficiency value ([13], pp. 15–18).

2.2 Evolution of data acquisition systems

2.2.1 Historical evolution

The first particle physics setups were characterized by the use of optical readout techniques, with data acquisition being carried out first by simple ocular inspection and recording results by hand, and later using photographs. Well-known primitive examples include Rutherford's gold foil experiment in 1909 demonstrating the existence of the atomic nucleus [39], and medical radiography, which still uses the same methods since its inception in 1896. Later on, with the development of accelerators and the search for new subatomic particles in full swing, the use of track detectors became widespread, starting with the cloud chamber in the 1920s [40] and followed by the bubble chamber in the 1950s [41]; in both cases, particle trajectories left a visible trail in the detector and photographs were used to store event results.

By the end of the 1950s, the limitations of this approach were apparent. The push for higher luminosities and events with a smaller cross section was increasing the interaction rate beyond acceptable levels for detectors with such a high dead time ([24], pp. 11–12) and, together with advances in component miniaturisation, electronics and computer technology, lead to the rise of electronic DAQ. In particular, the development of transistor-based amplifiers and ADCs allowed them to find their way to detector front-ends, and the advent of the minicomputer established it as a very convenient readout method. Meanwhile, tracking detectors with electronic readout increased their resolution by reducing detector pitch and thus increasing the number of channels to be read, making it impractical to implement point-to-point readout for all digital channels; thus, DAQ systems based on instrumentation buses became the norm [42].

The advent of modular DAQ

The proliferation of HEP experiments in the 1960s lead to the development of modular solutions for front-end and readout electronics. Detectors would be connected with cables to electronic modules arranged in a chassis or crate with card guides for mechanical support and accessibility. The crate would typically feature a *backplane*, i.e. a common board with physical connectors for individual module insertion that connected corresponding pins of different modules together, either for communication or for distribution of common signals, e.g. supply voltages. Eventually, crates would also be used to house power supplies and cooling.

A number of incompatible systems emerged initially until the first standardization efforts took place. The NIM (Nuclear Instrumentation Module) standard was the first to appear in 1964 [43], with a strong emphasis on the front-end, where the majority of the electronics was concentrated at the time. It specified mechanics (crate definition, module footprint and connectors), connector pinout and signal levels, however it provided no means of communication between different modules in the crate, i.e. no backplane bus. Despite its inadequacy for digital readout, NIM became ubiquitous in a couple of years [44]. Today it is still widely used for modules that require no digital data communication, like amplifiers or pulse generators.

In order to complement the shortcomings of NIM, the CAMAC (Computer Automated Measurement And Control) specification came out five years later [45]. Besides mechanical definitions for crates and backplanes, the standard introduced a digital backplane bus with support for 24-bit data transfers at 1 MHz and a control protocol which featured a single crate controller per CAMAC crate acting as a bus master, with all other modules acting as slaves. This scheme was appropriate for the available technology and requirements of the moment, and so most HEP experiments in the 1970s used a DAQ based on NIM modules for the front-end and a CAMAC bus for readout from a minicomputer [33].

Parallelized DAQ

Several factors started to highlight the limitations of this style of DAQ at the dawn of the 1980s. As HEP experiments evolved, the search for new interactions with a lower cross section forced an increase in luminosity and the introduction of complex detectors consisting of several independent subdetectors, as well as a higher number of channels for track detectors as their resolution improved. This implied an increase in bandwidth requirements, but also in the amount of uninteresting background events which needed to be filtered out in order to yield an output data rate that was manageable by storage systems at the time. Thus, it became mandatory to establish a trigger subsystem for event filtering [33].

Additionally, the flaws of a centralized DAQ architecture with a single host became apparent as the division into subdetectors favored partitioning the DAQ into autonomous subsystems that could be designed, tested, and upgraded separately; the development of the microprocessor in the 1970s was making this an affordable approach. These problems were first addressed by releasing multi-crate extensions for CAMAC, but they were cumbersome to use and didn't solve the problem of limited bandwidth due to CAMAC being based on old technology [42]. Later, in 1983, the FASTBUS specification was published [46], intended as a replacement for CAMAC. It substituted the TTL signaling standard for the much faster ECL, defined an asynchronous 32-bit bus protocol that allowed multiple masters, and supported multi-segment architecture by design.

Meanwhile, the Motorola 68000 microprocessor was introduced. Although it was not the first 16-bit device to appear, its architecture was particularly well-suited for bus-oriented applications and, in fact, the 68000 was released with a proprietary backplane bus specification. The popularity of the processor lead to the publication of the VMEbus standard in 1981 [47], which was basically an adaptation of the proprietary 68000 bus to more suitable crate mechanics; it was later enhanced with the VICbus specification in order to allow inter-crate communication [48], and by VXI (VME eXtensions for Instrumentation).

Both FASTBUS and VME became popular architectures for HEP DAQ, with CAMAC still finding some use due to the amount of available, time-tested equipment [42]. FASTBUS had several advantages over its competitor, namely larger board sizes more suitable for bulky front-ends and native multicrate support with higher bandwidth. In the 1990s, however, VME ended up dominating the scene due to industry support and ease of use [49, 50].

From data buses to serial links

Parallel, multi-drop data buses had been a staple of data acquisition systems for three decades but were already reaching their physical limits. In a parallel bus, maximum bus transfer frequency is related to the skew between different wires forming the parallel bus, including data and clock or strobe lines.[9] Additionally, multi-master buses need to allocate available bandwidth not just among all masters but also to the bus arbitration mechanism. Point-to-point links, on the other hand, eliminate the arbitration overhead, may achieve higher speeds due to reduced capacitance, and allow simultaneous transfers in different segments. Besides their capability for higher throughput, an additional advantage of serial links over parallel buses is that they enable true real-time behavior, i.e. deterministic, completely predictable response times [51].

During the 1990s, DAQ systems started gradually replacing their backplane buses with multiple serial links at different hierarchy levels as their local data bandwidth became the system bottleneck. Existing mechanical frameworks were not discarded; instead, the bus infrastructure could be exploited for less intensive tasks like configuration and power distribution [33], or by using non-standard connector pinouts and custom backplanes. The only exception to this trend may be the PCI bus [52], which started being implemented in PC motherboards in the early 1990s and enabled the integration of DAQ modules within cheap off-the-shelf PCs with transfer rates starting at 133 MB/s.

At that moment, many sectors within the IT industry were striving for increases in data transfer bandwidth, so a large number of technologies and standards were released; many specifications that were initially designed for a different sector found their way to HEP communities because of matching requirements or specific niche features. In the physical layer, the most important contributions were

[9] This limitation will be explained in greater detail in §3.1.1.

the now-ubiquitous Low Voltage Differential Signaling (LVDS) [53], an electrical signaling standard for gigabit-rate serial transmission over inexpensive cables, and the advances in optical transmission, which is specially useful in HEP due to its electromagnetic immunity. Other successful data transmission technologies had been conceived for Storage Area Networks (SAN), like Cypress' Hotlink [54] and Agilent's G-Link [55], and Local Area Networks (LAN), like Fast Ethernet [56]; in particular, the mass adoption of switched 100 Mbps Ethernet as a networking standard made it an extremely inexpensive choice. A comprehensive review of the competing transport technologies at the time can be found in [50].

In the first few years of the new century, serial links evolved into the multigigabit-per-second range, starting with Gigabit Ethernet [57], whose compatibility with previous incarnations of Ethernet guaranteed its popularity. Optical technology advances were also developed for increased throughput, either by aggregating parallel links into a single fiber using Dense Wavelength Division Multiplexing (DWDM) [58] or by using composite optical cabling. DAQ systems for experimental physics benefited from this by shifting towards point-to-point serial links at all levels including the front-end, with the underlying motivation of moving the digitization step closer to the detectors in order to increase resolution [59].

Integration with ASICs and FPGAs

The integration of microelectronic design within HEP DAQ had started in the early 1980s in the form of front-end ASICs [60]. As ASIC technology became affordable, both in terms of economic cost and development time, it became an increasingly attractive option for the implementation of the electronic stages closest to the detector, like amplifiers, shapers and digitizers. In [18], four key characteristics are identified that explain the suitability of ASICs to HEP front-end: high circuit density, reduced power consumption, support for custom specifications which may not be commercially available, and the possibility of radiation-tolerant design. All these aim at the implementation of fully custom, dense front-end designs that can be mounted as close as possible to the actual detectors. Consequently, front-end ASICs have been recognized as an essential component in large HEP experiments since the 1990s [61].

However, ASICs also present some drawbacks related to the organizational aspects of HEP experiments. They are fixed, monolithic, potentially very complex designs with a comparatively long development cycle, in a setting where setups take years to plan, build and deploy.[10] It is thus not possible to substantially modify the circuit design after the experiment starts and its quality can be fully assessed. This is less of a problem for analog front-end electronics, where detector signal parameters can be estimated accurately beforehand, but it is an insurmountable

[10] Deployment is often irreversible, particularly for the electronics closest to the detectors, which are extremely expensive to disassemble.

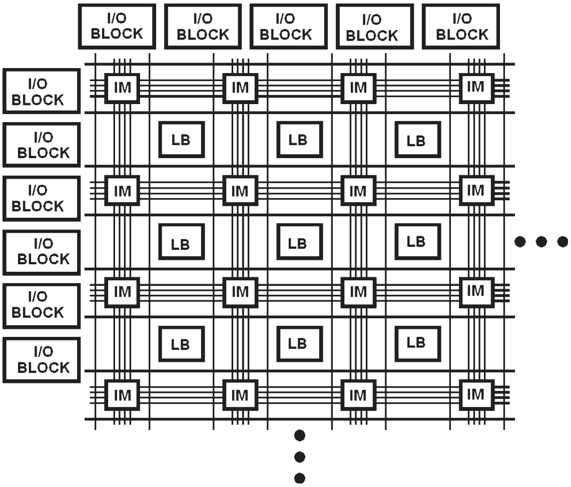


Figure 2.4: Basic structure of a FPGA (LB: logic block, IM: interconnect matrix). Reproduced from [62], © 2007 IEEE.

obstacle for the trigger and integration subsystems, where digital algorithms are expected to evolve considerably over the course of the experiment [30]. In this context, programmable logic is an appealing alternative.

The first generation of programmable logic devices was released in the 1970s and supported the implementation of small custom combinatorial functions with a few flip-flops for memory; they were small circuits with a few hundred equivalent gates and a full mesh interconnection topology between elements. In the 1980s, as these devices evolved into CPLDs following a sea-of-gates model, FPGAs were conceived and released as an alternative paradigm for programmable logic. They consisted of a bidimensional array of logic blocks, originally formed by logic elements hosting a simple combinatorial function and a flip-flop; logic blocks were interleaved with interconnect buses that intersected at programmable switching matrices, and universal I/O blocks were placed along the perimeter of the array. This structure is depicted in Fig. 2.4.

Simple programmable logic devices had been used in HEP DAQ since the 1980s for glue logic and very simple control circuitry [62]. Although the specifications of primitive FPGAs were not particularly impressive, they started being adopted in the 1990s in trigger subsystems mainly because of their dynamic reconfiguration capabilities, which allowed circuit modifications without having to access radiated areas physically [18]. During this time, FPGAs evolved in three key aspects: their logic capacity increased as fabrication processes offered higher integration levels, optimized design flows were established, and devices started incorporating dedicated RAM blocks alongside programmable logic blocks. Complex, specialized algorithms that would have been run on PCs or DSPs before could then be implemented more efficiently as logic operations, parallelized, and integrated with control logic and small memories together in a single device [33]. By the end of the 1990s, FPGA technology was mature enough and it became the platform of choice for the implementation of logic circuitry due to the high compatibility between commercial FPGAs and the actual needs of DAQ systems [18].

At the turn of the century, the trend of incorporating FPGA technology in HEP DAQ continued as FPGAs evolved to include an assortment of dedicated hardware blocks for specific functions. The introduction of hardware multipliers, later superseded by more complex DSP blocks, eased the implementation of signal conditioning and higher level event filtering that includes track analysis on a local level [18]. Support for high speed signaling standards such as LVDS was also included with special I/O blocks, integrated serializers and deserializers, and PLLs and DLLs.

2.2.2 Current and future DAQs

The experimental setups that are currently active and their associated DAQ systems were mostly designed and built during the first decade of the 21st century, so most of them follow the principles described at the end of the last section. In particular, clear trends towards early digitization of detector signals, heavy use of FPGAs at all hierarchy levels, and communication based on serial point-to-point links have been observed, and triggerless readout is becoming the preferred scheme in the last few years. Additionally, some newer managerial and economical considerations have been raised in the last years that have strongly conditioned the design of new DAQ electronics, regarding system uptime, cost and reusability. All of these items are detailed in this section.

FPGAs and serial links

There are two concurrent developments in the evolution of FPGA devices that have defined a turning point regarding their use in experimental DAQ. One of them is the inception and integration of embedded processor cores, either as synthesizable IPs like Nios II [63] and MicroBlaze [64] or as hardware blocks like PowerPC cores on Xilinx' Virtex series [65], giving rise to the concept of *platform FPGA* in 2000 [66]. In this way, a FPGA can be used as a reconfigurable prototyping platform for a System-on-Chip (SoC) [67], i.e. a full digital system embedded into a single device, with a microprocessor for control and management and a completely custom, fast, deeply pipelined datapath for signal processing [68].

The second important step is the inclusion of gigabit-class transceivers that include all necessary blocks for the establishment of a high-speed serial link, e.g.

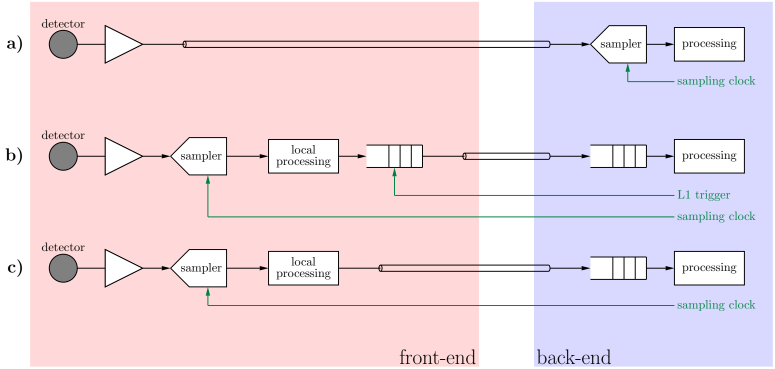


Figure 2.5: Three generations of DAQ readout schemes: a) Remote sampling, b) Local sampling, c) Triggerless.

clock recovery, serializers and deserializers, and line coding and decoding, starting in 2002 [69]. The general trend has become to use high speed point-to-point serial links at various levels within systems, encapsulate information flows in packets, and bind multiple channels together at the logical instead of the physical level in order to increase throughput [50]. FPGAs were already the technology of choice for interface to raw high-speed data links due to their ability to handle data formatting and transport much more efficiently than other computing devices such as microprocessors, DSPs or GPUs [35]; the inclusion of embedded transceivers further adds to their idoneity by removing the requirement of external transceiver devices.

As a result, two distinct stages can be identified regarding the use of FPGAs in HEP experiments. The first one, including the 1990s and the first few years of the new millennium, was characterised by their use mainly for integration and reconfigurability reasons. Starting in 2005, the SoC era is marked by an increased use of platform FPGAs as full systems on chip that help avoid dedicated hardware developments for prototyping [67], and a generalized use of serial links for point-to-point communication between DAQ modules. Practically all current DAQ systems rely on FPGAs for the readout of detectors.

Early digitization

Let us recap the different possible readout schemes that have already been presented. As outlined in Fig. 2.5, there are essentially three ways in which the readout of part of a DAQ system may be organized in terms of the location of sampling and trigger elements along the processing chain, corresponding to three different generations of DAQ systems.

- *Remote digitization*: This is the oldest but also the simplest readout scheme, and corresponds to the case of detector signals being transmitted and sampled at the back-end, after being preamplified and undergoing shaping or other kind of exclusively analog preprocessing at the front-end.

- *Local digitization*: The second scheme has been predominant in most large setups and involves moving the sampling step to the front-end cards. Zero-suppressed samples are stored in a local queue, possibly after some local digital processing steps, where they await an external L1 trigger decision that results in them being either discarded or transferred to the back-end, to be processed further after derandomization.

- *Triggerless*: This scheme is similar to local digitization, with the difference that no global trigger decision is applied at the front-end. Local triggers for sampling decisions and zero suppression may be present, but no sampled data are discarded. Instead, they are always transferred to the back-end, where the decision is taken whether to ultimately drop them or process them further.[11]

Comparison between DAQs with remote and local digitization is determined by the implications of locating the sampling step at the back-end or the front-end sections. In the former case, communication between the front-end and the back-end is exclusively analog, and all digital circuitry is located at the back-end. In particular, there is no need to relay the sampling clock or trigger decisions to the front-end, or to synchronize the front-end electronics in any way. Trigger and synchronization are relegated to the back-end, where they can effectively be implemented with boards mounted on backplanes with controlled-length connections.

On the downside, this scheme usually implies heavy losses in analog signal integrity due to transmission. Copper cables are used in general, with a limited length capability, and even then they cause attenuation and distortion before the detector signals are digitized, severely hindering its resolution. Moreover, all detector signals need to be transmitted to the back-end, potentially resulting in an extremely high cabling volume for fine-pitched detectors unless some multiplexing scheme is applied. These disadvantages are important enough that most experiments with relatively high resolution needs or channel counts have abandoned the remote digitization scheme altogether, although it is still being employed in new developments for relatively simple setups such as [71] due to its simplicity.

Local sampling schemes, either triggered or triggerless, can achieve optimal signal integrity by digitizing detector signals as close to the detector as possible. In very

[11] It should be noted that the sampler element is not necessarily an ADC. This distinction is important when analog memories are involved in the readout process. See [70] for a recent example of a triggerless readout scheme where samples are transferred to the back-end in analog form.

old systems, such implementations were unfeasible due to circuit size, but miniaturisation trends have now allowed the integration of complex digital electronics at the front-end. Additionally, links between the front-end and the back-end may be fully digital, which enables the use of optical fibers and thus of long distances. Cabling requirements are reduced by serialization if the digital links support the required data rate. One further advantage is given by the fact that digitized signals are immune to jitter in any subsequent clocks after sampling, hence early digitization reduces the impact of clock jitter on the resolution of measurements.

Disadvantages of local sampling schemes are related to the management of the control signals associated with the sampling and triggering functions, as they are generated at the back-end but need to be transmitted to the front-end. The main problem is the synchronization of these signals at the different front-end nodes in order to maintain sampling alignment where needed and not to break the time relationship between different subevents, i.e. coincidence and ordering between them, as it is essential to trigger decisions. In the case of triggerless systems, there is no need to transmit trigger decisions within a constrained event window, but the problem of synchronization is still present. Chapter §3 will be entirely devoted to this issue.

Triggerless systems

In general, it would be advantageous to reduce the front-end electronics to a minimum with the simplest possible link to the back-end section, due to the fact that the front-end environment imposes stronger constraints on items such as area occupation, power dissipation or radiation hardness, often forcing the use of custom ASICs. Any element in the processing chain that can be located at the back-end will be able to use commercial devices with lower cost and higher performance, not having to care much about size or power, and with increased ease of maintenance [72]. Unfortunately, implementing the sampling step at the back-end introduces too much signal distortion and loss of resolution; for this reason, the sampling operation is considered one of the essential functions at the front-end in most experimental setups.

After digitization, no further loss of information is possible in the detector signals,[12] so it should be a good approach to end the front-end chain at that point. Unfortunately, the data volume generated by the sampling step was too high to be successfully transferred to the back-end with older, slower communication technology; in fact, the L1 trigger was introduced at the end of the front-end chain as a means to reduce the bandwidth between the front-end and the back-end. Hence, until recently, the L1 trigger has been considered another essential part of front-end electronics purely because of technological limitations in digital data

[12] Barring bit errors in data transmission, of course.

transmission. It may therefore be argued that advances in digital communication will progressively make the triggerless readout scheme more attractive as digital links become able to handle the volume of zero-suppressed data from the detectors. In other words, triggerless readout is optimal whenever it is technologically feasible.

Besides the general advantage of relocating elements from the front-end to the back-end, more specific advantages of triggerless systems are listed below:

- While the implementation of trigger algorithms in programmable media such as FPGAs allows their reconfiguration after the electronics have been deployed, the data flow topology remains fixed, i.e. the decisions about which detector signals go to where and which decisions they are used for. In a triggerless system, all detector data reach the back-end and there is greater flexibility without the need to update front-end hardware. This makes triggerless readout particularly appropriate in experimental setups where a large number of different physics cases is to be accommodated covering separate research topics, as such a requirement implies the detection of many interaction types requiring vastly different triggering criteria [73, 74].

- Even if the data flow topology remains the same, in some experiments where low-level triggers are too selective it is possible to increase their captured event rate by removing the L1 trigger altogether and going triggerless, implementing all filtering decisions in complex software routines. This vastly increases bandwidth requirements but the trade-off is advantageous in upgrades to many existing experiments such as LHCb [75].

- In experiments such as [76] where very delayed coincidences are used as trigger conditions, i.e. subevents taking place with long time differences over 10 µs, implementing a L1 trigger might result in increased dead time for the readout system. Switching to a triggerless scheme resolves this issue.

- In large DAQs that include intermediate readout buffer modules, i.e. data concentrator cards that merge links from multiple detectors, their design can be quite generic as these modules all perform the same functions, acting as simple switches and mergers without filtering capability. They can thus be reused between subdetectors in the same experiment or between different experiments.

- Local digitization schemes with L1 trigger impose specific latency windows for trigger decisions and often force deterministic transmission latencies between the front-end and back-end and the alignment of clock signals in all front-end cards. In triggerless systems, the L1 trigger is removed and coincidence between signals is based on timestamps and resolved a posteriori, so these requirements are lifted and system synchronization is simplified. This advantage is only rarely exploited by experimental DAQs (such as [77]) due to inertia from well-established methods and available solutions from the previous generation of DAQs, particularly in upgrades of existing systems, but it will be a central point in this dissertation and expanded upon in §3.4.2 and §3.5.

Consequently, one of the clearest current trends in large DAQs is the shift towards triggerless readout [36], with the ultimate goal of standalone detector modules with a single high-speed digital output link that contains all valuable information [78]. A large number of data acquisition systems in the last few years or currently under design follow the triggerless paradigm. Important examples are given by all four experiments at the LHC: triggerless readout electronics have already been proposed or designed for all of LHCb [75] and some subdetectors such as the calorimeters at ATLAS [79] and CMS [80]; ALICE is also planned to go triggerless after the next long LHC shutdown [35]. Other examples of new triggerless systems include the CLIC linac at CERN [81], the Mu3e experiment at PSI [82] and the PANDA and CBM experiments at GSI [83, 84].

Availability

As the size and complexity of detector systems has expanded, a new, unforeseen requirement for DAQ standards has been realized. Measurement campaigns are typically very long and scheduled well in advance, and any downtime becomes extremely expensive; for instance, the cost of lost beam time at the proposed International Linear Collider is estimated at \$135 000/h [85]. Thus, high availability for uninterrupted operation becomes a prime concern as system complexity increases along with the number of potential failure points [44]. Focusing on that requirement as well as high throughput, the ATCA (Advanced Telecommunications Computing Architecture) standard was released in 2004, featuring native all-serial communications, redundancy, and mechanics that support true hot-swappable modules, allowing the replacement of faulty components without disturbing system operation [78].[13] Although the original intended target was carrier-grade telecommunications infrastructure, ATCA and the related µTCA standard have soon become accepted in the physics community[14] due to its suitability for high bandwidth, high availability real time applications [86], and specific extensions for physics have been actively discussed since 2009 [87]. An extensive list of HEP installations that have adopted ATCA/µTCA can be found in [51], as well as a comparison with its competitors VPX and CompactPCI Express.

[13] Note that this feature is practically unimplementable in cableless parallel bus systems, since the addition or removal of a bus node is likely to induce a nontrivial electrical disturbance in the shared medium.

[14] Acceptance became particularly widespread after mass adoption of ATCA for the LHC at CERN.

Modularity and reusability

Data acquisition systems for large detectors often require a vast amount of electronic modules to implement the whole readout and trigger scheme, including any intermediate data concentrator nodes. Once the acquisition hierarchy is designed, the total amount of electronic boards in the system is possibly fixed. It is then often not possible to reduce the cost and complexity of the system by decreasing the number of modules. However, the same can be achieved by reducing the total number of different module *designs* comprising the electronics.

The implementation of completely generic front-end boards is impossible due to the dependence with the corresponding detector, which forces specific constraints such as size, shape, or power consumption limitations. However, the operations performed after signal digitization are always relatively similar between different detectors or different experiments. This makes it possible to define and design generic readout modules that may be used in different subdetectors at the back-end.

The primary advantage of the reduction in the amount of electronic designs is the cost reduction given by cutting fixed costs, particularly man-hours dedicated to design and testing. Another important advantage is the reusability of the modules, which synergizes with system scalability as it permits changes in the topology of the readout system, such as modifying the number of detectors or modules dedicated to one task, without discarding the associated electronics but rather assimilating them in different subsystems.

Full modules that are completely generic are often not possible to design. However, the concept of electronics modularization and design for reusability and their advantages can be applied at several other levels:

- At the hardware level, by splitting electronic modules into submodules that carry out smaller parts of the signal processing chain, each of which is generic enough that it can be reused as part of another module. For instance, front-end electronics might be split into two boards attached through a connector: a detector-dependent one receiving the pulses and performing adapted shaping, and a generic one handling digitization and communication with the back-end. Back-end readout modules may also consist of generic base designs with added connectors for *mezzanine boards* that carry out additional functions or communicate with subdetector-specific interfaces. For instance, the ATCA standard natively includes support for up to 4 several Advanced Mezzanine Cards (AMC) per board, as well as Rear Transition Modules (RTM) for the addition of custom cabling interfaces [78].
- The principle may also be applied at the firmware level by designing generic boards containing reconfigurable devices such as FPGAs and a sufficient

amount of assorted connectors for the intended applications, and then implementing different firmware in each board so that they perform different functions. At higher levels, the same is accomplished by using microprocessor-based nodes with different software for each case.

- A different approach to reusability is the establishment of collaborations between different experimental teams whose electronics needs are similar enough that they can be merged into identical designs that are still almost optimal for all intended targets. This consideration is becoming increasingly important as the funding of public research is reduced.

Chapter 3

Synchronization in DAQ systems

As has been described in chapter §2, DAQ systems for physics experiments may include very large, spatially distributed digital systems, particularly when early digitization schemes are used. Temporal coherence between all sensing nodes is vital for the correct reconstruction and interpretation of captured events, but their synchronization is a tricky issue in light of their large amount and the distance between them. To give just one example, the variations in the propagation delay of long transmission media due to seasonal drift in LHC experiments is reported in [88] to be around 7 ns - well over the expected resolution for satisfactory operation.

In this chapter, the essentials of synchronization of digital systems are discussed, focusing on the particular case of DAQ systems. The discussion starts with a generic treatment, establishing the necessary concepts and terminology, and then gets progressively specific. The most common method for the synchronization of the time references in distributed nodes over non-dedicated data links is thoroughly described and analyzed, the shortcomings of its usual implementations are highlighted, and it is shown how to overcome them. A review of the existing methods that take these considerations into account and the available hardware solutions that support them is included in §3.4. Finally, the specifics of the synchronization method proposed in this thesis are detailed, and a particular proof-of-principle implementation is described and evaluated.

This is the central chapter of this dissertation and contains the main theoretical results and proposals, as well as a crude first experimental validation. The core of its contents was presented at the Real Time Conference in 2010 and published in 2011 [7]. At that moment, research on these topics was at an early stage and there were only a few published references by a couple of groups. The exposition has been expanded here in order to include more recent developments.

3.1 The concept of synchronization

In general, *synchronization* is the process of matching and aligning the time scales between two or more processes that take place at spatially separated points [89]. As has been noted in chapter §2, complex DAQ systems rely on the detection of subevents that are detected by different sensors and tagged by front-end electronics located at separate physical locations. Synchronization of their respective time references is therefore crucial in order to maintain the relative timing information between them.

Two particular time relationships between events or subevents are of importance in a typical DAQ and must be preserved: chronological ordering and time coincidence, the latter being defined with respect to a given coincidence window. It is accepted that perfect synchronization is physically unfeasible due to the finiteness of the speed of information communication; the expectation for DAQ systems is therefore that synchronization can be maintained at resolution levels that do not substantially affect the aforementioned ordering and coincidence conditions, i.e. that the synchronization error is well below the time resolution of the system. Of course, this specification depends heavily on the particular DAQ system under consideration.

In the literature, the term *synchronization* is employed to refer to several different, well-defined processes. An attempt will be made in this thesis to establish a clear distinction between them by assigning different names:

- The distribution of a common clock frequency, or a family of related frequencies, throughout different parts of the DAQ system. This is usually achieved by transmission of a given master clock, using PLLs and synthesizers to replicate the clock and generate different frequencies with a fixed, rational factor. This process will be referred to as *syntonization* or sometimes *clock distribution*.

- The establishment of a common time reference for a given subset of timestamping electronics. This is a necessary step in order to guarantee that all events that are assigned a time value by the DAQ can be later ordered chronologically and that the relative time difference between events can be estimated up to a certain resolution. This process will be called *time synchronization*, *clock synchronization* or simply *synchronization*.

- Sometimes, the term is used to mean the establishment of a communication link at the physical level, up to the point where transmission and reception of individual data bits is possible, while their interpretation is up to higher level protocols. This process will be referred to as *data synchronization*, *link synchronization* or *link establishment*.

A few other related concepts are sometimes encompassed under the same term that are not relevant to this work, so they will not be discussed here. The main focus will be on syntonization and time synchronization.

3.1.1 Clock distribution and data synchronization

The operation of digital systems is governed by clock signals whose edges indicate the appropriate time instants for certain actions, mainly the latching of values in registers, but also the capture of samples in digitizers in the particular case of DAQ systems. In this section, the distribution of timing signals in a generic digital system is considered with regard only to the former, i.e. focusing on the aspects of syntonization and data synchronization.

Currently, there are essentially three different paradigms for data synchronization within digital systems. In increasing order of complexity and potential performance, they are usually called *system-synchronous*, *source-synchronous* and *self-synchronous* signaling schemes ([90], pp. 178–193). These techniques encompass standards for timing schemes and data signaling that guarantee interoperability and (generally) syntonization between its component devices, and have associated clock distribution schemes whose design is subordinate to the correctness of data communication. Time synchronization between devices is not yet taken into account because these concepts apply to generic digital systems; the discussion will be extended to the main topic of time synchronization later.

System-synchronous clocking

In a system-synchronous scheme, all digital devices operate synchronously with a common time reference, using syntonized clock signals. The time difference between the active edges[1] in two different clock lines with the same frequency is referred to as *clock skew*. In an ideally synchronized system there is no skew between clock signals as they arrive to their target devices; in practice, large skew values between different devices are possible if the clock frequency is low enough.

The most simple and archaic form of clock distribution in this type of system is to distribute a common clock signal to all synchronous devices using a daisy chain topology, as shown in Fig. 3.1. This method entails multiple problems, including large clock skew values (equal to the propagation time of the clock signal

[1] More specifically, between the input threshold crossings of said edges, in order to cover the case where the receivers are not identical or even use different logic levels.

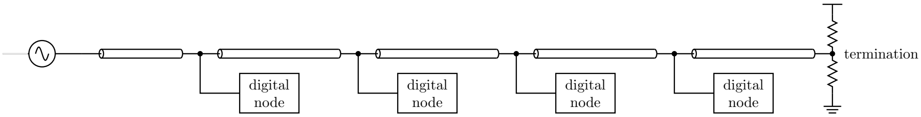


Figure 3.1: Clock distribution using a daisy chain topology.

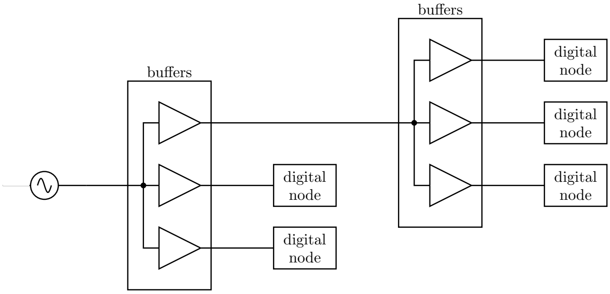


Figure 3.2: Clock distribution using a clock tree topology.

between different devices), a large capacitive load on the single clock line, and potential signal integrity issues due to multiple impedance discontinuities. These effects result in severe limitations on achievable clock frequency and hence on data transmission throughput, besides requiring a clock source with large current output capabilities.

Typical clock distribution networks avoid these problems by limiting the amount of nodes that each clock source feeds. In a system-wide or PCB setting with high operating frequency, each fast clock line is usually limited to one destination node for improved signal quality. Clock replication devices are then required in order to generate the appropriate number of clock signals. The resulting structure is a *clock tree* as outlined in Fig. 3.2, with a single oscillator as the root node and clock drivers at intermediate nodes that generate multiple copies of the input signal. The connections between nodes may be implemented using cables in order to distribute the clock between separate modules.

Replication devices used in clock trees can be divided into two types depending on their working principle. The simplest are simply called *clock buffers* and consist of arrays of buffer (i.e. non-inverting) logic gates with a common input. General-purpose logic buffers with shorted inputs may be (and have been) used, but it is preferable to employ devices designed specifically for clock distribution as they offer improved performance in parameters such as input capacitance or *output-to-output* and *part-to-part skew*, i.e. maximum skew between outputs from the same or different devices, respectively. Clock buffers may include clock frequency division capabilities, usually at the expense of increased output-to-output skew.

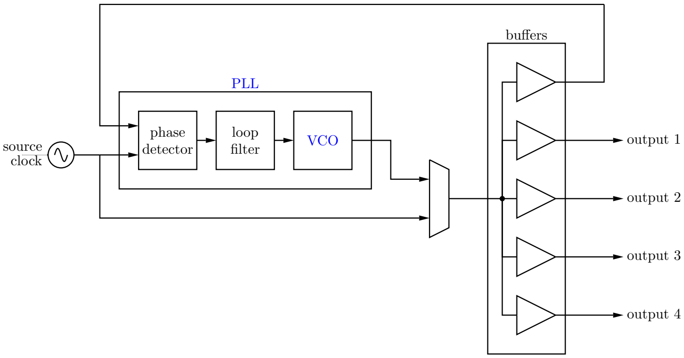


Figure 3.3: Basic scheme of a zero-delay buffer with selectable skew cancellation.

The second, more sophisticated class of clock replicators are typically called *zero-delay buffers* and make use of PLLs in order to cancel the skew between the clock outputs and the input, effectively providing a null propagation delay. One such device is portrayed in Fig. 3.3, and it essentially consists of a clock buffer and a PLL. The clock input is fed to the PLL phase detector, together with one of the buffer outputs, so that both are guaranteed to have the same phase after convergence. The skew between the other clock outputs and the input is then bounded by the output-to-output skew of the buffer. Moreover, the duty cycle of the output clocks is independent of the input signal, since it is provided by the PLL oscillator. More advanced devices can take full advantage of the PLL in order to include clock frequency multiplication capabilities, or adjustable delays for individual outputs.

Source-synchronous signaling

The system-synchronous scheme is the most obvious and straightforward way to define synchronization, but it can only be applied to clocks below a certain frequency. The reason behind this limitation can be seen by considering the process of data transfer between registers. The maximum clock frequency is determined by the critical path, i.e. the register-to-register path with the longest total propagation delay, where the data generated at its start on a given clock edge must be reliably captured at its end on the next edge. This critical path includes propagation through devices or gates but also the flight time through transmission lines. Hence, the physical size of the circuit imposes a limitation on the achievable clock frequency. This principle holds at all levels, be it inside an ASIC, at a PCB, or for the whole DAQ, with different frequency limits for each case. At the system level, the clock frequency limitation of source-synchronous clocking is estimated as 200 to 300 MHz.

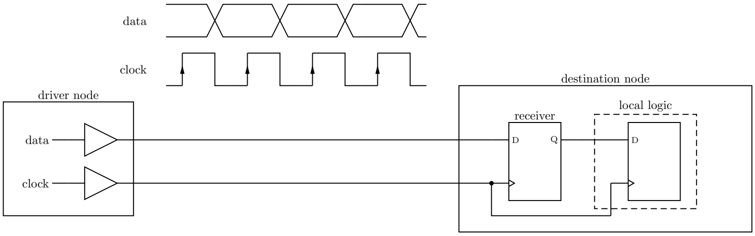


Figure 3.4: Scheme of a source-synchronous clock and data bus.

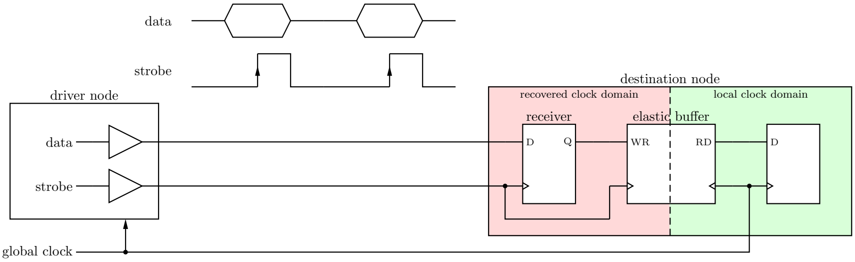


Figure 3.5: Asynchronous source-synchronous data bus with a clock domain crossing at the receiver using a dual-port FIFO as an elastic buffer.

Pushing the operating frequency (and thus the throughput of digital transmission lines) beyond this limit requires removing the dependence on overall circuit size by eliminating flight times from the timing equations. In order to achieve that, the phase alignment of clocks feeding all nodes needs to be sacrificed; instead, clock edges need to be shifted according to the data propagation delay so that both arrive at the same time, effectively canceling the influence of flight time. *Source-synchronous* timing schemes realize that goal by generating the timing signals and the data in the same device, and transmitting them together, so that both perceive approximately the same flight time. The situation is depicted in Fig. 3.4, where the clock edges are centered within the valid data windows for optimal sampling. Distributing clock and data along parallel PCB traces or cables with matched length, e.g. composite cables, ensures that the relative timing between both is maintained at the receiver.

In settings when data transmission is not continuous, such as in asynchronous buses, the timing signals do not need to be full clocks but may be *strobe* signals consisting of isolated edges that indicate the validity of the data coupled to them, as shown in Fig. 3.5. The source-synchronous scheme is then limited to the end nodes of data buses, and a separate clock tree needs to be implemented in order to distribute the clock and drive the rest of the circuitry.

In this asynchronous situation, two separate clock domains appear at the receiver: one for the local logic, and one for the clock or strobes coming from the transmitter. Even if they have the same clock frequency, their phase may be different. Direct coupling between the receiver and the local frequency may introduce metastability errors depending on the phase shift between clock domains; hence, a synchronization stage is usually introduced in between. One possibility is to add a *synchronizer chain*, i.e. a series of two or more consecutive flip-flops driven by the local clock; this technique removes the propagation of metastability errors almost completely, but may result in data corruption. A better alternative is an *elastic buffer*, i.e. a small double-port FIFO queue that is written with the incoming clock and read with the local clock. If both clocks are syntonized, then the queue never fills or empties completely, but stays at an intermediate fill level and data words are correctly transferred between domains.

Self-synchronous signaling

In theory, source-synchronous timing carries no frequency limit, because providing identical paths for clock and data signals ensures that they remain perfectly aligned. In practice, however, it is impossible to implement perfectly matched paths due to physical imperfections that enforce propagation delay differences either in the transmission lines or in the output and input buffers. These unavoidable skews imply a frequency limitation around 1 GHz to 2 GHz for this signaling technique ([91], p. 11-1–11-30).

The only way to surpass this limitation and avoid the impact of skew between different lines is to remove the possibility of there ever being a skew in the first place by using a single path for the transmission of both clock and data. The principle behind *self-synchronous* clocking is to merge clock and data together into a single waveform containing the data as well as enough information to be able to time the data reception but also recompose the source clock. The receiver is then responsible for extracting the embedded clock from the composite signal. This technique is thus halfway between baseband and modulated data transmission.

Notice that this signaling scheme forces the use of serial transmission in a single data wire. Because the transmission of data tends to be logically organized into m-bit words instead of an homogeneous bit stream, two different clock signals have to be considered in this serial case: the *bit clock* and the *word clock*. The former corresponds to the timing of single bits along the line whereas the edges of the latter indicate the start of transmission of logical data words, i.e. they determine whether bits in the data stream belongs to one word or another. The relationship between both clocks and the data signal is shown in Fig. 3.6: the word clock is a frequency-divided, edge-aligned version of the bit clock with frequency ratio m and edges occurring precisely at the first bits of new words. The clock signal being used for system timing is the word clock, since the bit clock frequency is typically

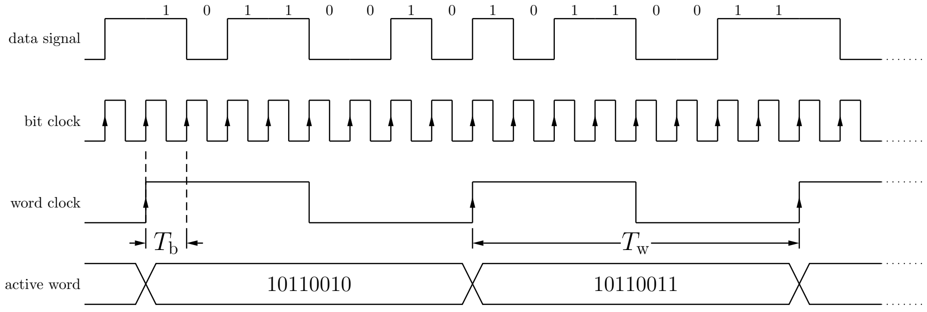


Figure 3.6: Relationship between the data signal and the bit and word clocks, with respective periods T_b and T_w, for words of $m = 8$ bits.

too high (above 1 GHz). The period T_b of the bit clock is usually called *bit period* or *unit interval* (UI).

Clocks embedded in self-synchronous signals are reconstructed using a PLL-based *Clock Recovery Unit* (CRU), which is usually seeded by an external reference clock with a frequency that is rationally related to the bit or word clock frequency. Only the nominal frequency of the reference clock is important, i.e. it need not be exactly syntonized to the embedded clock as long as the actual value is within the oscillator's tolerance. The PLL is responsible for extracting the bit clock waveform from the edges in the incoming data signal.

Serial clock recovery is only possible if a minimum rate of signal transitions is guaranteed on the data link, in order to keep the CRU PLL locked. Hence, special physical-level coding must be used in order to force transitions even in the case of long strings of equal symbols. There are two main approaches for this:

- Using a *scrambler* at the transmitter and a matching *descrambler* at the receiver. These are linear feedback shift register (LFSR)-based digital filters whose function is to transform the original bit stream into an equivalent one without long sequences of zeros or ones only ([92], pp. 416–418). While this technique adds no overhead to the transmission, i.e. it does not increase the number of bits to be sent, this comes at a cost: since the capacity of the output is the same as the input (because no redundancy is added), for every possible output there must be a corresponding input that generates it. Hence, there is a worst-case chance that the unwanted constant sequences appear at the output anyway. Scramblers are thus designed such that the probability of these unwanted sequences is small *under normal conditions*, i.e. taking into account the expected properties of the input bit stream, such as specific bit patterns in frame headers.

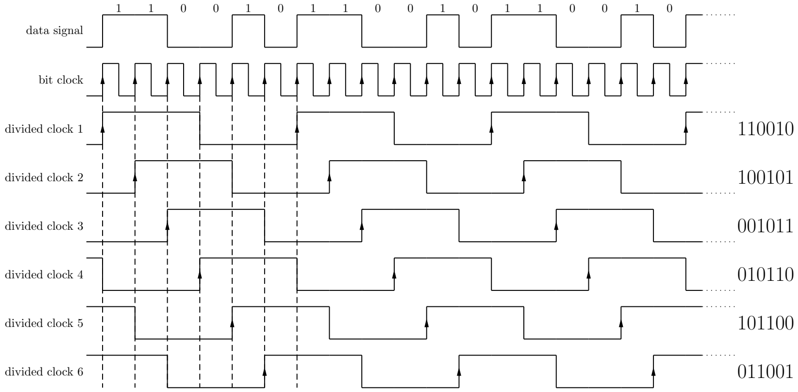


Figure 3.7: Possible divided versions of the bit clock for $m = 6$ and resulting deserialized data words.

- Using an isochronous line code at the physical level that guarantees a minimum transition rate at the cost of some redundancy. Usually, such line codes are also designed to produce DC-balanced waveforms by forcing the same mean amount of zeros and ones on the channel, hence permitting the electrical isolation between transmitter and receiver. The earliest example of this technique is the *Manchester code* ([93], pp. 274–275), where data bits are encoded as positive and negative edges, yielding at least one edge per bit at the expense of greatly increased bandwidth. The most commonly used alternative nowadays is the 8B/10B code [94] where data bytes are encoded as sequences of 10 physical bits, guaranteeing a transition every 5 bit periods at most; in this case, only a 20 % extra bandwidth is required. Other line codes similar to 8B/10B are described and evaluated in [95].

Once the bit clock is recovered, it may be used to correctly sample and decode the data embedded in the incoming signal. The word clock is then obtained by dividing the frequency of the bit clock by m. Notice that this division may produce m different versions of the word clock, all of them uniformly spaced with a phase difference of $2\pi/m$ between them, as shown in Fig. 3.7. Selection of the correct word clock requires interpretation of the decoded bit stream in order to identify the boundaries between data words, and is thus closely related to the line code employed in the data transmission. The most widely employed technique is the use of *self-synchronizing codes* [96], i.e. line codes that contain specific code words that can never appear as the result of concatenating substrings of any two valid code words; hence, if the synchronizing code word is found in the bit stream, then its boundaries automatically provide the alignment for the word clock, assuming no bit errors. For instance, in 8B/10B encoding, the *comma symbol* K.28.5 is encoded as either 0011111010 or 1100000101 and these bit strings can never appear as the result of two consecutive data words. Comma symbols included in frame headers can thus be used to achieve word synchronization using a simple sequence detector.

Special transmitter and receiver units are employed for the implementation of self-synchronous serial links that contain the necessary circuitry for all subfunctions, often merged into *transceiver* devices that contain matched transmitter-receiver pairs for the deployment of full-duplex links. Standalone chipsets are commercially available, either designed for specific serial communication standards [97–99] or supporting the implementation of proprietary link protocols [100, 101]. Additionally, since the start of the 2000s and following the trends described in §2.2.1, FPGAs from the main vendors have increasingly added hardware support for serial links in the form of transceiver hard IPs like the GXB on Altera devices [91] and the GTP family on Xilinx devices [102].

Fig. 3.8 shows the block diagram for generic high-speed transmitters and receivers embedded in FPGAs, although the contents of standalone transceivers are similar. The transmitter (top half) is the simpler part and may contain the following blocks, not necessarily in this order:

- A clock adaptation stage, like an elastic buffer, in order to cross from the local clock domain into the transmitter clock domain if they differ.
- A line coder block that adapts data words into a pattern where the embedded clock can be easily recovered. In general this may be a scrambler unit, however FPGA transceivers usually include 8B/10B encoders that transform 8-bit data words into 10-bit code words.
- A *serializer* that transforms parallel words into a serial bit stream at bit clock rate.
- An output driver that buffers the bit stream into the physical link at the appropriate electrical signaling standard.

Only the two last functions are necessary unconditionally; the other blocks may be absent in a given transmitter, or they may sometimes be bypassed by configuring the device.

The receiver (bottom half) is more complex, due to the need to regenerate the clock embedded in the data signal, and typically contains the following blocks, most of which merely undo the process carried out by the transmitter:

- A *Clock and Data Recovery* (CDR) unit, consisting of the PLL-based CRU and the sampling of the incoming data signal. This block outputs the extracted data bit stream as well as the embedded serial and parallel clocks. The serial clock is always equal to the bit clock, whereas the parallel clock has

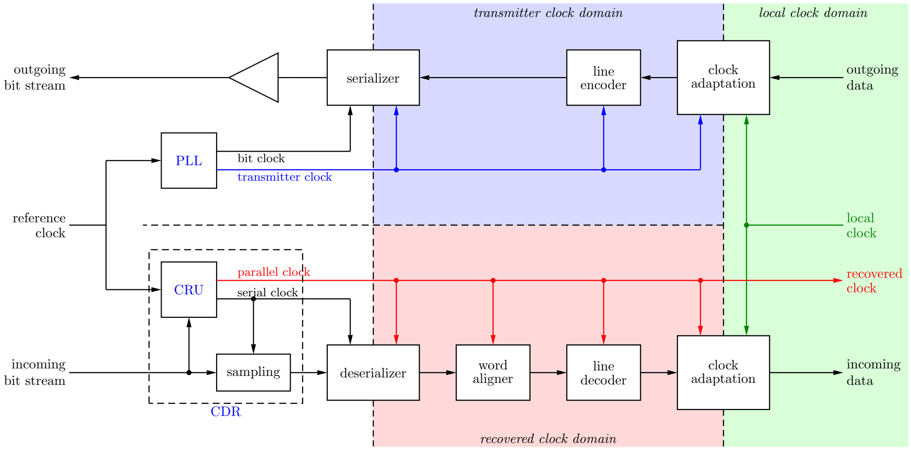


Figure 3.8: Block diagram of a generic high-speed transceiver on FPGA, with the transmitter on top and the receiver at the bottom, and different PCS clock domains highlighted.

the same frequency as the word clock but may be phase-shifted, depending on the implementation.

- A *deserializer* that transforms the bit stream back into parallel bit blocks at a rate equal to the word clock.
- A *word aligner* unit that is responsible for finding the correct boundary between data words, either by detection of specific self-synchronizing sequences or with assistance of user logic. This block outputs parallel code words as input to the transmitter.
- A line decoder, usually an 8B/10B physical layer decoder that transforms 10-bit code words back into 8-bit data words.
- Finally, a clock domain crossing interface from the recovered clock domain into the local clock domain, like an elastic FIFO buffer.

Again, it may be possible to bypass some of these blocks depending on implementation.

In the description of transceivers, two separate sections are usually distinguished: the *Physical Medium Attachment* (PMA) and the *Physical Coding Sublayer* (PCS), corresponding to the circuitry that works with serial and parallel clocks, respectively. In the diagram of Fig. 3.8, both sections are separated by the serializer and deserializer.

3.1.2 Latency and jitter

In general, the *latency* of a signal processing element is defined as the time difference between the input signal entering the element and the corresponding result appearing at the output. For digital signals such as clocks, the timing reference used to define this time difference is given by the corresponding active edges. In this regard, clock signals may be considered as a particular case of baseband digital data signals with a fixed data pattern (010101...).

In the particular, simplest case of signal *transmission*, be it a data link or a clock distribution element, latency coincides with the propagation delay of the given element. However, it is preferable to use the concept of latency because it also applies to self-synchronous signaling links, where the clock signal is embedded in the data stream at the transmitter and then reconstructed at the receiver; hence the link, viewed as a clock distribution element, is not a pure transmission element but rather includes processing stages.

Sources of latency variation

It is clear that the latency L of a clock distribution element is directly related to the phase difference between its input and output. In particular, for a clock with period T_{clk}, the linear equation

$$\varphi_{\text{out}} = \varphi_{\text{in}} + 2\pi \frac{L \bmod T_{\text{clk}}}{T_{\text{clk}}} \tag{3.1}$$

holds, where φ_{in} and φ_{out} denote the clock signal phase at the input and the output of the distribution element, respectively.

In a real situation, the phase of a given clock signal of fixed frequency does not remain completely constant but is rather subject to deviations. Besides the instantaneous signal degradation induced by electronic noise, (3.1) reveals that phase deviations may be attributed either to variations in distribution latency L or to fluctuations in the phase φ_{in} of the source clock signal itself. If a fixed source phase is assumed, or equivalently, if only the phase difference between source and transmitted clocks is considered, then the deviations in clock phase are determined by variations in the latency of distribution elements; in particular, the phase mismatch between clock signals derived from the same source is given by the latency mismatch between their distribution paths.

The contributions to the deviation of latency or phase between various clock signals may be split into four main categories depending on their expected variation rate. These are summarized in Table 3.1.

Fixed latency and phase mismatches are usually considered constant and are mainly determined by the difference in propagation delay between correspond-

Category	Variation rate	Typical sources
Fixed	Never for a fixed setup	Cable/fiber length mismatch Connectors PCB trace length mismatch Delay of active devices Static FPGA/ASIC routing
Static	Power cycles and resets	Deserializer clock division Elastic buffer delay
Drift	Long-term, slow	Environmental conditions (e.g. temperature, pressure...) Component aging
Jitter	Fast, random Fast, deterministic	Thermal noise Power supply ripple Electromagnetic interference

Table 3.1: Classification of the sources of variation and mismatch between link latency or clock phase. Adapted from [103].

ing elements in identical distribution paths. Once a particular physical setup has been deployed (including board allocation in racks and cabling), this delay remains constant; hence, phase mismatches caused by it can be obtained and compensated by calibration, and this calibration measurement only needs to be performed once. However, this calibration step may need to be repeated whenever there is a change of circuit boards, cable reconnection, or firmware upgrade. Even the mere flexing of cables can change delays enough to require recalibration [78].

The second category includes *static* latency terms that may take a number of possible values from within a range or set of allowable delays, however the particular value is fixed on convergence of a transient phase after power-up or reset. This is typically related to circuits that manipulate clock frequency or cross clock domains. The two most important examples are clock dividers where the divided clock can take m different forms (where m is the division factor), or elastic FIFO buffers crossing same-frequency clock domains whose latency contains a fixed but undetermined integer number of clock cycles.[2] Equalization of these latency terms through calibration is possible, but the procedure is required every time that the system is powered up or a DAQ subsystem is reset. As will be seen later in detail, some static latency terms can be determined without external calibration, like the ones introduced by word clock dividers in self-synchronous receivers, but others may require calibration, e.g. delay in some elastic buffers.

Deviations in the third category are usually smaller and often disregarded, but they are also slow enough that it is possible to correct them through calibration

[2] Note that the latter example affects latency but not phase.

in situations that require fine tuning. For instance, element delay drift caused by component aging may be compensated by recalibrating often enough, taking into account the expected rate of change. Drifts caused by variations in temperature, be it from environmental conditions or changes in power dissipation, can be partially corrected by measuring temperature locally and assuming a first-order dependence [104].

Clock jitter

The last category described in Table 3.1 comprises small, high frequency deviations in clock phase whose variation rate is comparable to clock frequency, and can be identified as a form of noise. Its aggregated contribution is called *clock jitter*. Formally, jitter is the deviation in a digital signal's transitions from their ideal positions ([105], pp. 4-1–4-10). This applies both to clocks and, more generally, to baseband digital signals.

Clock jitter is the main parameter establishing the quality of a clock signal, and describes a measure of noise in the time domain.[3] The rigorous mathematical definition of phase deviation as found in [106] is impractical, so a number of different jitter metrics are usually employed depending on the noise effect under study.[4] The time deviation of edges from their expected position, given one initial reference edge, is called *absolute jitter*, whereas the deviation of edge position between consecutive clock cycles is called *period jitter*; the former is used e.g. to describe the resolution of ADCs, while the latter is a fundamental component in timing margins for synchronous logic [107]. Both are depicted in Fig. 3.9.

In addition, jitter can be further split into two components with completely different behavior. On one hand, *random jitter* is caused by random noise sources in the clocking circuitry such as thermal noise, and its distribution is always approximated as normal with null mean by virtue of the Central Limit Theorem, as contributions to jitter can be assumed to be additive; jitter in raw oscillator outputs can also be shown to be distributed normally by analogy with random walk processes [106]. On the other hand, *deterministic jitter* follows a predictable pattern and has very specific sources, typically switching noise from power supplies but also crosstalk, systematic non-linearity effects, or inter-symbol interference in data streams. *Total jitter* is the sum of both and its joint distribution is the convolution of their individual distributions. For simplicity, a bimodal model is usually assumed where the distribution of deterministic jitter is concentrated on its two

[3] In the frequency domain, clock noise is typically described by the *phase noise* $\mathcal{L}(f)$, equal to the normalized spectral power density of the clock signal at frequency $f_{\text{clk}} + f$ [106]. Jitter variance can be obtained by integrating $\mathcal{L}(f)$.

[4] The terminology for these metrics is not standard and sometimes contradictory. The definitions found in [107] will be adopted here.

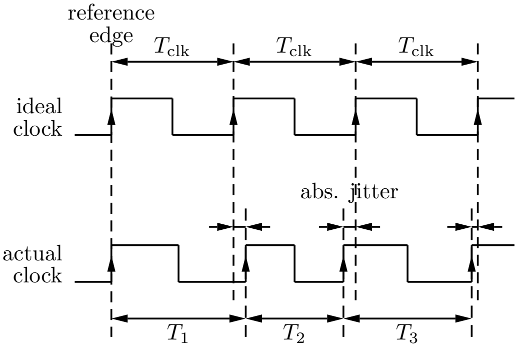


Figure 3.9: Graphical description of the most important types of jitter. For the i-th edge, absolute jitter is defined as $T_{\text{clk}} - T_i$ and period jitter as $T_i - T_{i-1}$.

peak-to-peak limit values and, therefore, the distribution of total jitter is the sum of two normals.

Deterministic jitter is always bounded and given by its peak-to-peak value D. However, random jitter, described by its rms value R, is unbounded and can potentially reach any value. In order to derive a peak-to-peak value T for total jitter, a variation bound is established by considering a reference maximum error rate, where an "error" corresponds to the random jitter momentarily exceeding the given bound. Total jitter is then given by the equation

$$T = D + 2NR \tag{3.2}$$

where N is a parameter related to the reference error rate; the most typical value is $N = 7$, corresponding to a bit error rate of 10^{-12}.[5]

Jitter levels in clock signals may be reduced by means of *jitter cleaner* devices. These are usually implemented as PLLs or DLLs that regenerate the signal with a local oscillator and possess a low loop bandwidth in order to filter the fast frequency variations induced by jitter; as a result, they also tend to feature long convergence times.

Deterministic latency

In many situations, a crucial specification that needs to be fulfilled by data links between different hierarchy levels in a DAQ is that of *deterministic latency* in order to guarantee synchronicity i.e. correct order of processing between separate nodes, for instance in response to low level trigger signals with constrained response windows. As in other situations regarding DAQ, the term is employed somewhat

[5] The exact meaning of parameter N and its relationship with the error rate will be explained in a different context in §6.3.3 and Table 6.9.

loosely and different authors sometimes use it to refer to two different concepts. An effort will be made here to establish rigorous definitions and naming conventions.

A data link, or an element in a data or clock distribution path, is termed as *fixed latency* or *constant latency* if it always presents the same latency value, or its variation is constrained to a comparatively small amount, not just during a given power cycle but also between different power and/or reset cycles, i.e. after resetting it or turning it off and on again. More specifically, this means that its latency is fixed and, in particular, contains no static non-fixed terms, according to the classification given in Table 3.1. Small variations due to temperature drift may be allowed or not depending on the context.

On the other hand, a path has *deterministic latency* if its latency is either fixed or contains static terms whose exact value may be obtained at runtime. Equivalently, the latency difference between two identical paths with deterministic latency may always be precisely known without resorting to complete calibrations that yield the total latency values, because fixed values are cancelled and static values are known. Of course, fixed latency paths automatically satisfy the conditions for deterministic latency. Moreover, a unidirectional path with bounded deterministic latency can always be extended to a fixed latency path by adding a programmable delay element and programming it so as to compensate for the static latency variation.

Note that neither of those definitions requires knowledge of the exact latency or phase values. The first concept only specifies that it is constant, while the second one allows variation as long as the value of the variation is known. The confusion in terms stems mainly from the fact that some authors use the term *deterministic* to refer to *constant* latency - incorrectly, since the actual definition of *determinism* is the lack of randomness in future system states but not necessarily in the initial state, which may be random or unknown. For the definition of deterministic latency employed here, the strictly stronger condition is imposed that the latency variation is known.

3.1.3 Time synchronization

In a DAQ system, or in general distributed real-time systems, subevents are detected in physically separate nodes and the time relationship between them (order and coincidence) needs to be resolved in order to identify distributed events. This is accomplished by assigning a time value, or *timestamp*, to each such occurrence and then comparing said values. In order for these timestamps to be comparable and yield a valid time difference in the metric of physical time,[6] a common time reference needs to be established, as well as a time synchronization mecha-

[6] Relativistic considerations about the lack of a fixed universal time metric will be disregarded in this discussion because of the comparatively small extent of the DAQ systems under study.

nism that allows each node that assigns time values to access the common time reference with a certain resolution.

Time reference of nodes

For any node i that is capable of assigning timestamps to events, the *local time reference* is a function τ_i that converts physical time into the timescale being used by the node to assign specific time values, i.e. $\tau_i\left(t\right)$ is the time value perceived by node i at physical time t. The essential restriction on local time references is that they must be *chronoscopic*, i.e. that τ_i must be a monotonically increasing function in order to allow the measurement of time intervals at any moment [109]; backward jumps in $\tau_i\left(t\right)$ may cause, for example, the measure of negative time intervals between two sequential events.

The local time reference at a timestamping node i is usually derived from a digital *timestamp counter* that operates at a frequency f_{clk} and increases its value by one each clock cycle, thus providing a coarse time reference with granularity $T_{\text{clk}} = 1/f_{\text{clk}}$. More accurately, let $C_i\left(t\right)$ be the value of the timestamp counter at physical time t. Then

$$\tau_i\left(t\right) = C_i\left(t\right) \cdot T_{\text{clk}} \tag{3.3}$$

whereas the value held by the timestamp counter varies as

$$C_i\left(t\right) = \left\lfloor \frac{t - t_0}{T_{\text{clk}}} \right\rfloor = C_0 + \left\lfloor \frac{t}{T_{\text{clk}}} + \frac{\phi_i}{2\pi} \right\rfloor \tag{3.4}$$

where $\lfloor x \rfloor$ denotes the integer part of x, t_0 is the physical time corresponding to the active clock edge when the counter value became zero, C_0 is an integer and ϕ_i is equal to the phase of the timestamp counter clock at time $t = 0$.

Note that τ_i as defined is a discontinuous function but it is still monotonic (although not strictly monotonic) and so satisfies the condition of chronoscopy. If the instantaneous phase $\phi_{\text{clk}}\left(t\right)$ of the timestamp counter clock is available, then (3.3) can be extended to a continuous time reference as

$$\tau_i\left(t\right) = C_i\left(t\right) \cdot T_{\text{clk}} + \frac{\left(\phi_{\text{clk}}\left(t\right) - \phi_i\right) \bmod 2\pi}{2\pi} \tag{3.5}$$

i.e. by taking the continuous clock phase into account.

In a real setting, the situation is slightly different in that the timestamp counter has a finite number of bits B and so the actual local time reference is given by the counter value

$$C_i\left(t\right) = C_0 + \left\lfloor \frac{t}{T_{\text{clk}}} + \frac{\phi_i}{2\pi} \right\rfloor \bmod 2^B. \tag{3.6}$$

This simplification cannot be applied in other situations such as the synchronization of GPS nodes [108].

Strictly speaking, this reference is not chronoscopic because of the jump from $2^B - 1$ to 0 as the counter rolls over. However, this effect can be negated by choosing a sufficiently high counter length so that the rollover period $2^B \cdot T_{\mathrm{clk}}$ is long enough that there is no possible confusion between events taking place with such time difference, and using two's complement arithmetic for the computation of time differences so the counter rollover does not affect them.

In a system with several timestamping nodes, the ideal situation corresponds to all of their local time references being identical so that the timestamps they assign can be compared without error. However, it is impossible to guarantee perfectly accurate access to a common time reference in practice. The difference between the local time references from two different nodes is called the *synchronization error* between them and should be ideally equal to zero.

Two different processes can be identified for the synchronization of local time references. *External synchronization* amounts to the alignment of local time references with actual, physical time as provided by an accurate external source, i.e. it consists in keeping

$$\left|\tau_i\left(t\right) - t\right| \leq \varepsilon \tag{3.7}$$

at all nodes i for some accuracy bound ε. By contrast, *internal synchronization* refers to the construction of a common time base for all nodes in the system by ensuring that

$$\left|\tau_i\left(t\right) - \tau_j\left(t\right)\right| \leq \varepsilon \tag{3.8}$$

for all nodes i, j independently of external or absolute time scales. In both cases, a common time reference $\tau\left(t\right)$ is established that all nodes can access and use to assign approximate time values with a synchronization error bounded by ε, the difference being that $\tau\left(t\right)$ corresponds to physical time in the former case, but not necessarily in the latter.

Recall that in an experimental physics setting, events are independent and hence absolute time scales are not necessary, but only relative time scales corresponding to one single event, i.e. only time differences matter.[7] Therefore, external synchronization is usually not important with respect to correct date and time, but it is with respect to the rate of change of the common time reference; in other words, the time scales of physical time and the common reference need to be matched even if they are not aligned. Hence, in this dissertation, the attention will be turned to internal synchronization processes but with the additional restriction that time intervals can be accurately measured locally, i.e. that

$$\left|\tau\left(t_0 + \Delta t\right) - \tau\left(t_0\right) - \Delta t\right| \leq \varepsilon \tag{3.9}$$

for time intervals Δt bounded by a maximum event duration.

[7] This also applies to settings with very long acquisition times such as astrophysics experiments that measure annual variations of signals; in that case, events may be thought of as lasting for several years with a very large amount of subevents represented by individual data points.

In order to clarify the implications of condition (3.9), consider a time reference given by a timestamp counter with nominal frequency f_{clk} but an unknown *clock drift* δ, defined as the relative deviation of frequency from its nominal value

$$\delta = \frac{f'_{\text{clk}} - f_{\text{clk}}}{f'_{\text{clk}}} \tag{3.10}$$

where f'_{clk} is the actual clock frequency. If the clock drift stays constant during the time interval $[t_1, t_2]$, then the absolute error in the measurement of the interval length is

$$(\tau(t_2) - \tau(t_1)) - (t_2 - t_1) = \delta \cdot (t_2 - t_1)\,. \tag{3.11}$$

It follows that (3.9) is bounded by $\delta \cdot \Delta t$, where δ is the maximum clock drift. Hence, the restriction (3.9) amounts to a specification on maximum clock drift for accurate time interval measurement, and does not depend directly on the synchronization method but rather on the quality of the clocks in the system.

The goal of a time synchronization procedure is to maintain a common time reference with synchronization errors as small as possible. In a real-time system, a *deterministic* clock synchronization mechanism is typically required that has a known bound on the error; in this case, the *synchronization accuracy* is defined as the maximum possible synchronization error ε between nodes in the system [109]. *Probabilistic* clock synchronization methods are also possible, where the possible synchronization errors are unbound but restricted to a small interval with high probability [110]; in this case, the performance of the synchronization method is described by the standard deviation or any other statistic of the distribution of synchronization errors.

Time reference of signals

Besides the approximate common time reference being maintained by each timestamping node, a time reference can be attached to each signal that describes the physical time corresponding to the occurrence of said signal at its source. More exactly, the time reference of a signal or derived datum is the common time value (i.e. referred to the common time reference) to which each state of the signal corresponds, and takes the form of a time value associated to every single signal value:

- For a single scalar value, or a reduced group of related scalar values, such as the summarized description of a pulse or subevent using a few parameters (e.g. charge, position...), the corresponding time reference is a single timestamp T that is identified with the point in time when the subevent took place.

- For a stream $x[n]$ of signal samples, e.g. in the case of a waveform after free-running sampling, the time reference consists of a sequence of separate timestamps $T[n]$, one for every single sample. If the sampling frequency f_s is assumed to be constant, as is usually the case, then the time reference may be reduced to the form (T_0, T_s) given by a single timestamp $T[0] = T_0$ for just one sample and the sampling period $T_\mathrm{s} = 1/f_\mathrm{s}$, since the timestamps for all the other samples can be obtained as

$$T[n] = T_0 + nT_\mathrm{s}. \tag{3.12}$$

The actual sampling period may vary due to jitter in the sampling clock, hence an absolutely rigorous time reference should include the timestamps for all samples individually; however, this variation is usually interpreted as added noise in the sampled signal instead, as will be pointed out in §4.3.3.

- For a continuous-time signal $x(t)$, the accompanying time reference is a function $T(t)$ that specifies the time values corresponding to each position of $x(t)$, where the parameter t is specified in the common time reference. Of course, it may as well be $T(t) \equiv t$ if the signal is parameterized by the reference time scale. In usual settings, the time scales of continuous-time signals are not stretched or shrunk and hence turn out to be of the form $T(t) = t + t_\mathrm{d}$ for some constant t_d that represents the relative delay of signal $x(t)$ from its source to the observation point.[8] It follows that the time reference of a continuous-time signal may be described by a single delay value t_d, and that the misalignment between two continuous-time signals in a DAQ is given by a relative delay difference.

Thus, in any case, the time reference of a signal or datum is completely determined by a single value (disregarding the known sampling frequency): a timestamp for digitized values and discrete-time signals, or a delay for continuous-time signals.

Synchronization levels

The information provided by each sensor in the DAQ is propagated upwards in the hierarchy levels from the physical detector to the event builder, in different forms along the path depending on the readout scheme of the particular detector. As outlined in Fig. 3.10, analog signals from the detectors are propagated to a number of timestamping nodes where they are sampled, i.e. converted from continuous-time to discrete-time, and their time references are fixed and assigned by way of a digital timestamp. The two points in this path that are critical for time synchronization are the sampling point and the timestamping logic (where the timestamp is assigned). Note that these points are physically different, usually located in an ADC and FPGA respectively, so that there is a nonzero latency between them. It will be assumed that the sampling is performed at frequency

[8] Signals may be discretized and then resampled, for instance when using analog memories [111], in which case $T(t) = T_0 + kt$ for a fixed frequency ratio k.

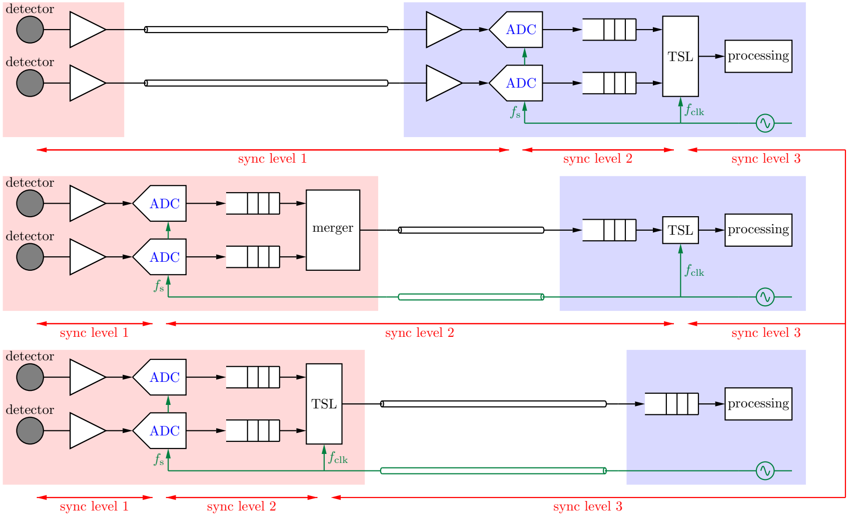


Figure 3.10: Path of signals and information from the detectors to timestamp assignment, split at the critical points that define different synchronization levels, in a system with subdetectors that use the various readout schemes described in Fig. 2.5. Front-end boards are shown in red, back-end boards in blue, and clock distribution in green; TSL stands for *timestamping logic*.

f_s and the timestamping logic is working at frequency f_{clk}. These frequencies are typically related by an integer proportionality factor for convenient ADC readout; in the most simple case, $f_s = f_{clk}$ holds.

These two points divide the signal path into three sequential sections, as shown in Fig. 3.10: from sensor to digitizer, from digitizer to timestamping logic, and from the timestamping logic to the end of the datapath. Correspondingly, the time synchronization process will be divided into three steps or levels: synchronization of signals as they arrive at the digitizers, at the timestamp assignment points, and finally synchronization between the timestamping logic in different nodes. These are labeled in Fig. 3.10 as levels 1, 2 and 3, and correspond roughly to delay calibration, digitizer synchronization, and timestamp synchronization, respectively. Each synchronization level assumes that the higher levels are already synchronized and refers to the common time reference provided by them.

In Fig. 3.10, these synchronization levels are shown for a DAQ system with different subdetectors implementing the various readout schemes described in Fig. 2.5: remote sampling for the top detector, and local sampling for the other two, with the timestamping operation taking place in the back-end and the front-end, respectively; both can correspond to either triggerless readout or local sampling with L1 trigger. Notice that each case corresponds to the implementation of a different synchronization level over the transmission lines connecting the front-end to the back-end. The first case only requires delay calibration, whereas the other two cases additionally require transmission of the clock to the front-end boards and synchronization between all front-end clocks.

At the lowest level, all arriving signals are continuous-time and their time references are completely determined by a delay value for each one, and so the process of time synchronization amounts to the calibration and correction of the relative delays between all of those signals. Note that it is necessary to do so between any two signals in the DAQ whose timing relationship needs to be resolved at some point in the hierarchy, and not just between signals being digitized at the same node, i.e. if the time reference for digitizers in different nodes is synchronized but analog signals arrive at them with different delays, then they will be misaligned anyway.

The second level corresponds to the time synchronization of the digitizers, assuming that the timestamp assignment logic for all nodes has a common reference. Here, the latency between the digitizer and the timestamping logic has to be considered. At the very least, this includes the propagation path from ADC to FPGA, which is not necessarily deterministic.[9] Other potential components are:

- The signal processing path in the FPGA and the timestamping algorithm, if present. Their latency is an integer number of f_{clk} clock cycles, typically fixed, but care must be taken when its execution time is not constant.
- The transmission and reception latency between ADC and FPGA depending on the signaling mode, e.g. in the case of high-speed ADCs with serial outputs that must be decoded with gigabit transceivers, such as [112].
- Any possible phase changes between clocks. In particular, if $f_{\mathrm{s}} = f_{\mathrm{clk}}$ and there is a phase change at some point, the clock domain crossing introduces an added latency equivalent to the phase difference as per (3.1), given by the time delay between respective active edges.

The effect of the phase of the sampling clock is of special interest. Specifically, consider a DAQ system where all sampling clocks in all digitizing nodes are syntonized. If the digitizers are not sampling simultaneously with the same phase, and this is not taken into account, then delay errors are introduced whose value is equal to the phase difference between the sampling clocks as related by (3.1). A common way to deal with this issue is to calibrate the clock distribution network so that sampling phases are indeed equal (for instance, see most of the examples described in §3.4.2). In this dissertation, a different path will be taken: the latency between the sampler and the timestamp assigner will be considered instead as an added delay term, from the active sampling clock edge to the active timestamp-

[9] This issue will be treated with additional detail in §6.1.1.

ing clock edge, and the problem of phase mismatch will be pushed back to the timestamping clock and the synchronization between timestamping blocks, thus establishing the timestamping point instead of the digitizing point as the common system-wide time reference.

The final level involves the synchronization of the timestamping logic blocks themselves, in the sense that any samples arriving at timestamping blocks in different nodes at exactly the same time should be assigned digital timestamps that can be recognized later as equivalent. This corresponds to the internal synchronization of the local time references in each node, provided by local timestamp counters; their actual value and its correlation with external date and time are not important, as long as it permits the discrimination between different events. From this point upward, the value of the timestamp counters is taken as the common time reference for the whole DAQ.

If the timestamp counter clocks in separate nodes are syntonized and in phase, then it is ideally possible to guarantee that they all hold the same values at any time. One possible way to achieve that goal is to deploy a *reset tree* alongside the clock tree that sends a common reset signal to all timestamp counters at the same time, so that they are all reset to zero simultaneously. However, it is impossible to maintain identical timestamp counter values at all times when a phase difference exists between two of them; at best, they will exhibit equal and different values alternatively during each clock cycle. The same applies whenever timestamp counters are not syntonized, as clock drift will tend to accumulate and increase the difference in counter values over time if left unregulated: if counters run at frequencies f_{clk} and f'_{clk}, then the fastest counter will have counted an extra $N \cdot \left|T'_{\text{clk}} - T_{\text{clk}}\right| / T_{\text{clk}}$ counts after N cycles. In this case, periodic timestamp resets using a reset tree may be used to guarantee that synchronization errors are bounded by

$$\varepsilon \leq 2\delta N T_{\text{clk}} = 2\delta T_{\text{reset}} \tag{3.13}$$

where δ is the maximum clock frequency drift from the nominal f_{clk} and resets are issued every T_{reset}, corresponding to N cycles.

Finally, note that the effect of this final timestamp synchronization level does not apply to all signals in the DAQ, but only to those that are actually tagged with a timestamp that will be used for comparison at a higher hierarchy level. In the case of signals for which only the timing relations with other signals arriving at the same node are important, these relations can be obtained locally and only the result (order and/or coincidence) is stored and forwarded, dropping their time of occurrence.

Synchronization schemes

A time synchronization scheme is needed that ensures that all timestamps assigned to individual pieces of detector data are comparable to resolve the relations of order and coincidence between them. There are several mechanisms to carry out this process, and one can distinguish three different logical levels where the synchronization of time references may be enforced, in increasing order of abstraction:

- *At the physical level:* This case corresponds to matching delays and clock phases between different signals in the whole DAQ. More specifically, in top-down order, this amounts to establishing syntonization across the DAQ and then aligning the phases of all timestamp counters, the phases of all digitizers, and finally matching the equivalent delays of all analog continuous-time signals from the detectors to the digitizers and then to the timestamping blocks. Basically, this procedure corresponds to the physical adjustment of propagation delays in the system, and is typically carried out by calibration, including the delays of analog detector signals and the balancing of the clock distribution tree in order to guarantee phase alignment. Moreover, the values of the timestamp counters from different nodes need to be perfectly matched with each other, e.g. by deploying a balanced reset tree.

- *At the logical level:* In this case, it is possible to have different phases for the timestamp or sampling clocks, or to have different delays for the analog lines or the digitizer-to-timestamp path. However, the values of these phase differences or delays are obtained beforehand, either through calibration or using any other measurement method, and then taken into account for the automatic, online correction of assigned timestamps. The most simple example consists in measuring the propagation delay t_d for each analog channel using a calibration run, storing them in LUTs at the timestamping nodes, and then implementing a correction step in the timestamping logic that subtracts the stored delay for each signal. If one assumes that these propagation delays remain constant, then this step corresponds to correcting for the signal's time reference $\tau(t) = t + t_\mathrm{d}$ and assigning the time of occurrence at the detector instead of the time when the pulse is read at the timestamping node. Other examples involve the synchronization of the timestamp counter values through algorithms based on message passing instead of physical calibration; they will be described in §3.2. Where these techniques can be applied, their main advantages over physical synchronization are the reduction in hardware requirements (e.g. delay lines or reset trees) and the avoidance of time-consuming calibration processes.

- *At the statistical level:* One final possibility is to leave the correction of timestamps to the offline post-processing stage. In this case, delays or phases are not calibrated, and large amounts of event data are captured and stored.

Later, statistical analysis of the whole set of captured data is carried out in order to determine the actual timestamp correction values that need to be applied to each signal; timestamps are then corrected offline and used for the resolution of time relations. This corresponds to merging the calibration run with the actual measurement run. For instance, the method presented in [113] for the correction of coincidence detection in PET belongs to this class, as it relies on data from valid captured events for the estimation of calibration parameters. Another example is the timing distribution for the Belle II experiment [114].

This choice of scheme greatly simplifies the synchronization hardware design where it can be applied, but it carries an important drawback: it requires the capture and storage of a large amount of data that will be discarded later. Because coincidence relations cannot be resolved until the software stage, it is not possible to apply triggers based on coincidence and thus the trigger subsystem is less selective, increasing the necessary data bandwidth and storage resources. While this may be acceptable in cases like HEP experiments where one would want to store large amounts of data for each event anyway, the technique is not well suited to applications such as PET where the trigger conditions are simple enough and one wishes to limit the data bandwidth requirements as soon as possible in order to reduce the cost of the scanner.

These schemes do not need to be used in a pure manner for the whole DAQ, but the synchronization process may rather be based on a mixed scheme instead, where the synchronization of different subsets of the system is enforced at different levels.

3.2 Synchronization over data links

This section deals with methods for the establishment of internal time synchronization, i.e. the alignment of the local time references at different timestamping nodes by matching their respective timestamp counters. One way to achieve this alignment has already been presented: the deployment of a reset tree, or equivalently, a global resynchronization signal with balanced paths. This technique can guarantee an upper bound on synchronization error that depends on parameters such as the relative drift of the clocks (zero if syntonized), the reset period, and the proper latency balancing of the clock and reset distribution networks. However, this approach carries several drawbacks:

- It is a physical instead of logical synchronization scheme, and hence requires lengthy calibrations.

- It is incapable of correcting for random latency variations, namely drift in propagation delays in long networks for large-scale distributed DAQs.
- It requires dedicated physical resources, like a system-wide reset tree alongside the clock tree. This forbids its use in very sparsely distributed systems, as well as in wireless or highly mobile systems where latency-balanced transmission is impractical or impossible [115].
- It is does not provide a chronoscopic time reference, in the sense that each timestamp reset creates a disruption in the reference time line where the order and coincidence relations are broken, unless one can guarantee that the resets take place at very precise timestamp counter values.

A different approach is to implement synchronization algorithms that operate at the logical level, based on transmission of messages over the data links between the different nodes. As will be seen, the alignment of timestamp counters can be realized by sending timing messages between the timestamping nodes that contain the known timing information, i.e. the local counter values, and use it to correct the possible timestamp offsets between nodes. This correction can then be applied at the logical level but also at the physical level, if needed, by including programmable delay lines and phase shifters.

The main advantage of this approach is the lack of dedicated infrastructure for the distribution of reset signals, making it possible to synchronize the local time references in global or wireless networks; for instance, it is widely employed for the alignment of time and date of computers across the Internet using the Network Time Protocol (NTP) [116], or for the synchronization of Global Positioning System (GPS) nodes [117].

These methods rely on an existing infrastructure for the transfer of messages between nodes. In the context of DAQ systems, it can be assumed that the media for data communication are present anyway, so there is no need for dedicated hardware. Instead, the resource employed by these synchronization schemes is the link bandwidth taken up by timing messages. Therefore, one wishes to minimize the size of these messages and their rate of transmission in order to reduce their impact on system cost or performance.

In [115] and [118], the methods for synchronization through timing messages are divided into three categories, depending on the basic messaging step that they rely on: *sender-to-receiver* methods, *receiver-to-receiver* methods, and *one-way messaging* methods. The first one involves the exchange of timing messages between two nodes which then proceed to synchronize their time references to each other. The other two approaches are based on a master node broadcasting timing messages to a number of slave nodes which use them for synchronization, either on their own (third case) or with the assistance of each other by exchanging time-of-arrival information (second case). These two categories rely on multicast messaging and thus can only be used in multi-hop scenarios or networks with shared transmission media. In light of the shift towards point-to-point transmission in HEP DAQ systems that was justified in §2.2, this section will focus on sender-to-receiver methods.

Point-to-point message based timestamp synchronization has been a well-known and established approach for a long time, however its use in precisely synchronized DAQ systems has been restricted by the fact that the achievable resolution is limited by the uncertainty of message delivery times. To be more precise, Lundelius and Lynch prove in [119] that it is impossible to synchronize a set of n nodes in a way that guarantees a synchronization accuracy better than

$$\left(1 - \frac{1}{n}\right) \cdot \Delta t_{\mathrm{p}} \tag{3.14}$$

where Δt_{p} is the variation in the propagation delay of messages. Optimization efforts have thus focused on reducing this delay uncertainty by providing hardware assistance and controlled transmission media.

For several years, the most advanced step in this direction has been the Precision Time Protocol (PTP) [120], aimed at increasing synchronization accuracy from the $O(1\,\mathrm{ms})$ of NTP to better than $O(1\,\mu\mathrm{s})$ and, in fact, has been shown to consistently achieve accuracies better than 100 ns in controlled situations such as local LXI instrumentation networks [121, 122]. The best possible PTP resolution has been shown to be around 500 ps rms under ideal circumstances, using extremely precise oscillators and a very high resynchronization rate [123]. This bound is still unsatisfying for many modern data acquisition systems, where accuracies well below 1 ns are mandatory.

This section is centered around the synchronization method behind PTP-like protocols. The synchronization mechanism will be described, the effect of various inaccuracies on its resolution will be analyzed, it will be shown that raw PTP is incapable of guaranteeing sub-nanosecond accuracy, and extensions will be proposed in order to achieve that goal.

3.2.1 Pairwise synchronization

The analysis will focus on pairwise synchronization, i.e. on the establishment of a common time reference between two separate nodes, just by interchanging messages between them with no external agent. Descriptions of the basic step can be found in countless references, including the NTP and PTP specifications, and it receives several different names. In this text, it will be called *two-frame synchronization*, as it is based on the transmission of two timing frames between the nodes. The establishment of global synchronization on a larger network can

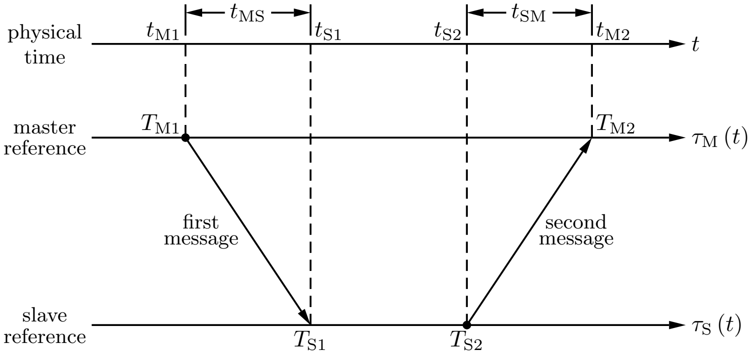


Figure 3.11: Outline of the two-frame synchronization method.

then be built around the pairwise step by defining a spanning tree and applying pairwise synchronization between any pair of nodes that form a branch in the tree.

Let us fix two nodes with timestamping capability whose local time references need to be synchronized. For simplicity, these two nodes will be called *master* and *slave*, even though both roles may be fully interchangeable. The master node is the one that initiates the synchronization procedure, and the goal is for the slave node to be able to adjust its local time reference $\tau_{\mathrm{S}}(t)$ so that it matches the master's reference $\tau_{\mathrm{M}}(t)$.

The basic step consists in the transmission of two timing messages, one from master to slave and then one from slave to master, as outlined in Fig. 3.11. In both cases, the respective nodes are responsible for timestamping the message's departure from the transmitter and its arrival to the receiver. The following notation will be used:

- Let t_{M1} be the physical time of the first message's departure from the master, and t_{S1} the physical time of its arrival to the slave.
- Let t_{S2} be the physical time of the second message's departure from the slave, and t_{M2} the physical time of its arrival to the master.
- Let t_{MS} and t_{SM} be the *travel time* or *flight time* for the first and second messages, respectively, i.e.

$$t_{\mathrm{MS}} = t_{\mathrm{S1}} - t_{\mathrm{M1}},\ t_{\mathrm{SM}} = t_{\mathrm{M2}} - t_{\mathrm{S2}}. \tag{3.15}$$

- The *round-trip time RTT* is defined as the total time it takes for a message to be transmitted from a source node to a destination node and then back to the source. In this context, it is equal to

$$RTT = t_{\mathrm{MS}} + t_{\mathrm{SM}}. \tag{3.16}$$

- Let T_{M1} and T_{M2} be the timestamps assigned by the master when the first message departs and when the second message arrives, respectively. Similarly, let T_{S1} and T_{S2} be the timestamps assigned by the slave when the first message arrives and when the second message departs, respectively. They are given by

$$T_{\mathrm{M}i} = \tau_{\mathrm{M}} \left(t_{\mathrm{M}i} \right), T_{\mathrm{S}i} = \tau_{\mathrm{S}} \left(t_{\mathrm{S}i} \right) \tag{3.17}$$

  for $i = 1, 2$.

Initially, the only values that are known to the system are the timestamps $T_{\mathrm{M}i}$ and $T_{\mathrm{S}i}$. In fact, $T_{\mathrm{M}i}$ are obtained by the master and $T_{\mathrm{S}i}$ by the slave, so they need to be transmitted to the opposite node in order to perform calculations. This can be achieved either by piggybacking timestamps on the same two timing messages, or using additional messages later; this choice has no impact on the synchronization procedure.

The timestamp difference $T_{\mathrm{S2}} - T_{\mathrm{S1}}$ describes the length of the time interval between the departure of the second message and the arrival of the first one, from the point of view of the slave's local time reference. The crux of the method lies in the observation that this description is as accurate in a pre-synchronized state as it is after synchronization, because the clock quality condition (3.9) ensures that

$$T_{\mathrm{S2}} - T_{\mathrm{S1}} = \tau_{\mathrm{S}} \left(t_{\mathrm{S2}} \right) - \tau_{\mathrm{S}} \left(t_{\mathrm{S1}} \right) \approx t_{\mathrm{S2}} - t_{\mathrm{S1}} \tag{3.18}$$

independently of the current state or timestamp value, with an error depending on the maximum clock drift. Similarly, the difference of master timestamps yields

$$T_{\mathrm{M2}} - T_{\mathrm{M1}} = \tau_{\mathrm{M}} \left(t_{\mathrm{M2}} \right) - \tau_{\mathrm{M}} \left(t_{\mathrm{M1}} \right) \approx t_{\mathrm{M2}} - t_{\mathrm{M1}}. \tag{3.19}$$

Taking the difference between these two values, one obtains

$$\begin{aligned} \left(T_{\mathrm{M2}} - T_{\mathrm{M1}} \right) - \left(T_{\mathrm{S2}} - T_{\mathrm{S1}} \right) &\approx \left(t_{\mathrm{M2}} - t_{\mathrm{M1}} \right) - \left(t_{\mathrm{S2}} - t_{\mathrm{S1}} \right) \\ &= \left(t_{\mathrm{M2}} - t_{\mathrm{S2}} \right) + \left(t_{\mathrm{S1}} - t_{\mathrm{M1}} \right) \\ &= t_{\mathrm{MS}} + t_{\mathrm{SM}} = RTT. \end{aligned} \tag{3.20}$$

Thus, the four captured timestamps can be used to obtain an accurate measure of the real round-trip time; it is, in fact, perfectly accurate in the absence of clock drift.

Knowledge of the round-trip time is not enough in itself to accurately synchronize the time references in both nodes. The way that typical time synchronization algorithms circumvent this issue, including NTP and PTP, is by making the following additional assumption:

Assumption (Link symmetry)**.** *The link is completely symmetrical in the sense that the flight time is the same in both directions, i.e.*

$$t_{\mathrm{MS}} = t_{\mathrm{SM}} = \mathit{RTT}/2. \tag{3.21}$$

Under the hypothesis of symmetry, the flight times in each direction can be determined from the captured timestamps and it is possible to correct the offset between the local time references in the master and slave nodes. This offset is given by

$$\Delta T_{\mathrm{S}}\left(t\right) = \tau_{\mathrm{M}}\left(t\right) - \tau_{\mathrm{S}}\left(t\right) \tag{3.22}$$

and is approximately constant, its variation being due to clock drift. It can then be computed by particularizing e.g. at t_{S1} as

$$\begin{aligned} \Delta T_{\mathrm{S}} &= \tau_{\mathrm{M}}\left(t_{\mathrm{S1}}\right) - \tau_{\mathrm{S}}\left(t_{\mathrm{S1}}\right) \\ &= \tau_{\mathrm{M}}\left(t_{\mathrm{M1}} + t_{\mathrm{MS}}\right) - T_{\mathrm{S1}} \approx T_{\mathrm{M1}} - T_{\mathrm{S1}} + t_{\mathrm{MS}} \end{aligned} \tag{3.23}$$

where condition (3.9) has been used again. Substituting $t_{\mathrm{MS}} = \mathit{RTT}/2$, with RTT given by (3.20), one obtains finally

$$\Delta T_{\mathrm{S}} \approx \frac{1}{2}\left(\left(T_{\mathrm{M1}} + T_{\mathrm{M2}}\right) - \left(T_{\mathrm{S1}} + T_{\mathrm{S2}}\right)\right). \tag{3.24}$$

Synchronization can thus be achieved by adding the value ΔT_{S} to the slave's local time reference, e.g. by updating its timestamp counter value without interrupting its counting operation.

It should be noted that the pairwise step in no way requires the corresponding nodes to be adjacent or their link to be dedicated; it may be implemented across larger networks using multi-hop links or shared media, as is usually the case with NTP. It will be seen, however, that restricting the situation to nodes connected with a point-to-point link increases the synchronization accuracy.

3.2.2 The effect of clock drift

A more detailed analysis of the algorithm described above may be performed, taking into account the effect of the clock drift values δ_{M} and δ_{S} in the master and slave nodes, respectively. Assume that both drift values remain constant for the time interval $[t_{\mathrm{M1}}, t_{\mathrm{M2}}]$ during which the messaging process takes place. Then the errors in the estimations (3.18) and (3.19) are equal to $\delta_{\mathrm{S}}\left(t_{\mathrm{S2}} - t_{\mathrm{S1}}\right)$ and $\delta_{\mathrm{M}}\left(t_{\mathrm{M2}} - t_{\mathrm{M1}}\right)$, respectively. Thus, the error in the estimation (3.20) of the round-trip time is

$$\delta_{\mathrm{M}}\left(t_{\mathrm{M2}} - t_{\mathrm{M1}}\right) - \delta_{\mathrm{S}}\left(t_{\mathrm{S2}} - t_{\mathrm{S1}}\right). \tag{3.25}$$

The calculation of the offset (3.23) also needs to take drift into consideration and be modified as

$$\Delta T_{\mathrm{S}}\left(t_{\mathrm{S1}}\right) = T_{\mathrm{M1}} - T_{\mathrm{S1}} + t_{\mathrm{MS}} + \delta_{\mathrm{M}} t_{\mathrm{MS}}. \tag{3.26}$$

Substituting the round-trip time with drift error in, and maintaining the assumption of perfect link symmetry, one finally obtains that the error in the estimation (3.24) is equal to

$$\begin{aligned}
&\frac{1}{2}\delta_{\mathrm{M}}\left(t_{\mathrm{M2}}-t_{\mathrm{M1}}\right)-\frac{1}{2}\delta_{\mathrm{S}}\left(t_{\mathrm{S2}}-t_{\mathrm{S1}}\right)-\delta_{\mathrm{M}}t_{\mathrm{MS}}\\
&=\frac{1}{2}\delta_{\mathrm{M}}\left(t_{\mathrm{M2}}-t_{\mathrm{SM}}-t_{\mathrm{M1}}-t_{\mathrm{MS}}\right)-\frac{1}{2}\delta_{\mathrm{S}}\left(t_{\mathrm{S2}}-t_{\mathrm{S1}}\right)\\
&=\frac{1}{2}\left(\delta_{\mathrm{M}}-\delta_{\mathrm{S}}\right)\left(t_{\mathrm{S2}}-t_{\mathrm{S1}}\right). \qquad (3.27)
\end{aligned}$$

Thus, the remaining error after the synchronization procedure takes place is proportional to the “waiting time” $t_{\mathrm{S2}}-t_{\mathrm{S1}}$ in the slave between both timing messages, and to the difference in clock drift values between master and slave, or equivalently, to the relative clock drift of the slave with respect to the master. A few conclusions may be extracted regarding the effect of clock drift:

- The whole messaging procedure should be performed as fast as possible by minimizing the waiting time $t_{\mathrm{S2}}-t_{\mathrm{S1}}$, thereby reducing the error due to drift. Moreover, a fast operation ensures that the hypothesis of constant drift throughout the whole procedure is actually valid. Notice that no restriction has been imposed until now on the two messages themselves - in fact, they may be any regular data messages being transmitted between the nodes for any other reason. The only restriction is thus that the delay between both should be small.

- The error depends on $\delta_{\mathrm{M}}-\delta_{\mathrm{S}}$, confirming the intuitive observation that the method for offset estimation must be more accurate if both clocks “drift in the same direction”. In particular, if the clocks are syntonized then the algorithm introduces no drift-related error even if their frequency has a non-zero drift.

- After the synchronization procedure takes place, the master and slave time references are synchronized but will tend to drift away from each other again due to the relative clock drift. Periodic repetition of the procedure is thus needed in order to keep the synchronization error within bounds. An argument similar to that provided for the error bound (3.13) in reset trees shows that the bound in this case is equal to $\left(\delta_{\mathrm{M}}-\delta_{\mathrm{S}}\right)T_{\mathrm{resync}}$, where T_{resync} is the period between resynchronization operations. If only the maximum drift δ is known, then the bound becomes $2\delta T_{\mathrm{resync}}$; the performance with respect to clock drift is therefore the same as that of a reset tree.

Regarding the resynchronization period, it should be noted that its requirements depend on whether the time synchronization algorithm estimates clock drift for better correction or not. In case drift is taken into account up to first order, i.e. assuming constant clock drift, it is shown in [123] that the optimal resynchronization frequency is around 1 Hz to 10 Hz; higher frequencies do not improve accuracy, whereas lower frequencies suffer from deviations due to temperature and other drifts.

3.2.3 The effect of link asymmetry

The implications of the assumption of link symmetry will be examined now. The synchronization errors due to clock drift will be disregarded for now, either by assuming that clock drift values are low enough or that the timestamp clocks in both nodes are syntonized, so that the approximation signs in equations (3.18), (3.19), (3.20) and (3.23) become equality signs.

Let us define the *link skew* or *link asymmetry* as

$$\Delta t = t_{\mathrm{MS}} - t_{\mathrm{SM}} \tag{3.28}$$

i.e. the difference in one-way flight times from master to slave and vice versa. It is then possible to express the flight times in terms of the round-trip time and the link skew by noticing that equations (3.16) and (3.28) provide a 2×2 linear system in the unknowns t_{MS} and t_{SM} that can be solved to obtain

$$t_{\mathrm{MS}} = \frac{RTT}{2} + \frac{\Delta t}{2},\ t_{\mathrm{SM}} = \frac{RTT}{2} - \frac{\Delta t}{2}. \tag{3.29}$$

Substituting this value into (3.23) and replacing the round-trip time with its known value (3.20) yields the modified expression for the time reference offset

$$\Delta T_{\mathrm{S}} = \frac{1}{2}\left((T_{\mathrm{M1}} + T_{\mathrm{M2}}) - (T_{\mathrm{S1}} + T_{\mathrm{S2}}) + \Delta t\right). \tag{3.30}$$

This equation is now exact except for the clock drift error term. It follows that the error introduced by the assumption of symmetry is equal to $\Delta t/2$.

For a different interpretation of this result, suppose that the value of Δt is known with an uncertainty $\sigma_{\Delta t}$, and that the original algorithm is modified in order to take link skew into account by computing the timestamp offset as (3.30) instead of (3.24) and using that value to update the slave's local time reference. The resulting uncertainty in the synchronization accuracy is then $\sigma_{\Delta t}/2$. By comparison, the Lundelius-Lynch error bound (3.14) given by [119] for $n = 2$ nodes is $\Delta t_{\mathrm{p}}/2$ with Δt_{p} being the maximum possible delay mismatch. Hence, it turns out that the presented pairwise synchronization method, when modified to account for link skew, is optimal in the sense that one cannot hope to guarantee a lower error; the only way to reduce the synchronization error is to reduce the uncertainty in Δt.

Systematic link asymmetry values caused by different latencies in the master-to-slave and slave-to-master data paths are included in Δt but not in $\sigma_{\Delta t}$, and hence cause an error in the original pairwise synchronization algorithm but not in the modified one if they are accounted for. In addition, there are a number of potential contributions to delay uncertainty caused by non-deterministic latency components that introduce errors in both cases. It must be taken into account that this algorithm is employed in a vast array of different situations with implementations spread across a wide range of abstraction levels, and each abstraction level adds its own latency overhead:

- In high-level implementations e.g. software routines in PCs synchronizing their local times via NTP, there are random delays due to variable execution times or possible interrupts, as well as a variable response time to interrupts in the receiver. In order to minimize this effect, messages need to be timestamped as late as possible in the transmitter and as early as possible in the receiver.

- When the method is implemented for synchronization of a pair of nodes that are not adjacent, but rather communicate through a multi-hop link, each hop may add a random latency component given by the waiting time in message queues. Moreover, in packet-switched networks one cannot guarantee that the master-to-slave and slave-to-master paths will go through the same nodes.

- In situations with shared transmission media, such as data buses or wireless networks, a Medium Access Control (MAC) protocol has to be defined for arbitration of the rights of transmission. This typically results in a random delay before message transmission until the channel is clear. The related uncertainty may be removed by assigning departure timestamps just as the message is being physically transmitted, but this requires specific hardware assistance.

In any case, it is clear that the lower the level of the algorithm implementation (i.e. the closest to the hardware), the lower the uncertainty in message transmission delay that can be enforced, and thus the better the synchronization accuracy that can be achieved.

3.2.4 The effect of clock phase

Finally, let us consider the best possible case, where the two aforementioned contributions to synchronization errors (clock drift and link asymmetry uncertainty) are reduced to the maximum possible extent. Specifically, consider a point-to-point link where the timestamping logic in both nodes is working with a syntonized frequency $f_{\mathrm{clk}} = 1/T_{\mathrm{clk}}$ with no drift, and the entire synchronization algorithm is performed by dedicated hardware. Moreover, the synchronization and timestamping logic is located at a lower level than the protocol logic for the data link, meaning that no protocol-related, potentially random delays are introduced into the synchronization process. Flight times are therefore deterministic and it is possible to obtain Δt and use it in the algorithm.

If the DAQ uses system-synchronous clocking, then there are no further synchronization errors to be analyzed. However, if the data links are based on source-synchronous or self-synchronous signaling, then there are further errors enforced by the granularity of the timestamp counters or, equivalently, by the phases of the different clock signals involved in the process. Two different steps in the synchronization method may be identified where rounding errors are introduced that, perhaps unsurprisingly, impose a limit on the overall synchronization accuracy in the order of T_{clk}.

The most obvious one regards the application of formula (3.30) for updating the value held by the timestamp counter. Since the counter can only describe time values that are integer multiples of the clock period, the correction offset may only be applied with a resolution of T_{clk}. Even in the ideal case of perfect link symmetry ($\Delta t = 0$), where the value inside parentheses in (3.30) is also an integer multiple of T_{clk}, the correction offset varies in steps of $T_{\mathrm{clk}}/2$ and thus may sometimes be impossible to be applied accurately. Essentially, limiting the variation of the local time references to steps of T_{clk} equates to neglecting the fact that the phases of the timestamp counter clocks on different nodes may be different for non-system-synchronous schemes: if their active edges are not aligned, then it is impossible for them to hold the same values at all times.

The other problem is more subtle and may be first observed by considering the formula (3.20) that yields the round-trip time in terms of the captured timestamps. Notice that this value is always an integer multiple of T_{clk}, because both terms in parentheses represent the time difference between specific active edges of the same local clock. This is incompatible with the intuitive observation that flight times depend linearly on the propagation time through external transmission lines, and thus their sum *RTT* might be forced to take any intermediate value just by tweaking the length of the transmission media.

The fundamental error is given by the definition of reception timestamps T_{S1} and T_{M2}. As outlined in §3.1.1, the received data streams in each node are initially clocked with the clock recovered from the link, from either the strobe line or the data line. However, the timestamps are assigned within the local synchronization logic that belongs to the local clock domain. There is a clock domain crossing in between, depicted as an elastic buffer in Figs. 3.5 and 3.8, due to the potential phase difference between both clocks even though they are syntonized. This crossing introduces an unforeseen delay into the flight time budget.

To be a bit more precise, consider Fig. 3.12, where the components in both data paths between the master and the slave nodes are detailed in the case where the

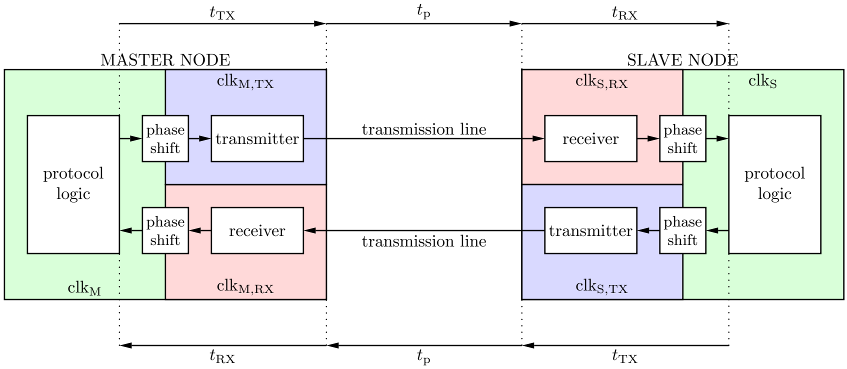


Figure 3.12: Diagram of a source- or self-synchronous bidirectional data link with different clock domains and potential crossings along the path loop.

clock is transmitted together with the data, either as a separate line or embedded in the latter. Each half-link between the timestamping logic blocks features a transmitter, a receiver, and external transmission media, with latencies t_{TX}, t_{RX} and t_{p}, respectively. Data generated by each receiver unit needs to timed with respect to the clock recovered from the link, and so communication between the receiver and the local logic may cross between clock domains if the local clock is derived from a different source. Similarly, transmitters may be driven by their own clock sources independently of the local timestamping logic. It turns out that there are three different syntonized clock domains in each node:

- The local clock ($\mathrm{clk}_{\mathrm{M}}$ and $\mathrm{clk}_{\mathrm{S}}$).
- The transmitter clock ($\mathrm{clk}_{\mathrm{M,TX}}$ and $\mathrm{clk}_{\mathrm{S,TX}}$).
- The receiver, or recovered, clock ($\mathrm{clk}_{\mathrm{M,RX}}$ and $\mathrm{clk}_{\mathrm{S,RX}}$).

Let us denote the binary relation of dependency between clocks as $\mathrm{clk}_i \prec \mathrm{clk}_j$, meaning that clk_j is derived from clk_i. It is clear that this is a partial order on the set of clock signals in the system. Under this notation, the relations $\mathrm{clk}_{\mathrm{M,TX}} \prec \mathrm{clk}_{\mathrm{S,RX}}$ and $\mathrm{clk}_{\mathrm{S,TX}} \prec \mathrm{clk}_{\mathrm{M,RX}}$ hold in the described setting; more specifically, the clocks are pairwise different, as the phase difference between each clock pair is given by

$$\phi_{\mathrm{S,RX}} - \phi_{\mathrm{M,TX}} = 2\pi \frac{t_{\mathrm{p}} \bmod T_{\mathrm{clk}}}{T_{\mathrm{clk}}} \tag{3.31}$$

as per (3.1), since the receiver clock is just the transmitter clock from the opposite node after being transmitted through the connection media.

As it turns out, there are up to four potential clock domain crossings with an associated phase shift along the whole loopback path from master to slave and back,

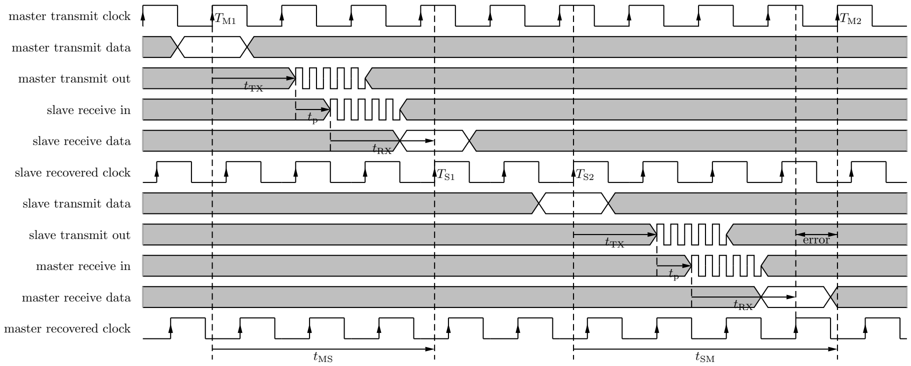


Figure 3.13: Timing diagram for pairwise synchronization across an asynchronous bidirectional link. Clock phase difference distorts the slave-to-master transfer time measure and needs to be taken into account.

all of them shown in Fig. 3.12: one between the local logic and the transmitter and another one between the receiver and the local logic in each node. It is entirely possible to avoid some of these crossings by introducing additional dependencies between the clocks. For instance, clk_M and clk_S may be forced to be the same as $\mathrm{clk}_\mathrm{M,TX}$ and $\mathrm{clk}_\mathrm{S,TX}$, respectively, in many situations. It may also be possible to use the recovered clock to drive the local logic and timestamp counter in one end, e.g. at the slave. However, it is not possible to do all of this on both ends at the same time, because that would lead to the following infinite loop in clock dependency

$$\mathrm{clk}_\mathrm{M} \prec \mathrm{clk}_\mathrm{M,TX} \prec \mathrm{clk}_\mathrm{S,RX} \prec \mathrm{clk}_\mathrm{S} \prec \mathrm{clk}_\mathrm{S,TX} \prec \mathrm{clk}_\mathrm{M,RX} \prec \mathrm{clk}_\mathrm{M} \qquad (3.32)$$

that forces equality of all terms, which has already been seen not to be the case (because $\mathrm{clk}_\mathrm{M,TX} \neq \mathrm{clk}_\mathrm{S,RX}$ and $\mathrm{clk}_\mathrm{S,TX} \neq \mathrm{clk}_\mathrm{M,RX}$). It follows that at least one of these dependencies must be false, and therefore at least one clock domain crossing must be present in the system.

For instance, consider the simplest case, where local clocks and transmitter clocks coincide and the recovered clock is also used as local and transmitter clock in the slave node, so that there is a single clock domain crossing, located between $\mathrm{clk}_\mathrm{M,RX}$ and clk_M. The situation is outlined in the timing diagram in Fig. 3.13. The phase difference between the local and recovered clocks at the master introduces an error whose effect is to round T_M2 up to the next active edge from the local clock, thus rounding the round-trip time (3.20) into the nearest higher multiple of T_clk.

Each of the effects described in this section introduces an error with an average value of $T_\mathrm{clk}/2$, if no additional information is known. Hence, a limit is imposed on the synchronization accuracy in the order of T_clk. Fortunately, the analysis also suggests several changes to be applied to the usual setting for the synchronization scheme in order to overcome this limitation:

- Timestamp values have to allow for a finer granularity than the timestamp clock period. Local timestamp counters need to include a fractional part that can be adjusted by the synchronization protocol, even if the local clock only updates its integer part.
- The link skew Δt needs to be incorporated into the algorithm and measured with an accuracy well below T_{clk}.
- Phase shifts in clock domain crossings need to be either compensated or accounted for in the skew term Δt. In particular, for each of the four potential clock domain crossings in the bidirectional link, clocks need to either be perfectly aligned (i.e. remove the crossing), or have a constant, known phase difference. Since at least one of these clock domain crossings must always be present, it becomes necessary to implement a method for accurate online estimation of the phase difference at any remaining crossings.

3.3 Link skew measurement

As the analysis of the pairwise synchronization algorithm over data links conducted in §3.2 has revealed, the proper approach to optimizing the synchronization accuracy between two nodes that are directly connected to each other involves an accurate modeling and measurement of the link skew Δt between the two half-links connecting them, including any phase shifts that may arise in clock domain crossings; the expected synchronization error is then equal to half the uncertainty in Δt.

This section is devoted to the analysis of the link skew. Its description obviously depends on the particular implementation of the data link, so some conditions need to be fixed from the start. For the remainder of this section, a full duplex data link will be considered, consisting of two self-synchronous serial half-links between the end nodes where the protocol and synchronization logic is implemented in FPGA devices with embedded transceivers, with the same controlled situation as in §3.2.4 i.e. syntonized local clocks and static half-link latencies. Moreover, it will be assumed that the local and transmitter clocks coincide in each FPGA, i.e. $\mathrm{clk}_{\mathrm{M}} = \mathrm{clk}_{\mathrm{M,TX}}$ and $\mathrm{clk}_{\mathrm{S}} = \mathrm{clk}_{\mathrm{S,TX}}$ in the notation of §3.2.4, so that only the clock domain crossings between local and recovered clock are considered. The situation with source-synchronous links or standalone transceivers with additional phase shifts is not entirely different and an analogous treatment may be applied to that case.

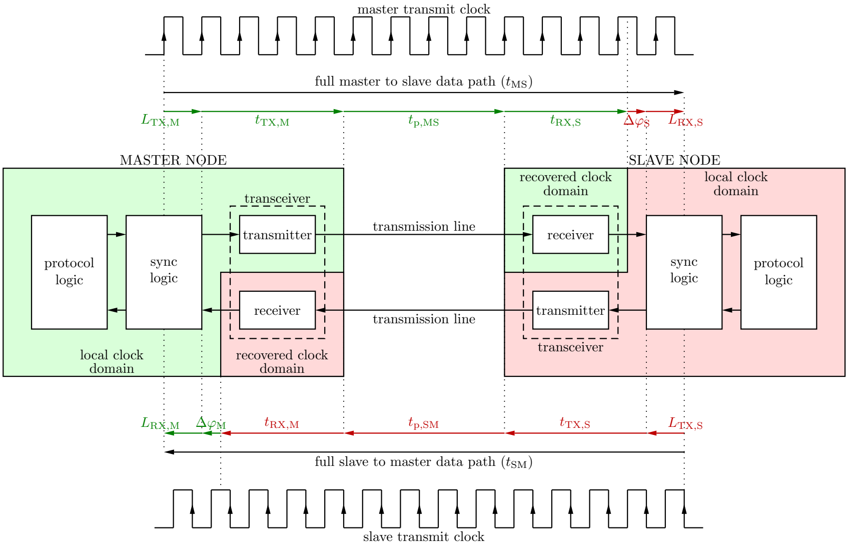


Figure 3.14: Diagram of the flight time components in a bidirectional data link.

The selected setting is outlined in Fig. 3.14, where all delay components and low-level hardware blocks involved in the synchronization are depicted. Specifically, this includes master and slave FPGAs containing the data transceivers and a "synchronization logic" block between them and the protocol decoding logic, which is responsible for accessing the timestamp counter, timestamping messages and implementing the synchronization protocol. Two separate but syntonized clock domains are considered for each FPGA: the recovered clock and the local clock, the latter being used also for transmission.

In order to avoid the appearance of new, unexpected errors stemming from the definition of the timestamps involved in the synchronization procedure, their treatment should be rigorous and careful. Recall that the timestamps $T_{\mathrm{M}i}$ and $T_{\mathrm{S}i}$ are defined in terms of the point in time when the messages depart and arrive; conversely, the travel time under consideration must therefore include any time delays between the points in time when these timestamps are latched and assigned. Since these timestamps are necessarily assigned in the synchronization logic block, it follows that the full data paths for t_{MS} and t_{SM} have to start and end within these blocks in their respective nodes. Specifically, the master-to-slave travel time is composed of the following sequential elements:

- A communication protocol latency $L_{\mathrm{TX,M}}$ between the instant that the transmission timestamp is latched and assigned as T_{M1} and the moment when the message enters the physical transmitter.
- The latency of the master transmitter itself, $t_{\mathrm{TX,M}}$.

- The propagation delay along the transmission line between the master and the slave, $t_{\mathrm{p,MS}}$.
- The latency of the slave receiver, $t_{\mathrm{RX,S}}$.
- The latency associated to the clock domain crossing between the receiver and the local logic. Any additional register-to-register latency between the same clock will be assimilated into the last or next element in this list. Hence, this latency corresponds to the delay between an active clock edge in the recovered clock domain and the next active clock edge in the local clock domain, which is equal to the phase difference (in terms of time) under the assumption of syntonization. This component is therefore equal to $\Delta\varphi_{\mathrm{S}} \cdot T_{\mathrm{clk}}/2\pi$, where $\Delta\varphi_{\mathrm{S}}$ is the phase difference between both clocks.
- A communication protocol latency $L_{\mathrm{RX,S}}$ between the moment that the message arrives at the synchronization logic block and the instant when the slave timestamp is latched and assigned as T_{S1}.

As a summary, the master-to-slave flight time can be expressed as

$$t_{\mathrm{MS}} = L_{\mathrm{TX,M}} + t_{\mathrm{TX,M}} + t_{\mathrm{p,MS}} + t_{\mathrm{RX,S}} + \frac{\Delta\varphi_{\mathrm{S}}}{2\pi} T_{\mathrm{clk}} + L_{\mathrm{RX,S}}. \tag{3.33}$$

A completely analogous analysis can be performed for the slave-to-master travel time, resulting in the decomposition

$$t_{\mathrm{SM}} = L_{\mathrm{TX,S}} + t_{\mathrm{TX,S}} + t_{\mathrm{p,SM}} + t_{\mathrm{RX,M}} + \frac{\Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}} + L_{\mathrm{RX,M}} \tag{3.34}$$

where the meaning of each term should be obvious by analogy with the notation for the case of t_{MS}. Subtracting both of them, the full expression of the link skew is obtained:

$$\begin{aligned}\Delta t = {} & (L_{\mathrm{TX,M}} - L_{\mathrm{TX,S}}) + (t_{\mathrm{TX,M}} - t_{\mathrm{TX,S}}) + (t_{\mathrm{p,MS}} - t_{\mathrm{p,SM}}) + \\ & + (t_{\mathrm{RX,S}} - t_{\mathrm{RX,M}}) + \frac{\Delta\varphi_{\mathrm{S}} - \Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}} + (L_{\mathrm{RX,S}} - L_{\mathrm{RX,M}}) .\end{aligned} \tag{3.35}$$

Two reasonable assumptions may be made in order to simplify this expression. First of all, the implementation of the communication protocol hardware will likely be the same in the master and the slave, and so the transmission and reception latencies should be equal in both nodes in terms of an integer number of clock cycles. Since the nodes are syntonized, it follows that $L_{\mathrm{TX,M}} = L_{\mathrm{TX,S}}$ and $L_{\mathrm{RX,M}} = L_{\mathrm{RX,S}}$ (although not necessarily $L_{\mathrm{TX}} = L_{\mathrm{RX}}$). Even if the implementations happen to be different, these latencies should be fixed and known in a fully controlled situation, and therefore introduce a known, completely deterministic skew.

The second assumption regards the propagation delays through the transmission lines. In the most general case, these delays need not be equal, and even if there is a known, systematic length difference between them, the fact that propagation delay coefficients suffer from drift due to environmental conditions implies that the delay difference between them may present some uncertainty along time. In an attempt to minimize this uncertainty, one may resort to the implementation of completely symmetrical transmission media, i.e. parallel, equal-length PCB traces for transmission and reception and composite cabling for the wires or fibers between both nodes, thereby ensuring equal length as well as identical exposure to changes in environmental conditions. It will be assumed that this is indeed the case, and hence $t_{\mathrm{p,MS}}$ and $t_{\mathrm{p,SM}}$ remain equal throughout the whole DAQ operation, save for a comparatively small uncertainty $\sigma_{t_{\mathrm{p}}}$.

Taking the aforementioned considerations into account, the expression (3.35) for link skew can be simplified into

$$\Delta t = (t_{\mathrm{TX,M}} - t_{\mathrm{TX,S}}) + (t_{\mathrm{RX,S}} - t_{\mathrm{RX,M}}) + (\Delta\varphi_{\mathrm{S}} - \Delta\varphi_{\mathrm{M}})\,\frac{T_{\mathrm{clk}}}{2\pi} + \sigma_{t_{\mathrm{p}}} \tag{3.36}$$

where the last term $\sigma_{t_{\mathrm{p}}}$ may be disregarded except for very specific situations. The main components of link skew are thus the asymmetry between transceiver latencies and the phase shifts between local and recovered clocks at each node. The next two sections deal with the distribution and measurement of each of those components. It will be seen that the main contribution to uncertainty in Δt comes from the measurement of phase differences, whereas the difference in transceiver latency can be very accurate with proper device selection.

3.3.1 Transceiver latency

The requirement that the uncertainty in Δt be minimized implies that the latency differences $t_{\mathrm{TX,M}} - t_{\mathrm{TX,S}}$ and $t_{\mathrm{RX,M}} - t_{\mathrm{RX,S}}$ should be known exactly if possible. Fortunately, the latency of a transceiver element in the data path may be assumed to be static, i.e. it remains fixed after the link has been established. However, in virtually all available implementations, each latency term may take different values after initialization. The condition to be imposed is therefore not that these latencies are fixed but rather that they are *deterministic* in the sense described in §3.1.2, i.e. that their static terms may be known at runtime. This ensures that the latency differences consist of a constant term (typically zero if both nodes use the same hardware) plus a variable term whose value can be obtained accurately.

Consider Fig. 3.8 depicting the usual components in an embedded transceiver and showing the boundaries between different clock domains. The nature of their latency can be analyzed:

- In any block that is completely located within a single clock domain, the path followed by the data consists entirely of register to register transfers, and so its latency is an integer amount of clock cycles. Hence, it may be assumed that these blocks feature a fixed latency - if the execution time of the processing they carry out happens to be variable, it is an easy design exercise to modify them to include a variable amount of wait cycles at the end in order to equalize the delay to the worst case i.e. longest value.

- The output driver at the transmitter is essentially an analog subcircuit with a fixed propagation time.

- Unlike the case of frequency division shown in Fig. 3.7, there is no uncertainty regarding the timing of the bit clock with respect to the transmit clock as long as they are generated at the same PLL and their edges are aligned - indetermination only appears for frequency division, but not for frequency multiplication. Hence, the latency introduced by the serializer is fixed.

- The CDR block at the receiver may also be assumed to have a fixed latency, as it is timed by the bit clock embedded in the data stream.

The only remaining blocks with potentially variable latency are then the deserializer and the clock adaptation stages.

Deserialization and word alignment

As was already discussed in §3.1.1, extracting the word clock from the embedded bit clock entails an uncertainty in that there are initially m possible versions of the word clock, where m is the length of the physical word; discrimination between them relies on the detection of particular bit sequences in the stream. These parallel clocks are uniformly distributed in phase with a difference of $2\pi/m$, as determined by the latency steps of $1\,UI = T_{\mathrm{clk}}/m$. Suppose that one of them is selected at random. Words generated by the deserializer do not then correspond to actual data words, but rather to bit shifted versions of them, containing parts of two consecutive original words. The correct words are thus found at some relative position with respect to the first bit in the word, in terms of a fixed number of shifted bits.

Consider the example described in Fig. 3.15, corresponding to the particular case where the order of bit transmission for each word is from the most to the least significant bit, i.e. *MSB first*. The latency introduced by the deserializer is equal to the time delay between an active edge of the correct word clock, corresponding to the actual timing of the received bit stream, and the next active edge of the selected parallel clock, which drives the timing of the deserializer output. Notice

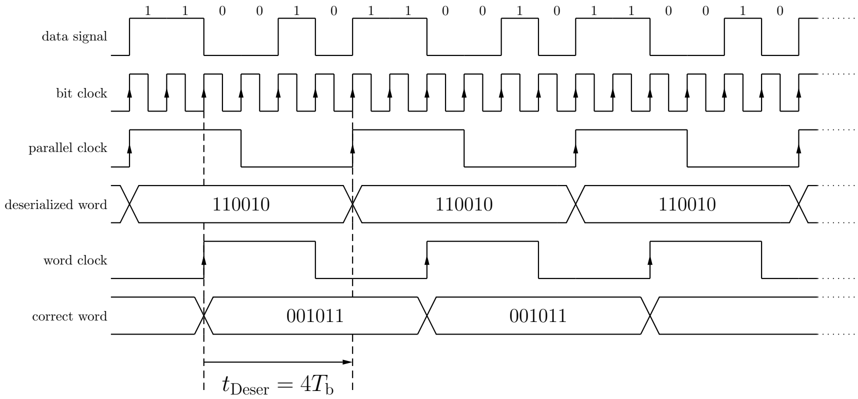


Figure 3.15: Illustration of the relationship between deserializer latency and relative bit shift between deserialized words and actual words, for the alignment word 001011 and MSB first transmission.

how this delay is precisely equal to

$$t_{\text{Deser}} = n\,T_{\text{b}} = n\,\frac{T_{\text{clk}}}{m} \tag{3.37}$$

where n is the number of bits that the correct word needs to be shifted *to the left* in order to obtain the deserialized words, or equivalently, the number of bits that the words generated by the deserializer need to be shifted *to the right* in order to obtain the correct words.

Analogous considerations reveal that, for *LSB first* transmission, n is equal to the number of bits that deserialized words need to be shifted *to the left* in order to obtain the correct words. In any case, the value of n in (3.37) is completely determined by the relative shift of correct data words, the shift direction being itself determined by bit transmission order. The deserializer thus has deterministic latency if and only if the value of n can be either determined or externally enforced.

The case is analyzed first where the parallel clock in the receiver is randomly chosen from within the m available choices during PLL convergence without regard for the alignment code word, and then remains fixed for the remainder of the link operation. A word aligner block is present after deserialization that searches for the presence of a certain self-synchronizing code within the received data. There are essentially two approaches to that searching process:

- A brute force approach where the word stream is deserialized and all substrings of the bit sequence formed by two consecutive words are compared to the alignment word. If the word is found, then its relative position with respect to the start of the deserialized words directly yields the value of n.

- An iterative approach where only the current word is compared to the self-synchronizing code, but a programmable bit shifter is present before that stage. If the alignment word is not found in the word stream, words are shifted by one bit, the shift count is increased and the operation is repeated. When the alignment word is finally found as a shifted word, the resulting shift count is equal to n, provided that the shift direction is correctly chosen according to the considerations given above.

The second method is usually preferred because it allows for a simpler implementation, the difference in alignment time being negligible given that alignment procedures do not limit themselves to the detection of the self-synchronizing pattern just once, but rather an appropriate amount of times over a given bit stream length. Bit shifts are often implemented as part of the deserializer itself in terms of *bit slips*.

The deserializer latency will thus be deterministic as long as the user logic has access to the number n of bit slips performed during alignment. Some transceiver IPs provide this number directly as an output signal. In the opposite case, deterministic latency designs require bypassing the internal word alignment procedures and hardware blocks in the transceiver and implementing it on user logic, either with the assistance of a bit slip interface if available or using a custom bit shifter, in order to keep track of the amount of bit shifts.

The second possible case is a receiver design where the parallel clock can be switched after PLL convergence, e.g. by way of a clock multiplexer. In this case, after selecting an initial parallel clock randomly, the word alignment logic can determine the correct word clock according to the process described above and then choose it as the parallel clock for the remainder of the link operation. In this case, the deserializer latency is fixed to zero (or T_{clk}) by design. The downside of this approach is the need to switch the recovered clock any time the link needs to be realigned for some reason.

Clock adaptation stages

The clock adaptation stages depicted in Fig. 3.8 are usually implemented as elastic buffers consisting of dual-port FIFO queues as described in §3.1.1, where words are being continuously written and read every clock cycle. If both clocks are syntonized and the phase difference between them is constant, then the latency of the queue element is also constant and its value consists of an integer number of clock cycles plus a fractional part given by the delay between respective active edges, which is determined by the aforementioned phase difference.

The integer number of clock cycles conforming the queue latency consists itself of a fixed access latency (possibly zero) and a variable term that corresponds to the queue fill level, i.e. the difference between its read and write pointers: it is obvious that all previously stored words have to be read before a newly entering data word can be output. The latency of an elastic buffer will therefore be deterministic if and only if its fill level is. Since the fill level remains constant during stabilized operation,[10] the question is reduced to its determinacy at the end of the link establishment process, i.e. after each reset. There are thus several conceivable ways to achieve deterministic latency:

- Including a dedicated hardware mechanism that ensures a fixed fill level after reset, usually half the total queue depth.
- Making the queue's fill level available to user logic as an output signal.
- Providing an alternative data path for measuring the queue latency, e.g. by allowing dynamic transceiver reconfiguration into and out of loopback mode in a way that does not interfere with clock phases. This allows implementing the measurement of the queue latency in user logic, as long as there is only one queue in the path, i.e. in order to measure the latency of the receiver queue, the transmitter queue would either need to be bypassed or its latency should be determined by another method.

If none of these methods are supported by the transceiver design, then it is still possible to guarantee deterministic latency by providing the option of bypassing the elastic buffer, so that the clock domain crossing is implemented as a raw register-to-register transfer, with the source and destination registers being driven by the write and read clocks, respectively. The latency of the crossing is then given by

$$t_{\mathrm{CDC}} = \frac{\Delta\phi}{2\pi} T_{\mathrm{clk}} = (\phi_{\mathrm{dst}} - \phi_{\mathrm{src}}) \frac{T_{\mathrm{clk}}}{2\pi} \tag{3.38}$$

and is already included in (3.35) and (3.36). Notice that some values of $\Delta\phi$ may give rise to setup or hold violations on the destination register and thus generate metastability, leading to possible data corruption even if filtered by a synchronizer. In any case, the queue may safely be bypassed whenever the source and destination clocks are equal, as is the case at the transmitter by assumption.

[10] More accurately, it alternates between two consecutive values each clock cycle, and changes coincide with the read and write clock edges.

3.3.2 Phase measurement

The remaining component for complete link skew characterization is the phase difference between local and recovered clocks. Specifically, a method is required for the measurement of the phase difference between clock signals with exactly the same frequency that are located inside of the same FPGA device.

Phase detection in the digital domain is usually performed by means of a *Phase-Frequency Detector* (PFD), simple circuits involving D-type flip flops for each clock whose outputs are fed back to their asynchronous reset inputs after a fixed delay, generating short pulses that can be used to drive a charge pump whose integrated output is proportional to the phase difference ([124], pp. 43–45). Unfortunately, PFDs feature a small dead zone around the 0° mark where phase measurement is not possible. Another disadvantage is that PFDs require dedicated analog circuitry and thus are difficult to implement entirely inside an FPGA.

One possible approach for the measurement of phase differences that avoids these issues is the *Digital Dual-Mixer Time Difference* (DDMTD) method, first proposed in [125] for a laser range finding application, and then adapted to the White Rabbit time synchronization scheme by Moreira et al. [126]. This circuit is linear over the full phase range with no dead zone and only requires digital gates and a PLL, which can be found in virtually all modern FPGAs. This section focuses on the description and analysis of DDMTD and proposes an improved correction method to enhance its phase resolution.

Digital Dual-Mixer Time Difference (DDMTD)

The DDMTD is essentially an all-digital version of the well-known Dual-Mixer Time Difference technique [127] developed by Allan for fine measurement of the phase difference between two syntonized tones. The original technique consists in mixing both tones $x_i(t) = \cos(2\pi f t + \theta_i)$, $i = 1, 2$ with a common tone $m(t) = \cos 2\pi f' t$ that has a slightly different frequency $f' \approx f$, obtaining the mixed signals

$$x_i(t) \cdot m(t) = \frac{1}{2}\left[\cos\left(2\pi\left(f - f'\right)t + \theta_i\right) + \cos\left(2\pi\left(f + f'\right)t + \theta_i\right)\right] \tag{3.39}$$

consisting of the sum of two tones, one with a very low frequency $|f - f'|$ and another with higher frequency $f + f' \approx 2f$. Passing them through a low-pass filter removes the second term and one obtains $x_i'(t) = \cos\left(2\pi\left(f - f'\right)t + \theta_i\right)/2$, i.e. two tones with the same phase relationship as the original ones but with a much smaller frequency, so that the phase difference may be measured much more accurately e.g. as the time difference between zero crossings.

In order to apply the same principle to the fully digital domain, consider two (or more) syntonized clock signals $c_i(t)$ with frequency f_{clk} whose phase difference needs to be measured, and assume that they are perfectly periodic. A separate,

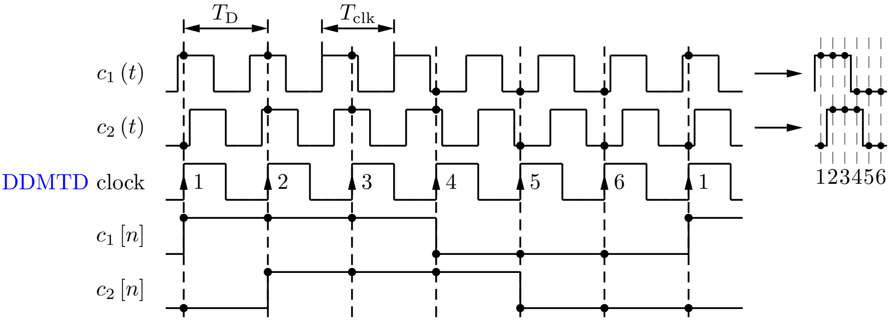


Figure 3.16: Signal waveforms involved in DDMTD for $N = 6$. On the right side, all input clock periods and sampling instants are collapsed together in order to illustrate the zooming effect of DDMTD sampling.

common DDMTD clock is generated with a frequency $f_D = 1/T_D$ on which no assumption is made initially. The mixing operation is here replaced by sampling with the common clock, implemented with simple flip-flops where the $c_i(t)$ are used as the inputs. The n-th sample is then given by

$$c_i[n] = c_i(nT_D) = c_i\left(n\frac{T_D}{T_{clk}}T_{clk}\right) = c_i\left(\left\{n\frac{T_D}{T_{clk}}\right\}T_{clk}\right) \tag{3.40}$$

where $\{x\}$ denotes the fractional part of x; the last equality follows from the fact that the clock signals $c_i(t)$ have period T_{clk}. The formula can be simplified by forcing a specific relationship between T_D and T_{clk} so that their values are very similar. In particular, the simplest case arises when

$$f_D = \frac{N}{N+1}f_{clk} \tag{3.41}$$

for an integer N called the *stretching factor*. Then the n-th sample becomes

$$c_i[n] = c_i\left(\left\{\frac{n}{N}\right\}T_{clk}\right) = c_i\left(\frac{n \bmod N}{N}T_{clk}\right) = c'_i\left((n \bmod N)\cdot T_{clk}\right) \tag{3.42}$$

where $c'_i(t) = c_i(t/N)$ is the original clock signal after being stretched in time by a factor of N. Hence, the streams of samples correspond to one period of the stretched signals $c'_i(t)$, which have frequency f_{clk}/N but their phase difference is the same as between the original clock signals $c_i(t)$; the time difference between their rising edges is thus amplified by N and more easily measurable. Moreover, the stream of samples has a period of N under the assumption of perfect, stable periodicity of the $c_i(t)$. Fig. 3.16 illustrates the method with example waveforms for $N = 6$.

Fig. 3.17 shows one possible implementation of the DDMTD method inside of an FPGA for the intended application, i.e. measuring the phase difference between the local clock and the recovered clock from the receiver in an embedded

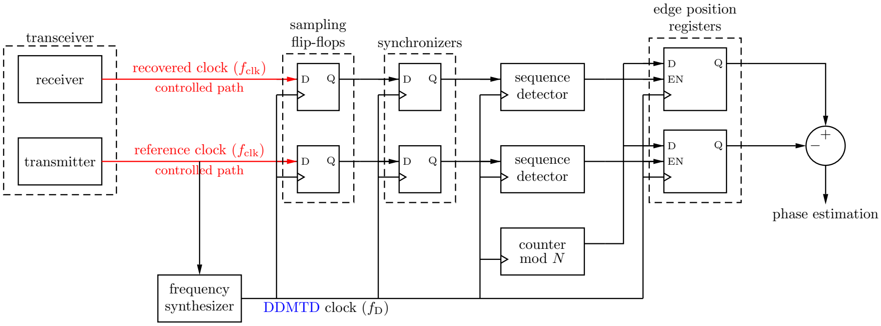


Figure 3.17: FPGA implementation of DDMTD for link skew measurement. The critical paths with controlled latency conditions are highlighted in red.

transceiver. The DDMTD clock is synthesized using the local clock as a reference, since it is likely more stable than the recovered clock, using an embedded PLL or DLL with the frequency ratio given by (3.41). The clock inputs under measure then undergo the following data path:

- Both inputs are sampled by D-type flip-flops under carefully controlled conditions: the propagation latency of both clocks to the data input of the flip-flops must be matched, as well as the sampling edges of the DDMTD clock at both flip-flops; any mismatch will introduce an equivalent phase error.

- Metastable outputs are likely to appear whenever an input clock signal toggles close to a sampling edge. The sampled signals are therefore filtered by a synchronizer chain in order to limit the propagation of metastable states.

- A sequence detector is used to look for active clock edges in the sampled bit strings. The simplest case is the sequence 01, corresponding to a positive edge. Longer patterns such as 0011 may be used for improved noise rejection.

- Detection of an active edge in $c_i\,[n]$ triggers the storage of the corresponding sample number n_i.

After the active edges are detected in both clock signals, an estimation of the phase difference is obtained, given by

$$\Delta\hat{\varphi} = 2\pi\,\frac{(n_1 - n_2) \bmod N}{N}. \tag{3.43}$$

Assuming that all edges get detected, an estimate is obtained for every period, that is every N DDMTD clock cycles or, equivalently, every $N+1$ input clock cycles. Phase difference estimations have a resolution of $2\pi/N$, so that the value of the stretching factor N establishes a trade-off between time resolution and measurement time. Moreover, it is preferable to choose a power of two for N, in order to simplify the FPGA implementation of the modulus operation in (3.43).

Uncertainty in DDMTD measurement

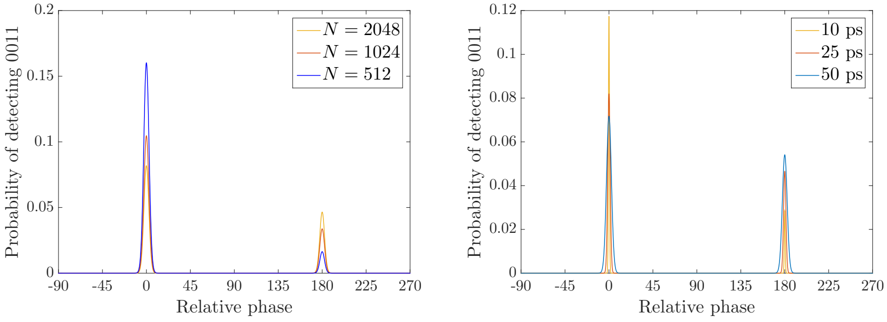
Raw phase difference estimates provided by DDMTD depend on the location where edges are found in the sampled bit stream. However, these detected edge positions are subject to errors derived from two primary sources which imply an uncertainty in the samples close to the input edges, which are precisely where greater precision is required:

- Metastability at the sampling flip-flops may occur whenever the rising edge of the input clock is sampled, resulting in potentially random samples after flip-flop convergence.
- Clock jitter causes the actual position of the input clock edge to vary from sample to sample, because every sample corresponds physically to different edges in time, and this may cause bouncing and glitches in the sampled waveform. Note that the jitter from both the input and the DDMTD clock contribute to this effect.

Both effects manifest themselves as a probability of error at the samples near the clock edges, resulting in fluctuations in edge positioning that translate into fluctuations in the phase difference estimates. Moreover, for edge patterns of length 4 or more, such as 0011, it is entirely possible that no such pattern is detected in a given clock cycle, resulting in a reduced rate of estimates.

The first analysis of the uncertainty in DDMTD phase estimates was published in [126] and expanded in [128], in the simplest case of sequence 01, i.e. simple rising edges. The appearance of glitches due to clock jitter as N increases is acknowledged and two correction methods are proposed and simulated, based on the mean and median positions of all detected 01 edges in the transition. However, there is an additional effect that is not discussed there and can distort the results provided by these correction methods: the appearance of glitches at the negative input clock edges.

Intuitively, it is clear that no transitions are expected in the sampled stream at the stable regions where the sampling point is far off the edges of the input clock, and that glitches and transitions can be expected at the edges. The important observation is that they can be expected at *both* edges. While rising edge patterns are more likely to be found near the positive edge, particularly for longer patterns, a bad streak of errors due to excessive jitter can cause the detection of the rising edge pattern near the negative input clock edge. A particular phase estimation



(a) 50 ps jitter on both clocks

(b) $N = 2048$

Figure 3.18: Probability of detecting the sequence 0011 at different phases for $T_{\text{clk}} = 6.4$ ns and equal absolute jitter values for the input and the DDMTD clocks. Left: Fixed jitter and variable N. Right: Fixed N and variable jitter.

provided by DDMTD has therefore a positive probability p of corresponding to the opposite phase i.e. being detected at the negative edge, and $1-p$ of corresponding to the correct phase.

In an attempt to formalize this line of reasoning and quantify its effects, approximate expressions for the probability of error and the distribution of phase difference estimates are obtained in appendix A under some simplifying assumptions. Equations (A.10) and (A.11) describe the probability that the rising edge sequence 0011 is detected at a particular normalized time t (i.e. at a phase distance of $t \cdot 2\pi$ from the actual rising edge), in terms of the stretching factor and the combined jitter of the input and DDMTD clocks. The qualitative dependence on DDMTD parameters is displayed in Fig. 3.18, where the two predicted probability peaks appear at phases 0° and 180° with respect to the rising edge.[11]

In Fig. 3.18a, a realistic (if somewhat high) jitter value of $\sigma = 50$ ps is assumed on both clocks, with a frequency of 156.25 MHz, and the effect of the stretching factor is observed. Higher values of N decrease the probability of detecting the sequence at the correct edge and increase it at the opposite edge, as they imply a finer sampling with more potential for errors and glitches near the edges. Hence, p increases with N, while the relative width of the peaks remains largely constant, i.e. the uncertainty of *correct* DDMTD measurements depends only on clock jitter.

For Fig. 3.18b, the stretching factor is fixed at 2048 and absolute jitter varies between 10 ps to 50 ps. Lower jitter figures result in narrower peaks, with increased phase resolution in both the correct and the wrong phase peak. Moreover, it also increases the probability of detection near the positive edge and decreases it near

[11] The simplified equations assume a duty cycle of 50 %; see appendix A for details.

the negative edge, i.e. p is decreased. This is to be expected, as reducing the jitter values results in glitches being constrained to a smaller phase interval.

Refinement of phase estimates

The limited resolution provided by raw DDMTD phase measurements merely implies a corresponding uncertainty in the synchronization procedure if used directly, but is not enough to invalidate their use in itself. However, the appearance of wrong values at the opposite phase is a more problematic issue. If DDMTD measurements are used for synchronization via (3.30) and (3.36) without undergoing some post-processing, then the choice of a wrong phase value will imply an error of $T_{\mathrm{clk}}/2$ in Δt and a synchronization mismatch of $T_{\mathrm{clk}}/4$. Simple filtering operations mitigate but do not completely remove the problem. For instance, if averaging is used, either as proposed in [128] or as a moving average, then the wrong phase values distort the result by an expected error $pT_{\mathrm{clk}}/2$ and so the synchronization accuracy is $pT_{\mathrm{clk}}/4$, which may be unacceptable depending on p.

A different approach is proposed here that completely filters out the effect of opposite phase readings. The idea is to use an unsupervised classification method to sort the set of all DDMTD phase estimates into several classes, which are unknown a priori, and then identify one of those classes containing only valid measurements. The method should be very simple and allow a lightweight FPGA implementation as an extension to DDMTD, possibly by profiting from the knowledge of the theoretical distribution of raw phase estimates.

Specifically, the most simple clustering algorithm may be used, sometimes known as Kohonen's algorithm ([129], pp. 130–133). This is a simple iterative method that splits the stream of input data into k clusters, each of them represented by a single value φ_i, $i = 1, \ldots, k$. For each new input value $\hat{\varphi}$ provided to the algorithm, an iterative step is performed where its membership to one cluster or another is assessed in terms of its distance to the representative values $d\left(\hat{\varphi}, \varphi_i\right)$, for some distance function d that needs to be defined. The value $\hat{\varphi}$ is assumed to belong to the closest cluster, i.e. the one whose representative value is at the smallest distance. The corresponding cluster is then updated in order to reflect the entrance of its new member by moving its representative value closer to $\hat{\varphi}$, while the others remain unaffected. The algorithm can be shown to converge in the most simple cases ([129], pp. 143–155).

The application to DDMTD measurement post-processing is straightforward. The expected distribution of raw phase estimates suggests the definition of $k = 2$ clusters in this case, with the expectation that they will come to represent the estimates near the correct value and its opposite phase. Using the difference $(n_1 - n_2) \bmod N$ directly as the estimates $\hat{\varphi}$, these are values ranging from 0 to $N - 1$ representing the phase difference with a proportionality factor $N/2\pi$. The distance function to be used is then the circular distance modulo N

$$d(\varphi, \varphi') = \begin{cases} |\varphi - \varphi'| & , \text{if } |\varphi - \varphi'| < N/2 \\ N - |\varphi - \varphi'| & , \text{if } |\varphi - \varphi'| \geq N/2 \end{cases} \tag{3.44}$$

that takes into account the periodic nature of phase. The initialization procedure for the method consists of taking the first DDMTD estimate $\hat{\varphi}_0$ and setting opposite phases $\varphi_1[0] = \hat{\varphi}_0$, $\varphi_2[0] = \hat{\varphi}_0 + N/2$ as the cluster representatives. For each subsequent phase estimate, its distance to both representatives is evaluated in order to find the closest one and they are then updated as

$$\varphi_i[n+1] = \begin{cases} \varphi_i[n] + \theta \cdot (\hat{\varphi} - \varphi_i[n]) & , \text{if } d(\hat{\varphi}, \varphi_i[n]) < d(\hat{\varphi}, \varphi_{3-i}[n]) \\ \varphi_i[n] & , \text{otherwise} \end{cases} \tag{3.45}$$

for $i = 1, 2$, where the parameter $\theta \in \,]0, 1[$ is called the *learning rate* and controls the performance of the method. The method will eventually converge to a state where the cluster representatives φ_1 and φ_2 hover around two opposite phase values. The cluster that gets selected more often is identified as the one corresponding to correct phase values, following the reasonable assumption that $p < 1/2$. Its representative value φ_i is then used as the current post-processed phase estimate.

The statistical properties of the resulting phase estimates are now analyzed. Assume that, possibly after a short transient period, the two representative values φ_1 and φ_2 are correctly located near the correct phase difference and its opposite, respectively. Each DDMTD phase estimate causes the update of φ_1 with probability $1 - p$ or φ_2 with probability p. Let us restrict the analysis to the update of φ_1 and the DDMTD phase estimates that are assigned to it; the situation for the other cluster will be analogous.

Let $\{X_n\}$ be the sequence of correct DDMTD phase estimates, and $\{\Phi_n\}$ the corresponding sequence of post-processed phase estimates. The random variables X_n are all identically distributed, with mean $\Delta\varphi$ equal to the actual phase difference to be estimated, and variance σ providing a measure of the DDMTD resolution. It is also assumed that the X_n are all independent of each other. The post-processed phase estimates are initialized as $\Phi_1 = X_1$ and then

$$\Phi_n = \Phi_{n-1} + \theta (X_n - \Phi_{n-1}) = (1 - \theta) \Phi_{n-1} + \theta X_n \tag{3.46}$$

for $n > 1$. Notice that Φ_{n-1} and X_n are independent, as Φ_{n-1} is completely determined by X_k for $k < n$.

First of all, Φ_n is shown to be an unbiased estimator of $\Delta\varphi$. Applying the expectation operator to (3.46) yields

$$E[\Phi_n] = (1 - \theta) E[\Phi_{n-1}] + \theta E[X_n] = (1 - \theta) E[\Phi_{n-1}] + \theta \cdot \Delta\varphi. \tag{3.47}$$

This is a first-order linear recurrence equation on $\{E[\Phi_n]\}$ with constant coefficients that can be easily solved. It admits the constant solution $E[\Phi_n] = \Delta\varphi$ and its characteristic polynomial has the root $\lambda = 1 - \theta$, so its general solution is of the form $E[\Phi_n] = \Delta\varphi + C(1-\theta)^n$ for a constant C. Substituting in $\Phi_1 = X_1$ with expectation $\Delta\varphi$ yields $C = 0$ and $E[\Phi_n] = \Delta\varphi$ for all n, as was to be proved.

The variance of Φ_n is obtained next. Taking into account the independence of Φ_{n-1} and X_n, the variance of (3.46) is equal to

$$V[\Phi_n] = (1-\theta)^2 V[\Phi_{n-1}] + \theta^2 V[X_n] = (1-\theta)^2 V[\Phi_{n-1}] + \theta^2\sigma^2 \tag{3.48}$$

i.e. given by a recurrence equation of the same type. In this case, the constant solution is $V[\Phi_n] = \sigma^2 \cdot \theta/(2-\theta)$, whereas the characteristic polynomial has the root $\lambda = (1-\theta)^2$, hence the general solution is of the form

$$V[\Phi_n] = \frac{\theta}{2-\theta}\sigma^2 + C(1-\theta)^{2n} \tag{3.49}$$

for some constant C. Substituting in $\Phi_1 = X_1$, one obtains

$$\frac{\theta}{2-\theta}\sigma^2 + C(1-\theta)^2 = V[\Phi_1] = V[X_1] = \sigma^2 \tag{3.50}$$

which allows C to be expressed in terms of θ and σ; finally, substituting the result back into (3.49) yields

$$V[\Phi_n] = \frac{\theta + 2(1-\theta)^{2n-1}}{2-\theta}\sigma^2. \tag{3.51}$$

In particular, as the number of DDMTD measurements increases

$$\lim_{n\to\infty} V[\Phi_n] = \frac{\theta}{2-\theta}\sigma^2 \approx \frac{\theta}{2}\sigma^2. \tag{3.52}$$

Hence, after a sufficient number of correct phase estimates have been obtained, the uncertainty of the estimate provided by the clustering algorithm is proportional to the uncertainty σ of the correct DDMTD measurements and to $\theta^{1/2}$. The learning rate therefore directly controls the resolution of post-processed estimates.

While a smaller value of θ will result in lower variance and better resolution, this comes at a cost: the response to variations in the real value of $\Delta\varphi$, e.g. caused by drifts in propagation delays, will be slower. For instance, consider the case where there is a sudden jump in phase difference from $\Delta\varphi$ to $\Delta\varphi'$. The analysis of variance remains valid, however the analysis of expectation needs to be updated, as $E[\Phi_1] = \Delta\varphi$ but $E[\Phi_n] = \Delta\varphi'$ for $n > 1$. The revised analysis yields the following general expression for the expectation of estimates:

$$E[\Phi_n] = \Delta\varphi' + (\Delta\varphi - \Delta\varphi')(1-\theta)^{n-1}. \tag{3.53}$$

Hence, $E\left[\Phi_n\right]$ moves from the old to the new value at an exponential rate which is slower the smaller the learning rate becomes.

An efficient FPGA implementation of the clustering algorithm is possible if θ is chosen as a negative power of two, so that the update step (3.45) only requires additions and logical shifts. Selection of the correct cluster may be implemented using a sliding up-down counter, that increments or decrements its value by one each step, depending on the cluster where the input is assigned, starting at the middle value and saturating at the extremes.

3.4 Review of precise synchronization technology

Research on methods for the precise synchronization of distributed DAQs in large-scale installations has been gaining traction since the middle of the 2000s, with the term *precise synchronization* usually being understood to mean *sub-nanosecond* synchronization, i.e. schemes that guarantee a synchronization accuracy smaller than 1 ns. This trend may be explained by a number of factors related to the general evolution of DAQ systems as exposed in §2.2:

- The growth in system complexity forces considerations on availability and forces the support for hot-swapping of faulty or upgradeable modules; automatic time calibration methods are therefore required that do not disturb the operation of the whole system, i.e. without explicit calibration runs.
- Advances in detectors result in improved time resolution, and the limitations on synchronization error become more stringent.
- The deployment of large-scale installations raises concerns about the stability of synchronization over long wires with relatively large latency variations.

Traditional clock trees cannot always yield the desired resolution in large installations; as was mentioned at the beginning of this chapter, variations in the propagation delay of cables or fibers can be one order of magnitude higher than the desired resolution. Regulation of the propagation delay, and therefore bidirectionality of the clock distribution subsystem, is required in order to overcome this problem. Pairwise synchronization over data links offers an alternative, but it has already been seen that it can only achieve accuracies of $T_{\text{clk}}/2$ without considerations about link skew and deterministic latency; this would force $f_{\text{clk}} > 500\,\text{MHz}$ which is currently not practical.

This section is devoted to the description of state-of-the-art technology that makes it possible to implement these schemes for precise clock distribution and synchronization. The availability of commercial devices that are fit for these implementations is discussed first, and a review of the published methods proposed and implemented in various experimental physics installations is included thereafter.

3.4.1 Commercial devices

Clock distribution schemes without a dedicated clock tree rely on the existing data links for synchronization, where transceivers are always the critical element with respect to the support of deterministic latency configurations. Transceivers may come in three different forms: either as standalone devices, as embedded cores in programmable devices (usually FPGAs), or in ASICs. This discussion is centered on commercially available devices and thus transceiver ASICs will not be considered here, save for the quick remark that designs for deterministic latency such as [130] are easy to implement if needed.

Regarding standalone transceivers, none of the commercial devices analyzed during the initial development phases of this thesis was found to feature deterministic latency, including [97–101]. In fact, according to [131], the only off-the-shelf chipset with this feature was Agilent's G-Link family [55], but its production had long been discontinued. There are several reasons why that situation is unlikely to change in the short term. The current popularity of FPGAs, particularly for custom transmission protocol designs, relegates standalone transceivers mostly to specific communication standards or inexpensive designs, none of which require deterministic latency usually. Moreover, support for deterministic latency usually implies an increased level of integration between the transceiver and the local logic, with a higher amount of communication signals (status outputs and configuration inputs) and controlled clock phases, and thus favors implementation within programmable logic devices. The analysis will therefore focus on the available FPGAs from the two main vendors, i.e. Altera and Xilinx.

Deterministic latency in Xilinx transceivers

Xilinx was the first to recognize the need for specific deterministic latency configurations. The first FPGA with embedded Multi-Gigabit Transceivers (MGT) was the Virtex-II Pro, released in 2002; its documentation already appears to allow "unofficial" configuration modes where the elastic buffers are bypassed and deterministic latency can be achieved, but this has not been tested ([132], pp. 89–93). Virtex-4 devices included MGTs [133] that have been reported to successfully support such configurations [134]. However, the most popular platform for the implementation of deterministic latency links seems to be the Virtex-5 with embedded GTP/GTX transceivers [102], which has been evaluated extensively in the available literature [74,135–138]. Needless to say, the successors to the Virtex-5 family remain compliant with deterministic latency.

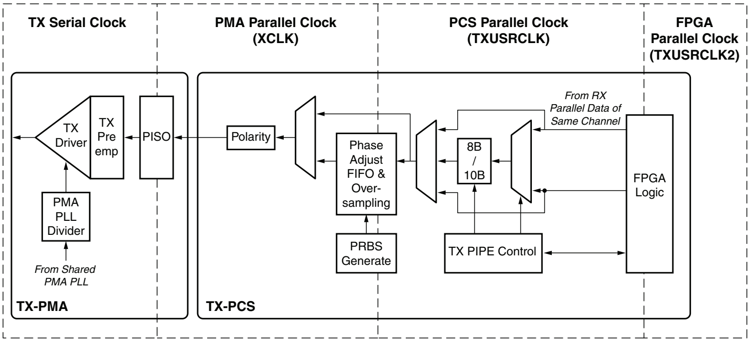

TX Serial Clock
PMA Parallel Clock (XCLK)
PCS Parallel Clock (TXUSRCLK)
FPGA Parallel Clock (TXUSRCLK2)
TX Driver
TX Pre emp
PISO
PMA PLL Divider
From Shared PMA PLL
TX-PMA
Polarity
Phase Adjust FIFO & Over-sampling
PRBS Generate
8B / 10B
From RX Parallel Data of Same Channel
TX PIPE Control
FPGA Logic
TX-PCS

Figure 3.19: Simplified block diagram of a Virtex-5 GTP transmitter and associated clock domains. Reproduced from [102], © 2009 Xilinx.

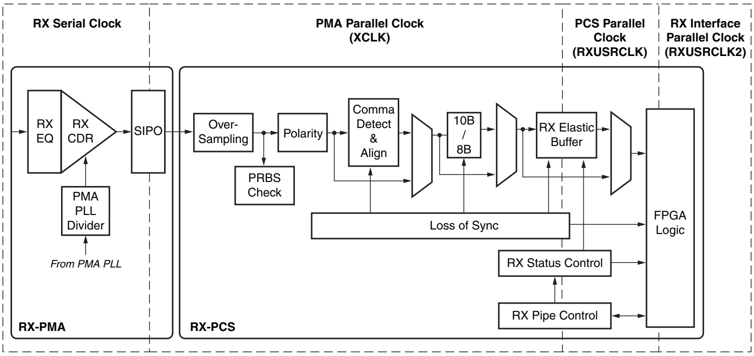


Figure 3.20: Simplified block diagram of a Virtex-5 GTP receiver and associated clock domains. Reproduced from [102], © 2009 Xilinx.

Figures 3.19 and 3.20 depict simplified block diagrams of the contents of a single GTP transceiver. Comparison with Fig. 3.8 reveals a strong similarity to the generic model. Transceiver latency is detailed in ([102], pp. 336–337), with uncertainties in the elastic buffer at the transmitter and the word aligner at the receiver; the elastic FIFO at the receiver appears to be deterministic, as reported in [135] and confirmed in ([102], p. 185) under stability conditions. In any case, both FIFOs may be bypassed as long as the device is configured in "TX phase alignment" and "RX phase alignment" mode, respectively, where GTP-generated parallel clocks are automatically phase-shifted in order to remove the need for adaptation stages.

To be more specific, in RX phase alignment mode, the recovered clock RXRECCLK (not shown) gets shifted with the assistance of a simple state machine implemented in user logic until the interface clock RXUSRCLK is aligned with the parallel clock

XCLK, which remains fixed. This alignment procedure assumes, as is explicitly stated in the documentation, that RXUSRCLK is externally forced to be driven by RXRECCLK; actually, this is not strictly necessary, which can be exploited for custom implementations [131]. If both clocks are indeed equal, then a register-to-register clock domain crossing appears at the receiver output between RXRECCLK and the local clock, with an equivalent latency (3.38).

On the other hand, in TX phase alignment mode it is the internal transmitter clock XCLK that gets shifted so that it matches the phase of the local clock TXUSRCLK. The physical transmission clock thus gets aligned with the user-provided transmit clock, which may be the same as the local clock, removing all clock domain crossings at the transmitter. The importance of this feature lies in the possibility of using the recovered clock RXRECCLK from the receiver directly to drive both the local clock for user logic *and* the transmit clock TXUSRCLK, thereby forcing XCLK to be aligned to the recovered clock, too. In that situation, there is no clock domain crossing at all in the node along the loopback path entering at the receiver and coming out of the transmitter. This means that one of the phase terms in (3.36) is guaranteed to be exactly zero by design; the other phase term still remains, as it is not possible to do this on both ends due to the clock dependency order as explained in §3.2.4. In any case, this greatly increases synchronization resolution, given that phase measurements are the largest contributors to link skew uncertainty.

Since the parallel receiver clock remains fixed after being randomly chosen during PLL convergence, the deserializer latency is variable, as was discussed in §3.3.1. Hence, the word alignment procedure needs to be taken into account. The receiver includes a comma detector and aligner for $m = 10$ bit wide words, designed with 8B/10B encoding in mind; it does not provide access to the amount of shifted bits, but it allows overriding automatic operation and implementing manual alignment with a bit slip interface. This makes it possible for the user logic to assist in word alignment and obtain the number n of bit slips while still benefiting from the 8B/10B infrastructure. In the GTP, bit slips are applied to the left and serial transmission is LSB first, so that n yields the static deserializer latency in terms of UI directly.

Taking all of the above considerations into account, the resulting expression for link skew using Xilinx GTP transceivers on both ends is

$$\Delta t_{\text{Xilinx}} = (n_{\text{S}} - n_{\text{M}})\,\frac{T_{\text{clk}}}{10} - \Delta\varphi_{\text{M}}\,\frac{T_{\text{clk}}}{2\pi} \tag{3.54}$$

where n_{M} and n_{S} are the bit slip counts in the master and slave receivers, respectively, and the only clock domain crossing occurs at the output of the master receiver.

Deterministic latency in Altera transceivers

Altera FPGAs, being roughly equivalent to their Xilinx counterparts in terms of functionality, density and performance with usually just a few months of difference, also started including Gigabit Transceiver Blocks (GXB) with the Stratix GX in 2004 [139]. Their block diagrams are similar to those of Xilinx transceivers, with a few differences in arrangement, e.g. Xilinx organizes GTPs in two-transceiver tiles while Altera uses four-channel blocks with some common clocking circuitry.

In particular, the Altera device that most closely matches the Virtex-5 is the Stratix II. Unfortunately, the GXB transceivers found in the Stratix II GX cannot be used for deterministic latency configurations. Elastic buffers are found in both the transmitter and the receiver as an interface with programmable user logic, and neither can be bypassed ([91], pp. 2-31–2-32 and 2-118–2-119); no mention is made in the documentation as to whether they have a predictable initialization. The situation is the same with its cheaper version, the Arria GX.

In order to implement precise synchronization over Altera transceivers, one must resort to newer, higher-end FPGA families, starting with the Stratix IV GX. In this device, it is possible to configure transceivers in a specific deterministic latency mode resulting in the block diagram shown in Fig. 3.21, using only serializers and deserializers, 8B/10B blocks, and elastic buffers (shown as "phase compensation FIFO") ([140], pp. 1-122–1-124). The latter are not completely bypassed but configured in register mode, using the same clock for writing and reading with a latency of exactly one cycle. At the receiver, this clock is the recovered clock `rx_clkout`. At the transmitter, the internal parallel clock $\mathrm{clk}_{\mathrm{TX}}$ necessarily comes from a PLL fed by an external reference clock; special circuitry is activated that forces a deterministic phase relationship between them. Word alignment can be achieved by externally assisted bit slipping or automatically, with a dedicated `rx_bitslipboundaryselectout` output that indicates the internal bit shift count.

The main difference between Altera and Xilinx transceivers for synchronization is related to the available clocking configurations. In a Virtex device, the recovered clock can be used (after jitter filtering) to drive a transmitter, and the phase alignment modes make it possible to have a single clock domain on one end, e.g. the slave. In a Stratix, however, the transmitter clock is necessarily derived from a special transceiver PLL fed by a reference clock that may come either from an input pin or from an embedded PLL; clock inputs for PLLs, in turn, may only come from dedicated input pins or from other PLLs. It follows that the transmitter clock has to come from an input pin at some point, and therefore cannot be the same as the recovered clock unless it is extracted from the FPGA and fed back in, possibly using additional external circuitry. Hence, the transmitter and recovered clocks are always different in each node, and link skew is equal to

$$\Delta t_{\,\mathrm{Altera}} = (n_{\mathrm{S}} - n_{\mathrm{M}})\,\frac{T_{\mathrm{clk}}}{10} + (\Delta\varphi_{\mathrm{S}} - \Delta\varphi_{\mathrm{M}})\,\frac{T_{\mathrm{clk}}}{2\pi} \tag{3.55}$$

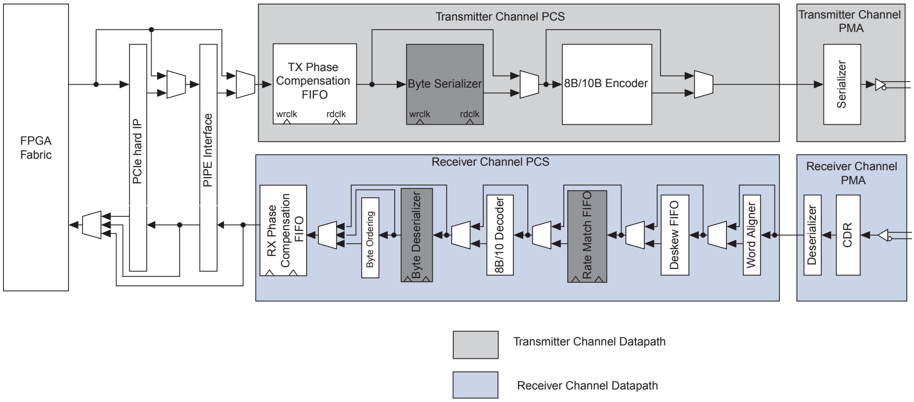


Figure 3.21: Block diagram of a Stratix IV GX transceiver configured in deterministic latency mode. Reproduced from [140], © 2015 Altera.

with the measurement of both $\Delta\varphi_{\mathrm{M}}$ and $\Delta\varphi_{\mathrm{S}}$ being necessary. This fact implies that the uncertainty that can be achieved with Altera devices is always worse than with Xilinx devices; specifically, the synchronization accuracy is expected to be roughly 40 % higher (a factor of $\sqrt{2}$) if the main contributions to uncertainty come from phase measurements and their resolution is approximately the same in both nodes.

3.4.2 Synchronization methods

The available references containing research on methods for precise clock distribution and/or synchronization methods can be divided into two categories. The first one describes the implementation of unidirectional data links with precise, deterministic latency, that allow completely synchronous operation of DAQs and triggers. The second one involves bidirectional links with the capability of precise pairwise synchronization between its end nodes.

Deterministic latency half-links

First of all, methods for the deployment of deterministic latency half-links over relatively long distances are described. These are fit for the design of completely synchronous DAQ systems or trigger subsystems, where latencies need to be precisely controlled in order to guarantee real time operation. These methods are usually simpler but carry a drawback: they allow neither the automatic synchronization of nodes nor the correction of delay variations, due to the fact that the deterministic transfer of information is only one-way.

- Aloisio et al. have worked on the establishment of serial half-links whose latency remains fixed even between resets or power cycles, mainly as a replacement for the obsolete Agilent G-Link [55] in LHC experiments. In [131], Virtex-5 GTPs are used to achieve deterministic latency on self-synchronous links within a system-synchronous setting, i.e. with independent distribution of a common clock similar to Fig. 3.5, by bypassing the receiver phase alignment circuitry and implementing it externally, or alternatively by taking advantage of the deterministic latency of their receiver elastic buffers. This solution is then adapted for the emulation of the G-Link [141], with a reported latency resolution of $\sigma = 30\,\text{ps}$.

  In [135], the scheme is extended to the implementation of fixed latency links without the need for external, independent clock distribution, and a resolution of $\sigma = 40\,\text{ps}$ is reported. A so-called *roulette* approach is proposed whereby the link is reset until a predetermined static latency is achieved. Unfortunately, the solution cannot be easily extended to bidirectional links without important additional considerations, despite what is claimed in the paper. It also fails to take into account the drift in propagation latency.

- Liu et al. [142] modify this scheme by adding a variable delay element in user logic that compensates for static latency variations in the receiver, avoiding the need to reset the link. The circuit covers various overlapping subcases that together result in valid operation over the whole latency variation range, and the reported performance is similar to [135]. Again, the solution cannot be extended as is to bidirectional links because it relies on the lack of clock domain crossings with non-zero phase differences.

- A clock distribution system over optical data links for the PANDA experiment is proposed in [73] where two different transceivers are used to receive the same incoming serial data in a Lattice FPGA. One of them is used as a reference and the second one is continuously reset, measuring the latency difference between them by comparing the respective output streams. Ten values are possible, corresponding to ten divided parallel clocks. When all of them have been detected, the second receiver is reset until a specific value is achieved, and its recovered clock is used. The main disadvantage of this method is that is forces the use of two receivers per half-link, besides potentially incurring in signal integrity issues due to the point-to-multipoint topology. In a later paper [74], the implementation is reported to have changed to a Virtex-5 using only one receiver and continuous resets, presumably the same method as [135] and obtaining similar results.

- Lemke et al. [134] propose an entire DAQ interconnection and clock distribution network based on point-to-point links consisting of two independent deterministic latency half-links using Virtex-4 MGTs. Deterministic latency is achieved, once again, by resetting the link until a particular bit shift value is taken. Matched length fibers are used in order to guarantee phase-aligned recovered clocks, and exact latency values are known by previous calibration. It is claimed, but not tested, that this method can lead to automatic latency measurement using pairwise synchronization; of course, this cannot be the case unless phase measurements are implemented or resolutions of $T_{\mathrm{clk}}/2$ are tolerated.

It is worth noting that most of these examples are based on restricting the set of valid divided clocks at the receiver to just one and rely on repeated link resets until this predetermined condition is met. Such implementations are therefore not real-time, because there is no strict bound on link establishment time as the repeated reset procedure could theoretically go on indefinitely, particularly if the probability of success for each try is as low as 10 %.

Self-calibrated bidirectional links

Methods for the implementation of deterministic bidirectional links are described next. These allow the synchronization of timestamp counters between its ends, and the automatic correction of link delay variations. They are thus fit for the deployment of triggerless DAQ systems, where exact latency values are not important as long as timestamping is precise.

- The most important reference in the field is probably the White Rabbit project at CERN. This standard defines extensions to Gigabit Ethernet networks that allow them to distribute timing over the switched network [143]; namely, syntonization is achieved according to the Synchronous Ethernet specification [144] and PTP is extended to support precise pairwise synchronization. A multimode optical fiber is used for both half-links with different wavelengths, implying that $t_{\mathrm{p,MS}} \neq t_{\mathrm{p,SM}}$ but rather
$$\frac{t_{\mathrm{p,MS}}}{t_{\mathrm{p,SM}}} = 1 + \alpha \tag{3.56}$$
where α is a physical parameter that depends on the refractive indexes of the fiber at both wavelengths. The analysis performed in §3.3 needs to be repeated under this condition; precise synchronization is possible, but in this case the transceiver latencies $t_{\mathrm{TX,M}}$, $t_{\mathrm{TX,S}}$, $t_{\mathrm{RX,M}}$, $t_{\mathrm{RX,S}}$ have to be known, rather than just their pairwise differences. This is achieved by physically forcing parallel clock waveforms on the external transmission media and measuring the phase difference between the serial signal and the corresponding parallel clock using DDMTD. Phase knowledge is then used to shift the slave clock in order to align it with the master. The White Rabbit approach enables the use of any transceiver with static latencies, but requires more phase measurements which decreases resolution. An additional drawback is that the parameter α itself has a non-zero temperature coefficient,

so the variations due to temperature drift are not completely compensated as reported in [104].

- Shang et al. [145] develop a clock distribution and time synchronization system over self-synchronous links for the LHAASO, a large-scale cosmic ray detector array. The design is based on the White Rabbit project but removes several unnecessary features. The main difference is in the implementation of phase measurement at the master, where DDMTD is used and edges in the sampled bit stream are detected instead of longer sequences; as a result, metastability and jitter result in glitches that hinder phase estimation, as analyzed in §3.3.2. The authors propose reducing the stretching factor N of DDMTD in order to reduce the glitching, and compensating for the resulting loss in phase resolution by measuring the distance between edges using a TDC. This approach seems completely misguided, as said distance is always an integer number of DDMTD clock cycles so its measurement cannot possibly be improved.

- An early attempt at link delay calibration over a bidirectional link is described in [103] for the alignment of clock phases in timestamping nodes in the AGATA detector. Link latency using Virtex-II Pro MGTs is treated as a random variable (incorrectly labeling it as uniform, when in fact its static variations are given by (3.37)), the random distribution of the round-trip time RTT is deduced, and a histogram is obtained by repeatedly resetting and measuring the link; after that, the RTT values where the link is approximately symmetrical are identified and the link is reset again until one of those values is obtained. This method is essentially a crude version of the *roulette* approach proposed in [135]. Unfortunately, it is extremely time-consuming and fails to take into account the link skew induced by phase differences, resulting in a resolution of $T_\mathrm{b}/2 = 500\,\mathrm{ps}$. Experimental results are reported in [146] that confirm this accuracy value.

- An improved clock alignment method for AGATA is described in [147], wherein optical multiplexers are included at each transceiver input and output that allow their manual bypassing. The round-trip time of the external propagation media is measured with a TDC at the master by bypassing the transceivers and injecting a pulse into the loopback path. The same measurement is repeated with the downlink transceivers included in the path. Together, these values yield the exact flight time in each direction, and programmable delays can be inserted at the slave in order to force phase alignment with the master. Alignment within 330 ps is reported without compensating for the latency of combinational logic in FPGAs.

- Rohlev et al. [136] propose a bidirectional link with Virtex-5 GTPs for the recovery of aligned clocks and one-way latency measurement for clock and timing distribution for the FERMI linac. The loop filter of the PLL handling the recovered clock is implemented in the FPGA, and synchronization is achieved by using remote loopback and delaying the recovered clock until a full clock cycle step is detected at the PLL. The scheme is similar to the one proposed in this thesis in section §3.5.1, but it forces the use of two separate GTPs in the slave for transmission and reception; workarounds for this drawback will be described later.

- In [148], the clock distribution scheme described in [134] for the CBM experiment is extended to the communication with front-end ASICs at a 250 MHz bit clock using two deterministic latency half-links. Recovered clock alignment to within $T_\mathrm{b}/2 = 2\,\mathrm{ns}$ is reported using pairwise synchronization with no wait time on the slave and selectively delaying transmission on the master. It is unclear and unexplained how this is achieved without phase measurements; a likely explanation is that the low bit rate allows estimation using synchronous logic at 250 MHz on the master side, which would be consistent with the low resolution values reported therein.

- Papakonstantinou et al. [149] propose a bidirectional point-to-multipoint passive optical network as an upgrade for the distribution of timing, trigger and control signals at the LHC. Deterministic latency links are implemented using Virtex-5 and Spartan-6 GTPs with adjustable delays for the recovered clocks. The latency of the optical link is measured by introducing a fiber loopback at one end and measuring the round-trip time at the transmitter, in order to monitor latency drifts; these are then used to instruct the slave nodes to modify their programmable delays dynamically in order to maintain phase alignment. Unfortunately, the last section of the fiber propagation delay (from splitter to end nodes) is not taken into account and a small error is introduced. An extension is proposed implementing the loopback at the end nodes, but this is incompatible with the multiplexed scheme for upstream communication, and it is also prone to the errors described in [104] due to different wavelengths.

- A somewhat unique approach is taken by Hidvégi et al. [150] for precise clock distribution at the XFEL installations. A unidirectional self-synchronous link prototyped using discrete parts is looped back at the slave node and includes two programmable delay lines at the master node: one at the transmitter and one at the receiver. A control loop is established wherein the phase difference between local and recovered clocks acts on the delay lines, forcing identical delays on them until the phase difference converges to zero. At this point, the round-trip time is an integer number of clock cycles and the link is completely symmetrical by design, so the one-way latency is exactly $RTT/2$. The clock phase at the receiver can take two possible values depending on the parity of RTT but this is easily accounted for. This scheme automatically compensates for drifts due to temperature variations and achieves approximately 20 ps peak-to-peak skew between the master and slave clocks over distances of several km at the expense of significantly increased external circuitry.

- A hierarchic clock distribution system for ALICE based on source-synchronous data links is described in [151] where the received clock is transmitted back to the master, phase differences are estimated using DDMTD and the result is used to phase-shift the slave clock in order to align it with the master clock. Skews under 200 ps are reported. An extension to self-synchronous optical links is proposed, which is essentially a modified version of the method proposed in this thesis that forces alignment of the master and slave clocks instead of compensating for the phase mismatch.

- Anvar et al. [77, 152] report a synchronization scheme for the KM3NeT neutrino telescope that follows essentially the same approach proposed in this thesis and described in section §3.5.1, with few differences in implementation. Bidirectional data links are deployed using Xilinx transceivers with variable but deterministic latency and a single clock domain crossing along the loopback path whose phase shift is measured with DDMTD. Timestamp counters in different nodes are not aligned but rather corrected offline after acquisition.

All of these methods are summarized in Table 3.2, together with the proposal presented in this thesis in section in section §3.5.1, and the main features of each reference are stated. Most methods actively force the alignment of slave clocks, either with the master clock or between each other; this is required in installations that need completely synchronous operation such as accelerators, but optional for simple triggerless acquisition. Some methods do not require the use of deterministic latency transceivers as described in §3.4.1 and can use standalone devices. Finally, only a few of the described methods support a continuous refresh of the synchronization method over the data link without interrupting it (except for the transmission of synchronization frames). Those who don't either require a dedicated synchronization link, or rely on a single initial calibration and are therefore unable to compensate for latency drifts. Reported peak-to-peak synchronization accuracy values are also given; however, comparisons between them need to take into account the large spread in physical link conditions, ranging from small cables to kilometer-long fibers.

Project	Reference	Accuracy (ps)	Needs DL transceiver	Seamless data link	Aligns clock	Notes
White Rabbit	[143]	160	No	No	Yes	Drifts with temperature
LHAASO	[145]	100	No	Unclear	Yes	
AGATA (early)	[103]	500	No	No	Yes	Long lock time
AGATA	[147]	330	No	No	Yes	
FERMI	[136]	Unclear	No	No	Yes	Double transceivers
CBM	[148]	2000	Yes	No	Yes	Low data rate
LHC PON	[149]	300	Yes	Yes	Yes	Asymmetry errors
XFEL	[150]	20	No	Yes	Yes	
ALICE DTCC	[151]	200	Yes	No	Yes	
KM3NeT	[77]	800	Yes	Yes	No	Offline correction
This work		120	Yes	Yes	No	Online correction

Table 3.2: Summary of existing precise synchronization schemes over bidirectional data links. DL stands for *deterministic latency*.

3.5 Proposed synchronization scheme

Now that the related concepts and the state of the art have been fully described in the preceding sections, the main proposal made in this chapter can be stated. It consists in a scheme for clock distribution, syntonization and internal synchronization of all nodes within a DAQ system. The scheme is based on point-to-point self-synchronous data links, where one of the end nodes takes the role of master and the other one is the slave. The clock and time reference in the master are considered fixed, and the slave adjusts them, i.e. it recovers the master clock frequency and modifies its time reference so that it matches the master's. As mentioned in §3.2.1, this can be extended to the whole DAQ system by defining a spanning tree hierarchy, as long as any two nodes in it can be connected by a path containing only point-to-point links working as described.

The main idea behind the proposal is to use fractional timestamps natively as part of the local reference, i.e. timestamps that feature both an integer part, given and updated by local timestamp counters, and a fractional part that accounts for the timestamp clock phase and is handled exclusively by the synchronization algorithm. Of course, non-integer timestamps are already used to tag events in DAQ systems whenever a time resolution finer than T_{clk} is desired, either by performing an analysis of sampled waveforms and extracting a fine time mark or by using digitizers with finer granularity such as TDCs on a local level. However, non-integer timestamps are only used to tag events, i.e. as the time reference of *signals*, using the terminology of §3.1.3. The novel approach that is proposed here is to use them also for the time reference of *nodes*, i.e. to include a non-integer part explicitly in the timestamp counters.

In order to see how this concept is intimately related to the alignment of clocks between nodes, consider the situation where the local clocks on all relevant nodes are syntonized and phase-aligned. In that case, a global phase reference has been established system-wide, and all timestamps can be thought of as being referred to it. The fractional part of the timestamp exists implicitly, but it is exactly the same at all nodes and so it can be regarded as being equal to any reference value like zero, and there is no need to store or keep track of it explicitly. Hence, in synchronization schemes that force clock alignment, there is no need for fractional timestamp counters.

However, this relationship also works the other way around: in a scheme that does not use fractional timestamps for local time references, it is not possible to synchronize nodes that are not phase-aligned, at least at lower levels (correction at the statistical level is possible, as shown e.g. by [77]). While the alignment of local clocks is strictly required in some systems e.g. for control of accelerator and fusion facilities, it is arguably not needed for DAQ systems that provide fine timestamping for signals anyway, especially in triggerless systems where constant latency between nodes is never required. In these systems, it is the synchronization scheme itself that constrains the system by needlessly imposing the condition of clock alignment to it. It is argued in this dissertation that the native use of fractional timestamps for local references removes that restriction; it is equivalent to transferring the management of phase information from the *physical* level (the local clock phase) to the *logical* level (the timestamp counter).

Practically all existing methods for precise synchronization, and in particular those reviewed in §3.4.2, split the synchronization process into two steps, *coarse* and *fine synchronization*:

- The coarse synchronization step consists in the alignment of the timestamp counter values exclusively, and typically amounts to the pairwise synchronization algorithm presented in §3.2 with integer timestamps, with a resolution in the order of T_{clk}.
- Fine synchronization considers and measures the phases of the timestamp clocks at both ends in order to improve the resolution obtained by the coarse step, and usually adjusts the clock phases so that they match.

In most implementations, these two steps are considered as completely separate and implemented independently, as they act on different elements (the timestamp counters on user logic and the clock phases, respectively). Notice, however, that the pairwise synchronization algorithm as described in §3.2 never assumes a particular granularity in the timestamp values used for the computations. Local timestamps with a fractional part can therefore be used without restriction. In fact, by lifting the restriction of aligned phases and embedding the phase information into the timestamp counter as proposed here, the algorithm actually implements both the coarse and the fine synchronization steps simultaneously.

As was already shown in §3.2.3, the resulting algorithm is optimal for the two-node subsystem (in the Lundelius-Lynch sense), and results in a synchronization accuracy equal to half the accuracy in the estimation of link skew. By using deterministic latency transceivers, this value is given by the resolution of phase difference measurements between local and recovered clocks in a FPGA, either directly in Xilinx devices or multiplied by $\sqrt{2}$ in Altera devices, as follows from (3.54) and (3.55), respectively. If the DDMTD method with clustering post-processing is used for phase estimations as presented in §3.3.2, said resolution can be optimized by choosing a suitable value for the learning rate θ of the clustering step.

The main features of the proposed method, as it relates to the state of the art outlined in §3.4.2, are the following:

- It syntonizes the clocks at both ends of a data link without requiring a dedicated clock distribution network.
- It is a precise synchronization method, i.e. it achieves synchronization accuracies below T_{clk}, in the order of 100 ps.
- It is based on self-calibrated bidirectional links instead of deterministic latency half-links. It therefore does not guarantee the same latency in a given link for each power cycle. One-way latencies can however be computed as a byproduct of the pairwise synchronization algorithm.
- It automatically compensates for drift in the propagation delay of transmission media and is thus apt for implementation over long optical fibers.
- It does not force the alignment of clocks on both ends of each synchronization link.
- It works seamlessly on the data links, in the sense that it does not disturb them except for the transmission of synchronization frames, and even so they take a negligible fraction of the available data bandwidth.

A proof-of-principle implementation of the proposed method on FPGA evaluation boards is described in this section, together with experimental measurements that prove its feasibility and give an estimation of the expected synchronization resolution. Unfortunately, the described implementation is unable to give a precise measure of the achieved resolution due to the limitations of the employed evaluation boards. Accurate characterization is therefore postponed until chapter §6.

3.5.1 Proof-of-principle implementation

In order to validate the proposed synchronization scheme, an implementation was realized on two Xilinx ML505 evaluation boards [153] and tested during the year 2010. The description of the setup and method and the experimental validation were published in [7].

Each ML505 board hosts a Virtex 5 LXT device, specifically a XC5VLX50T, with several GTP transceivers available to the user. In particular, the differential inputs and outputs of one of the transceivers are accessible through on-board SMA connectors. The boards also include a 156.25 MHz oscillator that is used as the base clock frequency f_{clk} for the timestamp counter, local logic and word clock. The data links are 8B/10B-encoded, for a physical transmission speed of 1.5625 Gbps and a net data rate of 1.25 Gbps.

The goal is to achieve syntonization and synchronization using only the four GTP input/output wires to connect both boards. Moreover, the link should be self-calibrated i.e. no external calibration elements should need to be used, and it should be able to automatically compensate for latency variation due to environmental drift.

In order to guarantee the lowest possible uncertainty in link skew, the firmware was designed to allow the use of the same FPGA programming file in both nodes, so latencies and propagation delays between embedded elements are matched. An on-board switch was used to select between master and slave mode by controlling clock multiplexers and other logic inside the FPGA.

Clocking scheme

In Virtex-5 devices, GTP transceivers are arranged into blocks called “tiles”, i.e. groups of two GTPs with some common clocking circuitry. This is a typical feature of FPGA-embedded transceivers as a means to reduce circuit size and external pin count; for instance, Altera GXBs are arranged into blocks of four transceivers. Transceivers in the same block, in particular, share the reference clock input for their CRUs, and thus the working frequencies of both members of the tile must be related by a rational factor. The reference clock is used to directly generate the serial transmit clock for the transceivers it feeds.

The goal at the slave node is to recover the clock embedded in the incoming data stream and use it locally. Moreover, for pairwise synchronization to yield the desired accuracy, the slave transmit clock $\mathrm{clk}_{\mathrm{TX}}$ must be syntonized with the recovered clock $\mathrm{clk}_{\mathrm{REC}}$. Hence, the slave transmitter must be located in a tile whose reference clock input $\mathrm{clk}_{\mathrm{REF,TX}}$ is derived from the recovered clock.[12] However, in order to obtain the recovered clock, a reference clock $\mathrm{clk}_{\mathrm{REF,RX}}$ for the slave receiver is needed in the first place. In terms of clock dependency,

$$\mathrm{clk}_{\mathrm{REF,RX}} \prec \mathrm{clk}_{\mathrm{REC}} \prec \mathrm{clk}_{\mathrm{REF,TX}} \prec \mathrm{clk}_{\mathrm{TX}}. \tag{3.57}$$

It follows that $\mathrm{clk}_{\mathrm{REF,RX}}$ and $\mathrm{clk}_{\mathrm{REF,TX}}$ cannot be *exactly* the same.

One possible solution is to use separate transceivers for transmission and reception at the slave node that are located in different tiles, and feed the transmitter reference clock with the receiver recovered clock, as described e.g. in [136]. This implementation is likely to imply a waste of GTP resources, using only one half of some transceiver blocks. In any case, it is impossible to realize it in the ML505 boards because they only have available connectors for the inputs and outputs of a single transceiver, making it mandatory to use that single transceiver for both half-links.

A more sophisticated solution is proposed in [154] and adapted here: including a special clocking circuit for the reference clock input that can be dynamically switched after the reception half-link is established without losing lock. It is based on the use of a *voltage-controlled crystal oscillator* (VCXO), i.e. a narrowband VCO whose frequency variation range in the same order of a typical oscillator's tolerance, and thus can be used to track another oscillator's exact frequency while maintaining its nominal value. The idea is to drive the reference input with the VCXO while simultaneously forcing it to match the recovered clock by means of a PLL. Changes in the VCXO output frequency during alignment to not disturb receiver operation because it is only used as a reference and its frequency remains in range; in particular, its output phase remains continuous.

The full clocking scheme for the master and slave nodes is depicted in Fig. 3.22 (top and bottom, respectively). The same firmware is used in both cases, and the unused elements are shaded in gray in each schematic. Clock multiplexers (`BUFGMUX`) are used at various points in order to select master or slave mode operation. The recovered clock `RXRECCLK` and the reference input `CLKIN` are used as input to a digital PLL whose PFD and loop filter are implemented digitally inside the FPGA. A small external mezzanine board is connected to the ML505 at the slave node that contains a 156.25 MHz VCXO (Connor-Winfield V702) and a 16-bit, 180 kSPS DAC (Texas Instruments DAC8571 [155]). The VCXO output is filtered by one of the PLLs embedded in the FPGA before feeding the GTP. The VCXO is initially kept at its center frequency, and frequency locking starts when the recovered clock becomes available.

The implementation of the digital PLL in the FPGA follows the recommendations from [156], using a PI i.e. proportional-integral loop filter with programmable

[12] Unless there is a separate common clock distribution network where the reference can be obtained from, but this goes against the stated wiring goals of the proposed scheme.

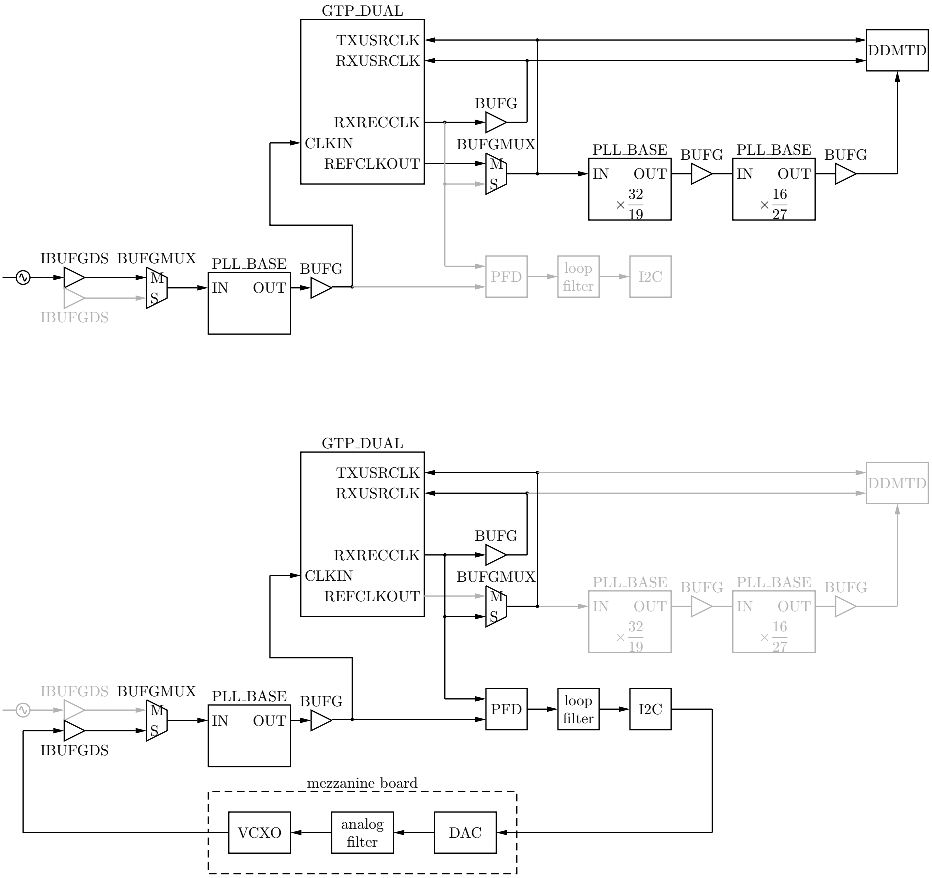


Figure 3.22: Transceiver clocking scheme used in the ML505 implementation at the master (top) and slave (bottom) nodes. The same FPGA firmware is used in both nodes; in clock multiplexers, M and S denote the clock inputs used for the master and slave nodes, respectively. The mezzanine board is only present at the slave.

coefficients. Digital versions of several types of PFD were implemented for test: an Alexander detector [157], a Hogge detector ([124], p. 46), and a classic PFD. Many combinations of coefficients and detectors were tested experimentally in order to obtain optimal parameters that guaranteed PLL lock in all cases within a reasonable time. In the end, the Alexander detector was chosen.

It must be noted that jitter requirements for CRU clock references are usually tight in most transceivers, e.g. 40 ps total jitter for the Virtex-5 GTP ([158], p. 16), whereas recovered clocks have been shown to exhibit typical jitter values that are higher by an order of magnitude [95]. Hence, jitter cleaning circuitry is often mandatory for reference clock filtering before it enters the GTP. In this particular case, the recovered clock is processed by a PLL with a fully digital phase detector, which also results in a VCXO output with relatively high jitter ([124], pp. 48–49). The reason is related to the operating principle of all-digital phase detectors, which do not fully converge but instead generate up and down pulses to track the input phase even in its locked state. This causes the PFD output to oscillate around zero, which after the loop filter turns into a small ripple at the DAC or VCO input and jitter at its output. Linear phase detectors avoid this problem by virtue of featuring a stable, non-quantized output in the locked state. A detailed analysis of jitter in these types of phase detectors can be found at [159]. It was experimentally verified that the amount of residual jitter in the Alexander detector used in the proposed implementation was non-trivial, and that the additional embedded PLL shown in Fig. 3.22 (just above the VCXO) was indeed necessary for reliable operation.

The DDMTD method is used for measurement of the phase difference between the local and recovered clocks. A stretching factor $N = 512$ is implemented by synthesizing the DDMTD clock from the local clock using two cascaded PLLs with frequency multiplication factors 32/19 and 16/27, as shown in Fig. 3.22, yielding raw measurements with granularity 12.5 ps. These raw measurements are filtered with the clustering algorithm using a learning rate $\theta = 1/256$ and generating estimates with 12-bit precision. In order not to introduce systematic errors in phase measurement, both clocks under measure go through clock buffers/multiplexers with similar delays; these buffers and the DDMTD sampling flip-flops are manually placed close together, and the corresponding routing delays are verified to be nearly identical.

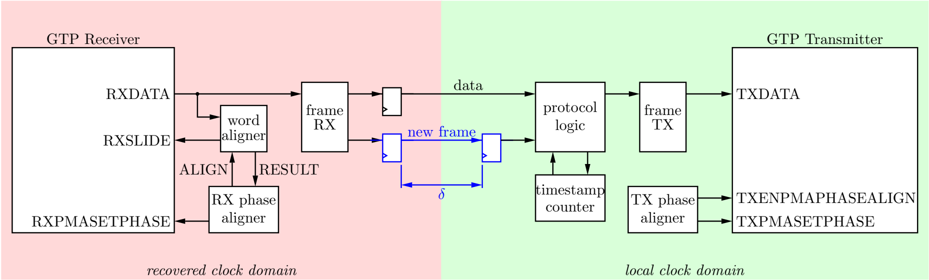


Figure 3.23: Block diagram of the modules used for link establishment and pairwise synchronization implemented in user logic, separated into clock domains. The critical clock-domain-crossing path is highlighted in blue.

Link establishment

The data link between the master and slave nodes is self-synchronous, based on byte transmission using 8B/10B line coding. It must therefore be continuously transmitting data, at least in the master to slave direction, in order to maintain syntonization. For this proof-of-principle implementation, the link is always transmitting frames with synchronization information, as opposed to a realistic situation where the rate of synchronization messages is ideally as low as possible in order not to take up too much link bandwidth.

Fig. 3.23 shows a block diagram of the contents of the firmware implemented in the FPGAs for the self-calibrated data link. These modules are split into a receiver and a transmitter section, clocked by the recovered and the transmitter clock respectively; recall that this distinction only applies at the master, as both clock domains end up being the same at the slave.

Simple FSMs are used to implement the TX and RX phase alignment procedures described in §3.4.1 that allow bypassing the elastic buffers in the GTP data path. A word alignment module is also present that detects the self-synchronizing K.28.5 comma symbols within the deserialized stream and controls the bit slip interface (`RXSLIDE`) of the GTP receiver. The RX phase alignment and word alignment operations are intertwined; the joint procedure is as follows: RX phase alignment is carried out first. The word aligner block then analyzes a long string of consecutive received bytes and looks for comma characters. If it finds at least 64 commas within the next 1024 bytes, the half-link is considered as locked. Otherwise, a bit slip is issued and RX phase alignment and comma search are repeated. The operation continues indefinitely until word alignment is achieved.

The logic implementing the synchronization algorithm is located at the transmitter clock domain, and is based around a 48-bit timestamp counter with 36 integer and 12 fractional bits. At a clock period of $T_{\mathrm{clk}} = 6.4\,\mathrm{ns}$, this implies a granularity

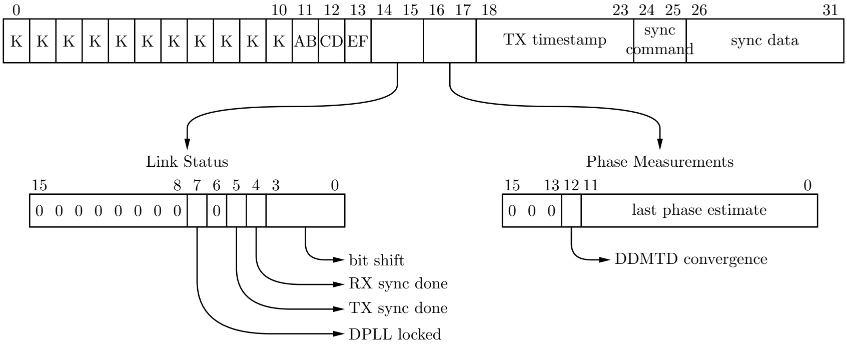


Figure 3.24: Frame format for the communication protocol between the ML505 boards. K stands for the 8B/10B comma symbol K.28.5.

below 1.6 ps and a rollover period over 7 min. Frame receiver and transmitter modules are also included that encode and decode the appropriate data into the correct frame structure. These modules are also responsible for assigning and storing the frame departure and arrival timestamps at specific, deterministic delays with respect to the frame boundaries.

The frame format used for the tests is shown in Fig. 3.24. It contains a string of comma symbols (enough to pad the frame up to 32 bytes) that are used for word alignment at the receiver end. A 3-byte header containing a specific signature is also included to ease the detection of invalid operation at the receiver. A 16-bit Link Status field comes next, with several control bits indicating the current status of the link establishment procedure:

- One bit flagging completion of the phase alignment procedure at the transmitter.
- One bit flagging completion of phase and word alignment at the receiver.
- One bit indicating the lock status of the digital PLL (at the slave).
- A 4-bit field containing the local bit shift value, i.e. n_M or n_S from (3.54).

Other bits in this field are either set to zero or used for other unrelated test purposes. The remaining 16 bytes contain the payload used by the pairwise synchronization algorithm; their structure will be detailed later.

The detailed procedure for link establishment acts on the elements in the loopback path sequentially, starting at the master transmitter, then the slave receiver, the slave transmitter and finally the master receiver, as follows:

1. The TX phase alignment procedure is carried out at the master. When finished, the corresponding bit is activated in transmitted frames in order to let the slave know.

2. Joint RX phase alignment and word alignment is performed at the slave. It ends whenever word alignment is achieved *and* the received frames, which are then decoded and aligned, indicate correct TX phase alignment at the master.

3. The digital PLL is activated and the link waits until it locks, i.e. the reference clock input of the slave GTP is syntonized with the recovered clock.

4. TX phase alignment is then executed at the slave.

5. The joint RX phase alignment and word alignment procedure is finally carried out at the master. It goes on until alignment is achieved and control bits in the received frames indicate that all other conditions have already been fulfilled, i.e. alignment and PLL convergence at the slave.

Once these five steps are completed, data synchronization between both nodes is fully established, and the time synchronization procedure may begin.

Synchronization protocol

The last 16 bytes in the frames described in Fig. 3.24 contain the data needed for the time synchronization algorithm. The first fields in this subframe are common to master and slave and are automatically filled by the frame builder; they contain the 48-bit departure timestamp assigned to the frame, the last 12-bit phase estimate obtained by DDMTD, and a flag indicating the convergence of the clustering algorithm used for phase estimation refinement. The final 8 bytes contain a 16-bit field that identifies the specific command issued by the protocol logic and a 48-bit field that contains associated data to be transferred.

The algorithm implemented by the synchronization logic and the associated state diagrams are outlined in Fig. 3.25. The synchronization procedure is initiated by the master once every $T_{\mathrm{resync}} = 1\,\mathrm{s}$. A total of three frames are issued, identified by values 1 to 3 in the command field, whereas the value 0 is used for empty/idle frames. The first two correspond to the usual two-frame pairwise synchronization method indicated in Fig. 3.11; the value of T_{S1} is piggybacked in the data field of the second frame. After this frame is received, the master has all necessary data for the computation of ΔT_{S} according to (3.30): T_{M1} and T_{M2} were stored by the master, T_{S1} and T_{S2} are included in the second frame, and Δt can be obtained as (3.54) from n_{M} and $\Delta\varphi$, which are measured locally, and n_{S}, which is included in the frame. A third frame is finally sent to the slave containing the value of ΔT_{S}

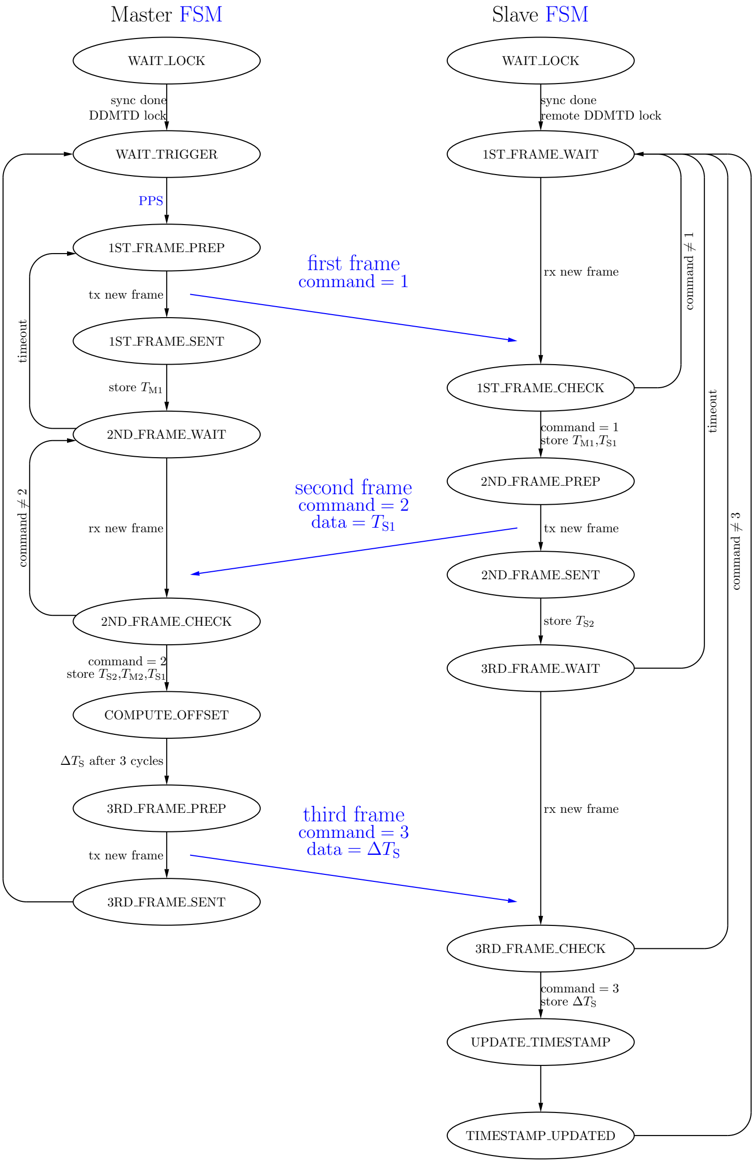


Figure 3.25: State diagrams of the pairwise synchronization algorithm FSMs for the master and slave nodes, with details of the involved frames highlighted in blue.

and the slave timestamp counter is corrected correspondingly. A timeout is also implemented for each state dedicated to waiting for a response frame from the other node in order to avoid hang-ups due to lost messages.

Using this scheme with just two nodes, only the slave timestamp counter gets updated by the algorithm and hence the fractional part of the master timestamp counter remains fixed at zero. The fractional part of the slave timestamp therefore yields a direct measure of the phase difference between the local clocks on the master and slave nodes.

Stability of timestamp synchronization

The implementation of phase measurement generates estimates of the phase error $\Delta\varphi$ modulo T_{clk}. Since the estimation method has a finite resolution, the resulting estimates will showcase a non-zero fluctuation around the center value. This creates a potential for error in the timestamp synchronization algorithm whenever the real value of $\Delta\varphi$ is close to the period crossing values. For example, suppose that the phase difference is positive but very close to zero. It is then possible for some estimates of $\Delta\varphi$ to fluctuate below zero and be detected as $T_{\text{clk}} - \epsilon$ for small, positive ϵ. If one of those estimates is read and used by the timestamp synchronization algorithm, the resulting value of Δt will have an error close to T_{clk} and thus the slave time reference will be updated with a synchronization error of about $T_{\text{clk}}/2$. Moreover, this error may appear intermittently as successive iterations of the synchronization algorithm take values of $\Delta\varphi$ with alternating sign.

The simplest solution to this issue is to avoid phase differences close to zero by establishing guard intervals around the period crossing points, regarding the link as unsuccessful whenever $\Delta\varphi$ falls within the specified guard range and forcing link reset until the value of $\Delta\varphi$ is safe. Note that this is similar to the *roulette* approach described in §3.4.2 for deterministic latency half-links in that it implies the rejection of some of the possible initialized states; it is therefore not a real time approach and it increases the average link establishment time, depending on the size of the guard intervals. Of course, the effect is lessened in this case as the probability of success is far higher than 10 %, and depends on the size of the guard interval.

Clock domain crossing[13]

There are two different clock domains at the master side, one for the receiver logic and another one for the local and transmitter logic. All received data therefore need to cross domains at some point, either through a "safe" interface such as an elastic buffer or as raw register-to-register transfers. Since the use of elastic buffers introduces a latency uncertainty element, any latency-critical transfers need to avoid them. In particular, for accurate assignment of arrival timestamps to synchronization frames, it is required to have a minimum of one handshake signal perform a raw register-to-register clock domain crossing.

The implementation shown in Fig. 3.23 achieves that minimum value by using a "new frame" indicator as a domain-crossing handshake signal, highlighted in blue. A rising edge appears in this signal whenever a new frame is completely decoded, with a deterministic latency relationship with respect to its reception. It is then used at the local clock domain to latch the timestamp counter value, i.e. to assign the arrival timestamps.

All other data within received frames are latched at the receiver clock domain, however they are read later by the protocol logic. The latches hold their value for several consecutive clock cycles, therefore there is no risk of metastability when reading them later even if the reading is done asynchronously between clock domains; the handling of the handshake signal guarantees that their contents are stable.

To see how this handshaking mechanism creates a potential for latency uncertainty, consider the timing diagrams depicted in Fig. 3.26, detailing the propagation delay δ between the output `new_frame.Q` of the flip-flop containing the new frame flag at the receiver domain and the input `new_frame_sync.D` of the flip-flop sampling it at the transmitter domain. Two cases are shown in Fig. 3.26, corresponding to a frame being received at different points within the same transmitter clock cycle where the local timestamp counter holds the value 1:

- In the first one, the rising edge arrives at the sampling flip-flop before the active clock edge, and the arrival timestamp is assigned with a latency of 1.
- In the second case, however, the rising edge in `new_frame_sync.D` arrives late, and the timestamp gets assigned with a latency of 2.

[13] This is a fundamental issue that was not covered in the paper [7] because it was discovered shortly after the submission. The author attributes this mistake to the fact that the related error can be removed using essentially the same mechanism already in use for phase stability: the problem could not be observed while its solution was already in place.

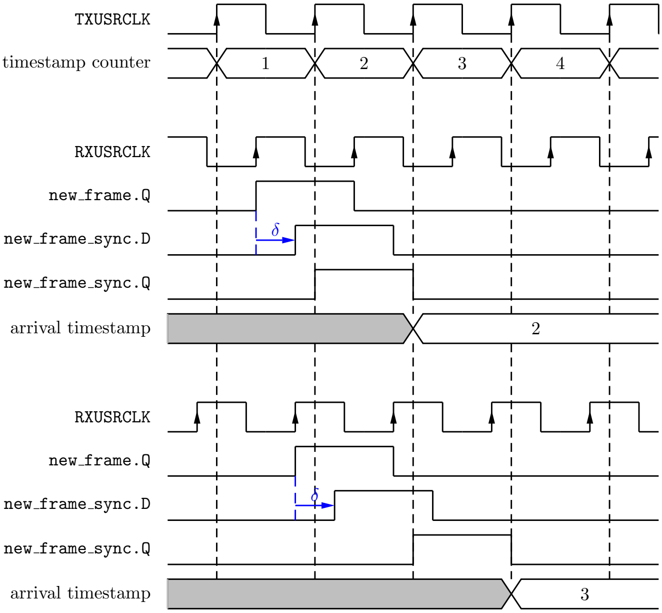


Figure 3.26: Timing diagrams for the assignment of arrival timestamps at the master node depending on the phase difference between the receiver and transmitter clocks.

There is thus an indeterminacy that depends on the phase difference between the receiver and the transmitter clocks. More specifically, if $\delta > \Delta\varphi$ then there is a cycle slip and the timestamp assignment latency is one cycle higher than if $\delta < \Delta\varphi$.

In order to resolve this indeterminacy, the relationship between δ and $\Delta\varphi$ needs to be known. Unfortunately, direct comparison between δ and $\Delta\varphi$ presents practical problems: the real value of δ can only be estimated beforehand, is difficult to calibrate, and fluctuates with temperature, whereas the measurement of $\Delta\varphi$ has a finite resolution. A better solution is thus to expand the guard interval of forbidden values of $\Delta\varphi$ to include the delay δ. It is possible to guarantee a relatively small maximum delay value δ_{max} for δ in the firmware via timing constraints, as the corresponding path only contains flip-flop to flip-flop routing with no combinatorial logic in between. A guard interval $[-\epsilon, \delta_{\mathrm{max}} + \epsilon]$ of invalid phase differences may then be established. Any value of $\Delta\varphi$ outside this interval will result in a constant, stable latency.

It is worth noting that $\Delta\varphi \equiv 0$ at the slave, hence the assignment of arrival timestamps will be working in the opposite situation in each node, having its latency increased by one cycle at the slave. In the implementation presented here, this is accounted for by delaying the `new_frame_sync` signal an additional clock cycle at the master, i.e. using a second flip-flop in series.

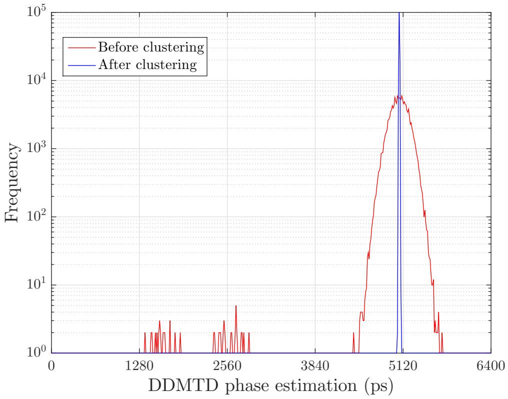


Figure 3.27: Logarithmic histograms (hits per bin increased by 1) of DDMTD phase estimates for a single link session, before (red) and after (blue) the clustering algorithm.

3.5.2 Experimental results

Several measurement runs were performed in order to validate the synchronization method and to obtain an estimation of the attainable resolution. An RS-232 interface was included in the FPGA firmware that continuously sent the latest estimations of important parameters to a host PC for storage and later analysis, including the values of n_{M}, n_{S}, $\Delta\varphi$, RTT, and the fractional part of the slave timestamp counter.

Phase estimation method

The first test was intended as an evaluation of the effect of clustering-based post-processing on DDMTD phase estimations. The bidirectional link was established once and the set of continuous raw DDMTD measurements for the resulting, stable phase difference at the master node was recorded. The clustering algorithm was then activated and the set of processed phase estimations was also recorded for various values of θ. The results for $\theta = 1/256$ are histogrammed in Fig. 3.27, where the measurements centered around the correct phase difference are substituted by a single, much narrower peak with the same center value. In this case, phase measurement resolution is reduced from 309 ps to 16 ps FWHM and the peak-to-peak width of the set of correct measurements goes from 1.3 ns to under 50 ps. Most importantly, the clustering step results in all unwanted measurements at the opposite edge being completely removed.

Measurements were taken for θ ranging between 1/16 and 1/1024. The resulting resolutions are summarized in Table 3.3, together with the expected resolution

θ	Measured FWHM	Expected FWHM
1/16	55.6 ps	54.7 ps
1/32	39.4 ps	38.7 ps
1/64	28.4 ps	27.3 ps
1/128	20.9 ps	19.3 ps
1/256	15.9 ps	13.7 ps
1/512	12.7 ps	9.6 ps
1/1024	10.9 ps	6.8 ps

Table 3.3: Measured resolution of DDMTD phase estimations as a function of the learning rate θ.

values according to (3.52) for a base FWHM resolution of 309 ps without clustering. The results match rather well for high values of θ, validating the theoretical prediction. For smaller θ, the measurements start to deviate from the predictions as if there were an additional source of error. Indeed, computing the quadratic difference between measurements and predictions yields an almost constant value around 8 ps FWHM, which suggests the existence of an additional, independent source of error of this magnitude. This value also imposes a lower bound on achievable resolution, and justifies the choice of $\theta = 1/256$ for subsequent tests as it is the highest value that results in a resolution below twice the lower limit.

Validation of the synchronization algorithm

Correct operation of the pairwise synchronization algorithm was tested next. The method was implemented with a refresh rate of 1 Hz and a guard interval of $[-640\,\text{ps}, 1280\,\text{ps}]$, enough to avoid zero crossings of DDMTD and to guarantee phases greater than the constraint $\delta_{\text{max}} = 1\,\text{ns}$. FPGAs in both nodes were also configured to output small pulses whenever their local integer timestamps fell within a certain range, with an approximate frequency of 150 Hz. These pulses were captured and monitored using a Tektronix TDS3014C oscilloscope, and the time difference between both pulses was measured. This procedure was repeated, resetting and power cycling the evaluation boards between tests. In all cases, the synchronization algorithm converged in a single iteration, i.e. pulses on the oscilloscope became aligned with a difference smaller than $T_{\text{clk}} = 6.4\,\text{ns}$ within 1 s of link establishment, as the pairwise synchronization step was carried out. Further iterations yielded timestamp correction offsets in the range of ±20 ps. Moreover, the time difference measured with the oscilloscope was within 160 ps of the phase difference indicated by the fractional part of the slave timestamp; greater precision could not be obtained with the employed measurement method since the oscilloscope used in the tests had an 80 ps interval between samples. An example oscilloscope output is shown in Fig. 3.28.

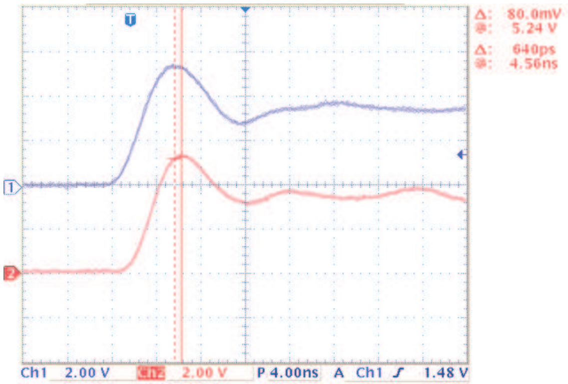


Figure 3.28: Example oscilloscope screenshot showing the delay between the master (up, blue) and slave (down, red) timestamp pulses. The synchronization algorithm yields a 620 ps delay, the oscilloscope measure is 640 ps.

The time difference between pulses, or equivalently the phase difference between the master and the slave clocks, was observed to vary in discrete steps of about 640 ps. This is consistent with the theoretical expectation: the local clock at the slave is the recovered clock (i.e. $\Delta\varphi_{\mathrm{S}} = 0$) and its phase relationship with the master clock, according to (3.1), is given by the one-way latency t_{MS} of the master-to-slave half-link. Hence, by (3.33) and (3.37)

$$\begin{aligned}
\varphi_S - \varphi_M &= 2\pi \frac{t_{\mathrm{MS}} \bmod T_{\mathrm{clk}}}{T_{\mathrm{clk}}} \\
&= \text{constant} + 2\pi \frac{t_{\mathrm{RX,S}} \bmod T_{\mathrm{clk}}}{T_{\mathrm{clk}}} \\
&= \text{constant} + 2\pi \frac{n_{\mathrm{S}}}{10} \bmod 2\pi
\end{aligned} \tag{3.58}$$

so the phase difference should indeed vary in steps of $T_{\mathrm{clk}}/10$ and be completely determined by the value of n_{S}. Not all 10 values were observed due to the guard interval that prevented some of them from being chosen during link establishment.

Estimated resolution

Since the delay measurements were taken with a low bandwidth, low resolution oscilloscope (100 MHz and 1.25 GSPS), the last test could only provide a first validation of the method, but was not enough to adequately measure its resolution. In order to obtain an estimate, the digital results from 100 different power cycles were recorded instead of the oscilloscope measurements. Guard intervals were removed for this test. The combined histograms of the resulting phase estimations

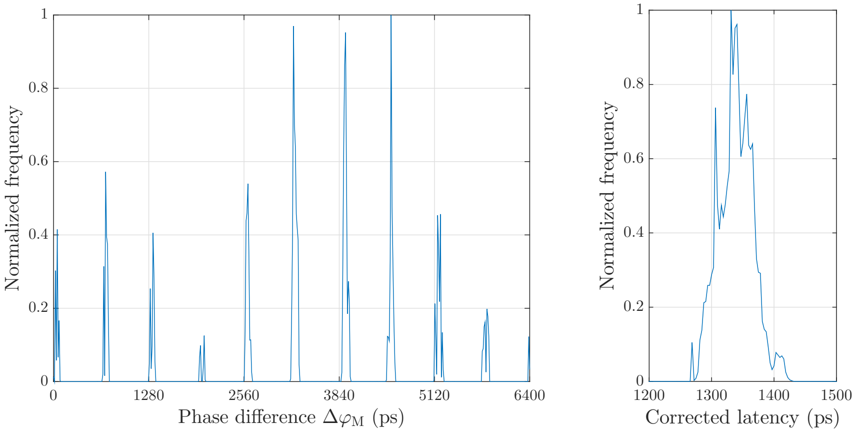


Figure 3.29: Distribution of phase differences measured over 100 separate power cycles. Phase is displayed as the equivalent time delay for a 156.25 MHz clock. Left: raw phase measurements. Right: corrected for latency variation as (3.60).

are shown in Fig. 3.29 (left), clearly showcasing ten different, uniformly spaced peaks. Again, this is consistent with the developed theory: using (3.33), (3.34) and (3.37), the round-trip time is equal to

$$\begin{aligned} RTT = t_{\mathrm{MS}} + t_{\mathrm{SM}} &= \text{constant} + t_{\mathrm{RX,S}} + t_{\mathrm{RX,M}} + \frac{\Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}} \\ &= \text{constant} + \frac{n_{\mathrm{S}} + n_{\mathrm{M}}}{10} T_{\mathrm{clk}} + \frac{\Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}}. \end{aligned} \tag{3.59}$$

However, RTT is known to be an integer number of clock cycles. Hence the value of $\Delta\varphi_M$ for different power cycles varies with a granularity given by the bit shifts, i.e. in steps of $T_{\mathrm{clk}}/10$.

In order to obtain a combined estimation of phase resolution, the values of

$$\left(\frac{n_{\mathrm{S}} + n_{\mathrm{M}}}{10} T_{\mathrm{clk}} + \frac{\Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}} \right) \bmod T_{\mathrm{clk}} \tag{3.60}$$

from all power cycles were histogrammed together in Fig. 3.29 (right). Equation (3.59) implies that this is ideally a constant value; indeed, all pulses collapse into a single one, and its width yields an estimation of the resolution that the phase difference $\Delta\varphi_M$ is obtained with. The measurement has a resolution of 64 ps FWHM and 28 ps rms and a peak-to-peak deviation of around 200 ps.

Finally, this test also provided experimental validation of the need for guard intervals, as an additional $T_{\mathrm{clk}}/2 = 3200$ ps delay between oscilloscope pulses was observed to appear for two specific phase peaks, or equivalently for two particular values of $n_{\mathrm{M}} + n_{\mathrm{S}} \bmod 10$, corresponding to phase differences $\Delta\varphi_{\mathrm{M}}$ close to zero and smaller than the propagation delay δ of the handshake signal.

Chapter 4

Positron Emission Tomography

After the generic discussion of High Energy Physics setups and data acquisition systems in chapter §2, a specific application is considered in this chapter. *Positron Emission Tomography* (PET) is a medical imaging technique that is based on the detection of annihilation radiation generated by short-lived positrons. Accordingly, PET may be considered a particular discipline within the field of High Energy Physics and, in particular, the design of PET scanners shares a large number of common concepts and components with that of HEP experiments and small-scale accelerator installations. The next few sections are devoted to the presentation of the fundamentals of PET and the physical, electronic and mathematical principles behind it. The requirements of data acquisition systems for PET are discussed separately in the chapter §5.

4.1 Overview

Positron Emission Tomography is a medical imaging technique that provides 3D images of functional processes in the body by detecting annihilation radiation emitted from within the patient. In order to achieve that, a *radiopharmaceutical* or *radiotracer* is administered to him; this is a radioactive substance that is chemically equivalent to some organic compound typically found within the human body but has had one of its atoms replaced by a radioactive isotope. Radiotracers employed in PET are characterized by the fact that they emit positrons when they decay. Shortly after they are emitted, positrons interact with electrons in the tissue causing an annihilation event which results in two gamma rays, or more rigorously two annihilation photons, being emitted in opposite directions.

Radionuclide	Half-life (min)	Applications in PET	Radiotracer
^{11}C	20.4	Protein synthesis	Methionine
^{13}N	9.97	Blood flow (heart)	Ammonia
^{15}O	2.03	Blood flow (heart and brain)	Water, CO_2
^{18}F	109.8	Glucose metabolism	FDG
		Bone imaging	F^- ions

Table 4.1: Typical radiotracers employed in PET scanning and some of their applications. Data extracted from [160] and [161].

A PET scanner features two or more gamma detectors, each typically consisting of scintillators coupled to photoelectric transducers that surround the body or body part under study. The role of a PET scanner is then to detect pairs of annihilation photons that are coincident in time. Each detected photon yields a detection position in the scanner; thus, each coincident pair defines a straight line, called *line of response* (LOR), that contains the position where the positron was annihilated.

Once the radiotracer enters the metabolic chain, the PET scanner obtains lines of response that are known to intersect the volume where the substance is located. After a large number of LORs is captured, a computer can apply an image reconstruction algorithm to obtain a 3D image of the location and concentration of the radiotracer. In particular, it is possible to visualize individual organs and, more importantly, to study of the metabolism of particular substances dynamically by obtaining distribution maps for successive time intervals, i.e. PET can provide *functional* rather than *structural* images.

Radiotracers are typically generated using small-scale particle accelerators, particularly cyclotrons.[1] Specific radiotracers are chosen for the tracking of different organs or processes. Table 4.1 summarizes some of the most common positron-emitting radioisotopes employed in PET imaging together with their application and radiopharmaceuticals that contain them.

A radionuclide's *half-life* is the time that it takes for half of the radioactive contents in a sample to undergo decay, regardless of the sample size. Radionuclides with low atomic number such as the ones outlined in table 4.1 typically exhibit a short half-life. This implies a high emission rate and requires a shorter scan time for the patient in order to obtain the same amount of captured data. However, it also implies the need to synthesize the pharmaceutical at a close physical distance

[1] An older method of radiotracer production was to expose a regular material to neutron radiation from nuclear reactors. However, this process carries no transmutation because neutrons have null charge, so there is no way to separate the unstable isotopes from the stable ones chemically, resulting in samples with a very low fraction of radioactive compound.

from the patient and the detector, so the scan may be performed before most of the radiotracer is gone. Hence, the medical cyclotrons employed for radiotracer production have to be located in the same urban area, and often in the same hospital, as the PET scanner.

The main applications of PET are found in oncology, particularly in the detection of tumors, e.g. neuroimaging and Positron Emission Mammography (PEM), and in the study of blood flow in cardiology and neurology [161]. One of the key radiopharmaceuticals is fluorodeoxyglucose (FDG), a structural analog of glucose marked with an unstable ^{18}F atom. This radiotracer may be used for the assessment of glucose metabolism. Since most malignant tumors consume large amounts of glucose for their growth processes, FDG-PET is frequently used to provide images of tumors in order to locate them prior to surgery or to monitor their evolution for cancer detection and treatment evaluation.

A brief history of PET

Medical imaging encompasses a series of procedures that provide images of the human body for clinical purposes. Primitive medical imaging techniques can be traced back to classical Greece, where the temperature of the body surface was used as the source of information [162], and to the early 19th century when the first serious attempts at endoscopy took place [163]. However, the history of modern medical imaging, i.e. purely non-invasive techniques that yield planar sections or 3D representations of the inside of a patient, started in the late 19th century with the discovery and development of radiography [164], where the patient is exposed to an X-ray source and the intensity of X-radiation is measured after it has passed through him, yielding a planar image that represents the radiation attenuation coefficients within the body that may, in turn, be related to different types of tissue.

The 1920s mark the start of the development of nuclear medical imaging with the introduction of the idea of using radiation emitted by radionuclides within the body. The first nuclear imaging studies were performed on plants [165]. At this stage, however, studies had to rely on naturally occurring radiotracers. The first demonstration of the production of artificial radioactive atoms wouldn't be recorded until 1934 [166], soon after the first cyclotrons had been put to use. Research on the production of radiotracers would end up focusing on the use of nuclear reactors, taking advantage of the research efforts in the context of World War II.

On the detector side, the first external measurements of radiotracers in the human body were made in the 1940s using hand-held counters at different positions covering a target area [167]; the first automated scanner is due to B. Cassen in 1950 [168]. The early 1950s were also witness to the first two published works concerning the detection of annihilation radiation from positrons for medical diagnose [169], targeting brain tumors in both cases [170, 171]. The first detector designed specifically for single photons from positron-emitting radiotracers was disclosed in 1953 [172].

The landmark achievement in this direction is the *gamma camera* presented by H. Anger in 1957 and named after him [173]. The camera coupled several photodetectors to a large scintillator and combined their outputs using what is now known as *Anger logic* in order to achieve position-sensitive detection. In this way, it was able to increase the total detection area significantly with respect to its predecessors.

Although the Anger camera was only able to obtain planar images, it has been highly influential in the development of nuclear imaging and, even now, the working principle of most existing detectors are based on Anger's. For instance, in Single Photon Emission Computed Tomography (SPECT), Anger cameras are rotated around a patient in order to obtain a full set of projections, which are then used to reconstruct the tomographic image ([174], pp. 7–8). Similarly, the first true coincidence positron scanner appeared in the early 1960s [167] and consisted of a number of Anger cameras operating in coincidence, as had been proposed by Anger [175].

After vigorous research efforts during the 1960s, the first true PET tomograph appeared in 1973 [176]. Since then, detector performance has evolved thanks to advances in many technological aspects including, by chronological order, the production of radiotracers [177] (particularly FDG [178]), detector geometry [179], the introduction of computerized tomography [180], research in image reconstruction methods [181], and higher quality scintillators [182, 183]. Nevertheless, the principle of operation in today's PET scanners remains the same.

PET may be combined with other imaging techniques that provide structural images with very high resolution, such as Computed Tomography (CT) or Magnetic Resonance Imaging (MR). While it is possible to obtain images in separate detectors and combine them using software-based image fusion techniques, this approach is slow and subject to artifacts related to the movement of the patient between both scans. Combined dual-modality scanners have the potential to yield accurate images including both structural and functional information with minimal delay and inconvenience for the patient. PET/CT has been commercially available since 2001 [184]; figure 4.1 illustrates the difference between raw PET images and the combination with anatomic information from CT. Combined PET/MR systems exist since 2008 [185] and are one of the most important research topics in the field as of the writing of this text; in particular, the problem of operating PET detectors within the strong magnetic fields generated by a MR scanner is of great importance.

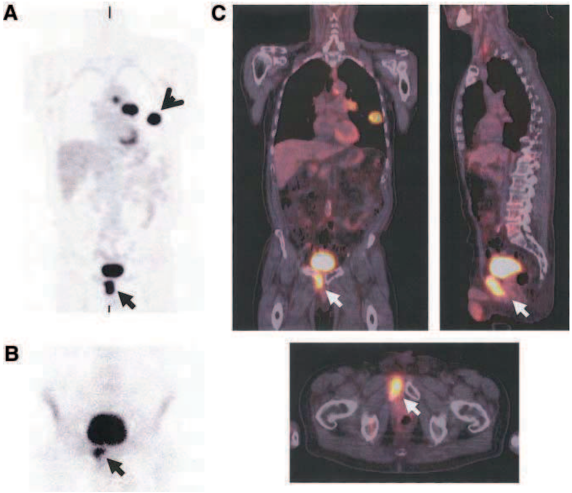


Figure 4.1: Comparison between images obtained by PET (left) and PET/CT (right). Reproduced from [184], © 2004 Society of Nuclear Medicine and Molecular Imaging, Inc.

4.2 Physical principles of PET

A typical PET scanner is formed by one or more rings of detectors that cover the active area where the patient will be located. Figure 4.2 depicts a diagram of a single PET ring and the basic physics behind the detection of one event. One of the radiotracer molecules administered to the patient decays, generating a reaction that eventually results in the emission of two annihilation photons in opposite directions. The surrounding detectors consist of scintillator crystals that convert the annihilation photon into a shower of optical photons and arrays of photodetectors that collect the light and generate related electrical signals. These signals are then used to estimate the impact positions of annihilation photons on the detectors in order to reconstruct the LOR. This section covers the main processes and components involved in the described scheme: the radiotracer disintegration, the scintillation, and the photodetectors.

4.2.1 Radiation generation

PET imaging relies on the detection of annihilation radiation in opposite directions originating from the volume under study. The physical processes that lead to the generation of the annihilation photons are described next, as well as the fundamental limitations they impose on the final resolution of the method.

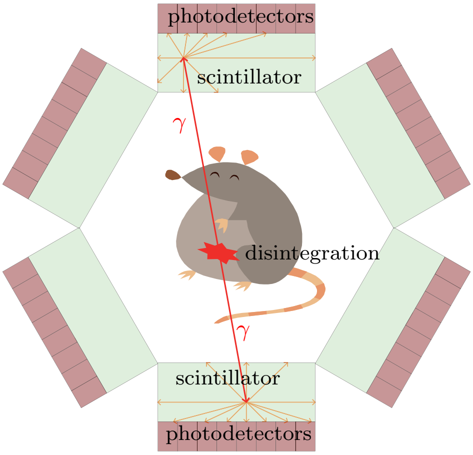


Figure 4.2: Diagram of a small animal PET scanner ring and the basic physical processes behind the acquisition of a single line of response.

Radioactive decay modes

Atomic nuclei are composed of protons and neutrons, which can be understood to be distributed in bands with different energy levels just like electrons typically are. Hence, it is possible for a given atom to reach more stable states (i.e. states with lower energy) by redistributing their protons and neutrons or even through the conversion of one into the other. In each case, whenever there is a transition to a more stable state, the law of conservation of energy demands that the energy difference be released in some form. Depending on the transition mode, this energy may be released either as a photon or as kinetic energy in an expelled particle.

The following is a list of the decay modes, i.e. types of transition between nuclear states, that are most commonly associated with nuclear medicine ([186], pp. 27–43). It should be mentioned that for any given unstable isotope there may be several possible decay modes with different probabilities, called *branching ratios*.

- *β^- decay:* This process consists of a neutron n converting into a proton p^+, and releasing an electron e^- so as to maintain electrical charge. A chargeless, massless particle known as *antineutrino*[2] $\bar{\nu}$ is also expelled. The reaction can be summarized in the chemical equation

$$n \rightarrow p^+ + e^- + \bar{\nu}.$$

[2] More accurately, an *electron antineutrino* $\bar{\nu}_e$. As of the writing of this text, it is hypothesized that the neutrino and antineutrino are actually the same particle, but the conjecture is yet to be proved or refuted.

The energy difference is released by the atom as kinetic energy, distributed between the electron and the antineutrino. Hence, while the total energy is always the same for a given unstable isotope, the energy of the released electron is not constant, and follows a continuous distribution given by Fermi's Golden Rule ([187], pp. 272–288).

- *β^+ decay:* Also called *positron emission*, this the fundamental decay mode for PET. This process is the dual of β^- decay and features a proton converting into a neutron and releasing a *positron* e^+, i.e. an elementary particle that is equivalent to an electron but with opposite electrical charge, and a *neutrino* ν:

  $$p^+ \rightarrow n + e^+ + \nu.$$

  As is the case with β^- decay, the released energy is split continuously between the positron and the neutrino according to Fermi's Golden Rule.

- *Electron capture:* In this case, a proton is transformed into a neutron by absorbing an orbital electron from the same atom. A neutrino is generated in the process.

  $$p^+ + e^- \rightarrow n + \nu.$$

  As the electron is absorbed by the nucleus, it leaves a vacancy in its orbital. An electron from a higher energy orbital abandons it in order to take its place, and the energy difference is released as a photon.

- *Isomeric transition:* These transitions only consist of a reordering of protons and neutrons in the nucleus. This never occurs spontaneously but rather after one of the other types of decay leaves the nucleus in a metastable state. The energy lost by the nucleus is released as a photon. Sometimes, this photon is absorbed by an orbital electron in the same atom, which is then expelled with a kinetic energy that equals the energy difference between the former and current atomic states. The whole process is then called *internal conversion*.

- There are additional nuclear decay modes, such as α *decay* where an α *particle* is released, composed of two protons and two neutrons (i.e. a helium nucleus). However, this decay mode only appears in very heavy elements and so its study remains outside the scope of this dissertation.

In general, unstable atoms with a high mass, i.e. with a high ratio of neutrons to protons, are more likely to use β^- decay modes transforming neutrons into protons, while less massive atoms are more likely to undergo processes that transform protons into neutrons. In this second case, both β^+ decay and electron capture are possible. The branching ratio of each interaction depends on the unstable isotope being considered, but β^+ decay is usually more likely than electron capture.

Temporal distribution of radiation

Decay of unstable nuclei is a statistical process, and the exact moment when the decay is going to happen cannot be determined in advance. The probability distribution describing the time of decay is called *memoryless*, in the sense that the probability that the decay takes place in any given time interval is independent of the starting point of said interval. In other words, the atom's age has no influence on its decay process.

It can be proved, using elementary arguments, that this simple feature is enough to give an almost complete characterization of the distribution of unstable atoms within a given sample ([188], p. 8): Let $N(t)$ be the number of unstable nuclei in a sample at time t. In the next time interval dt, the probability that each individual nucleus undergoes decay is $\lambda \cdot dt$, where λ is a constant parameter that depends on the isotope under consideration. Hence, the expected change in N is precisely

$$dN = -N\lambda \cdot dt. \tag{4.1}$$

Solving this differential equation yields the formula

$$N(t) = N_0 e^{-\lambda t} \tag{4.2}$$

where N_0 is the amount of unstable nuclei in the sample at the time $t = 0$.

The expected activity of a sample is thus described by an exponential distribution with a parameter λ that determines the particular radionuclide's decay rate. Instead of λ, the value that is usually given is the isotope's *half-life*, i.e. the time period after which the sample's activity is expected to be reduced to half its starting value. The half-life $T_{1/2}$ is given by

$$T_{1/2} = \frac{\log 2}{\lambda} \approx \frac{0.693}{\lambda} \tag{4.3}$$

as can be deduced easily from (4.2).

Positron annihilation

Whenever an unstable isotope undergoes β^+ decay, a positron is expelled with a randomly distributed kinetic energy. This positron then undergoes a thermalization process, whereby it interacts with the surrounding matter until it reaches thermal equilibrium. The main ways of interaction are collisions with either nuclei or orbital electrons in nearby atoms.[3] Elastic collisions with nuclei, also known as *Coulomb scattering*, imply a small loss in kinetic energy and a random change of

[3] An additional source of energy loss is the emission of *bremsstrahlung*, however this effect turns out to be negligible under typical conditions in clinical PET, particularly low kinetic energy and an environment composed mainly of light elements ([189], pp. 2–4).

direction. On the other hand, collisions with electrons imply ionization or excitation and are the main contributors to positron energy loss; the average energy loss per unit distance is given by the Bethe formula ([20], pp. 31–32).

The positron will thus describe an erratic trajectory until thermalization is complete and its kinetic energy is low enough. The distance between the position where the positron was originated and the position where it ends its movement is called the *positron range*. Notice that the processes governing energy loss and particle movement are completely separated, as movement is almost completely determined by collision with nuclei. This enables estimation of the positron range in the following way: the Bethe formula provides the expected total trajectory length, which can then be used to estimate positron range according to the density characteristics of the environment. The presence of a static magnetic field further reduces the positron range by forcing the positron trajectories to bend due to Lorentz force; this is particularly useful in simultaneous PET/MR systems [190].

When the positron finally reaches a rest state, it annihilates with a nearby electron, converting their joint mass into energy which is emitted in the form of two annihilation photons with the same energy, each equivalent to the electron rest mass and equal to $E_e = 511\,\mathrm{keV}$. By the law of conservation of momentum, the photons are emitted in almost diametrically opposed directions; however, there is a small angular deviation θ so as to account for the non-zero momentum p of the orbital electron at the moment that it annihilates with the positron, given by [191]

$$\theta \approx \sin\theta \approx \frac{p}{E_e}. \tag{4.4}$$

The distribution of p (and hence of θ) depends on the surrounding material, which can usually be assumed to be water due to the high water content in tissue. The most likely case is that the positron annihilates with an electron that is bound to a nucleus, yielding an angular deviation whose distribution has a FWHM of 0.25° [192]. Other, less likely cases are described in ([193], pp. 20–24).

Summary

As a summary, the physical process of generation of annihilation radiation in PET is sketched in Fig. 4.3. The basic principle of PET, i.e. the detection of annihilation photons determining LORs that are known to intersect the radiotracer distribution, is subject to inaccuracies due to two independent physical causes:

- The photons are not generated at the radiotracer position, but rather at a small distance. The distribution of positron range depends on the radionuclide and its FWHM varies between 0.1 mm and 0.5 mm for the examples listed in Table 4.1 [192].

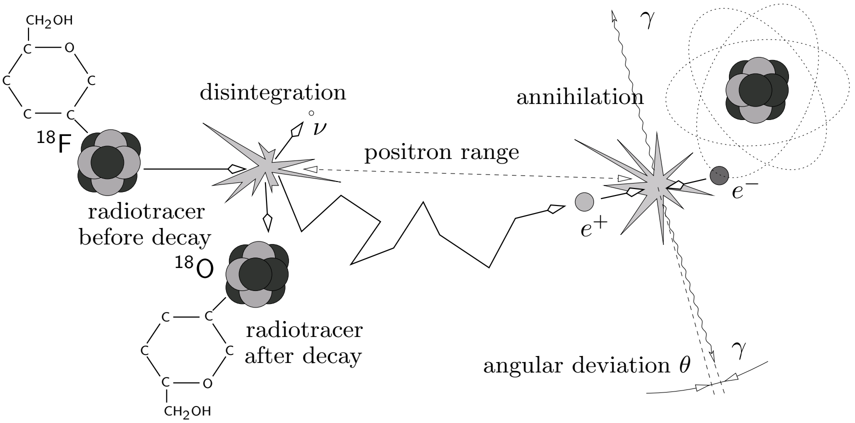


Figure 4.3: Physical processes and inaccuracies behind annihilation photon generation in PET, including β^+ decay, annihilation, positron range, and angular deviation. Adapted from [194] with permission of the author.

- Annihilation photons are not exactly diametrically opposed but present a small angular deviation in the range of 0.25°, hence the LOR is at a small but positive distance of their generation point. The more separated the radiation detectors, the larger this error becomes.

These two effects determine the intrinsic physical limitations of PET imaging and impose a lower bound on the spatial resolution that can be achieved with this technique.

4.2.2 Radiation detection

According to the classification of particle detectors that was presented in Table 2.1, the preferred radiation detectors for PET scanners are scintillators and semiconductor detectors.[4] The latter have typically suffered from large intrinsic noise ([194], pp. 23–25) and have thus been employed less frequently. Hence, most PET detectors are based on scintillator crystals, and detection is carried out in three steps:

- First, radiation is stopped by the detector and its energy transferred to a charged particle.

- The charged particle causes the emission of optical photons as it moves through the scintillator.

[4] Several designs based on ionization detectors were attempted in the past [195–197], but these devices were generally too bulky to fit in a compact scanner. Lately, alternative liquid Xenon detectors have been proposed [198].

- Finally, optical photons are collected by a photodetector and converted to an electrical signal.

This section covers the first two steps, i.e. the conversion of an annihilation photon into optical photons. The last detection step will be described in §4.2.3.

Interaction of annihilation photons with matter

The annihilation photons used in PET imaging are all generated with an energy $E_e = 511\,\text{keV}$, equivalent to a frequency $\nu = 1.24 \times 10^{20}\,\text{Hz}$. At this frequency range, the behavior of electromagnetic radiation regarding its interaction with matter is closer to that of particles than that of waves. The dominant effects, outlined in Fig. 4.4, are caused by the collision of the photon with a bound electron that is released in the process ([186], pp. 45–68):

- *Photoelectric absorption:* This process usually takes place when the photon collides with an electron in one of the innermost shells of an atom. The photon is completely absorbed and the electron is expelled from the atom with a kinetic energy equal to the difference between the absorbed photon and the electron's binding energy.[5] This electron is then called *photoelectron*.

  The vacancy created in one of the inner shells of the atom is occupied by an electron from a shell farther from the nucleus; typically, it is an electron from the immediate next shell. A vacancy is thus created in this second shell which gets filled in a similar manner by an electron from a third, more external shell, etc. In this way, a cascade of electrons is generated until the outermost shell of the atom is reached. For each electron transition, a photon is released with an energy equal to the difference in binding energy.

- *Compton scatter:* When the electron involved in the interaction is unbound or belongs to one of the outermost atomic shells, the photon is not completely absorbed; an elastic collision takes place instead, and part of the photon energy is transferred as kinetic energy to the electron, which is then called *recoil electron*.

  After the collision, the photon may be scattered in any direction. The deflection angle θ follows the probability distribution given by the Klein-Nishina formula ([20], pp. 50–51), and the energy E_{sc} of the scattered photon is given by

$$E_{\text{sc}} = \frac{E_{\text{inc}}}{1 + \frac{E_{\text{inc}}}{E_e}(1 - \cos\theta)} \tag{4.5}$$

[5] In PET, the binding energy is usually much smaller than the photon energy, hence photoelectrons are expelled with an energy approximately equal to that of the impinging annihilation photon.

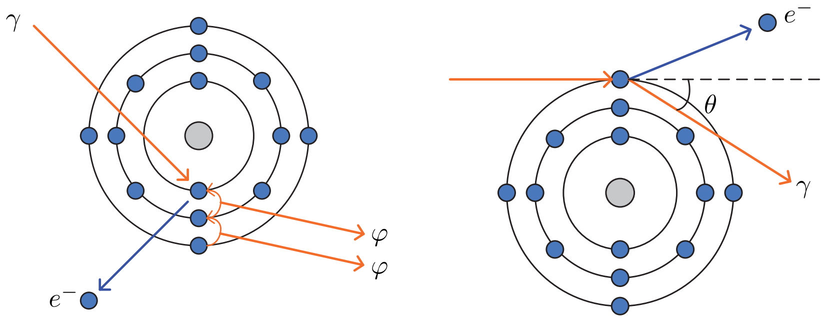


Figure 4.4: Left: Photoelectric absorption and generation of characteristic radiation. Right: Compton scatter.

where E_{inc} is the energy of the incident photon and $E_e = 511\,\text{keV}$. In particular, a small deflection angle implies a small energy loss.

Figure 4.5 shows the distribution of θ and E_{sc} for annihilation photons with $E_{\text{inc}} = E_e$. The beam width at half maximum is approximately 86°, so the deflection angle is below 45° in most cases. Consequently, the photon is much more likely not to modify its path and energy significantly (*forward scattering*) than it is to change directions abruptly and lose a large fraction of its energy (*backscattering*). The scattered photon may, in turn, interact with matter again via photoelectric absorption or further Compton scatter.

Two additional interaction modes are possible for gamma rays, *Rayleigh scatter* and *pair production*, although their probability at the photon energies employed in PET is negligible. In particular, the contribution of Rayleigh scatter is only noticeable for low photon energies, and even then it does not significantly alter the photon path or energy. On the other hand, pair production, where the photon is replaced by an electron-positron pair, is only possible if the photon energy exceeds $2E_e = 1.02\,\text{MeV}$, which cannot happen in PET applications.

In the energy range employed in PET, photoelectric absorption is the dominant interaction mode for high atomic number Z, whereas Compton scatter is more likely within lighter materials. In general, the overall probability of interaction always increases with the density of the matter ([174], pp. 13–15).

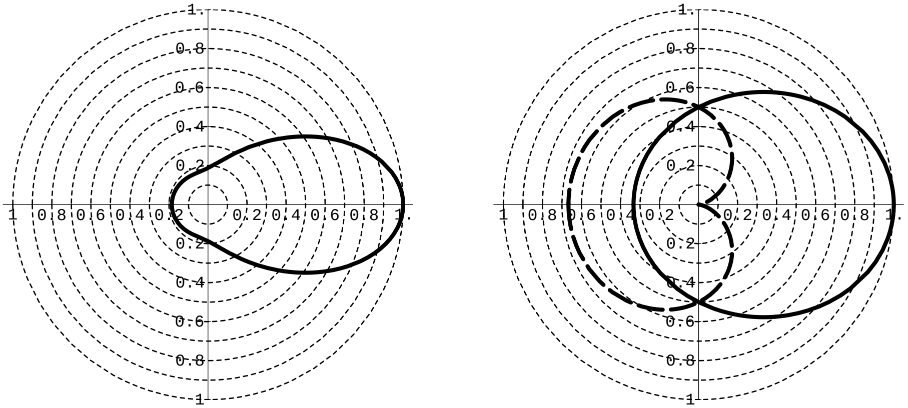


Figure 4.5: Compton scatter of 511 keV photons depending on the deflection angle θ. Left side: normalized probability of θ for a constant solid angle. Right side: energy of the scattered photon (solid line) and recoil electron (dashed line), normalized to 511 keV. Reproduced from [194] with permission of the author.

Attenuation

The probability of interaction of an annihilation photon with the surrounding material before arriving at the detector is modeled by a *linear attenuation coefficient* $\mu(\mathbf{x})$ that is defined as the total probability of interaction per unit length including all possible effects, and depends on the particular point $\mathbf{x}$.

Given a linear beam of photons in a particular direction, its intensity is gradually attenuated as it traverses matter due to interactions. Let $N(x)$ be the amount of photons left at position x; then, by definition, the rate of change over an infinitesimal distance dx is

$$dN(x) = -\mu(x) N(x) dx. \tag{4.6}$$

Solving this differential equation yields

$$N(x_1) = N(x_0) \exp\left(-\int_{x_0}^{x_1} \mu(x) dx\right) \tag{4.7}$$

for the amount of photons left at position x_1 relative to the starting position x_0. In particular, the probability that a given photon does not interact on the path from x_0 to x_1 is given by

$$p(x_0 \to x_1) = \exp\left(-\int_{[x_0,x_1]} \mu\right) \tag{4.8}$$

where term inside the exponential denotes a line integral over the segment $[x_0, x_1]$.

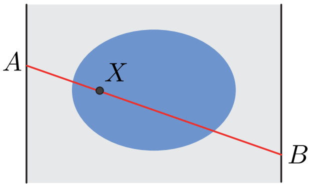


Figure 4.6: Example of a line of response AB passing through decay position X.

Let us now consider the situation in the context of PET as described in figure 4.6, where a positron annihilation takes place at position X and emits two annihilation photons in opposite directions along a straight LOR with endpoints A and B. The probability that both photons reach the detectors at A and B is then

$$\begin{aligned} p &= p\left(X \to A\right) p\left(X \to B\right) \\ &= \exp\left(-\int_{[X,A]} \mu\right) \exp\left(-\int_{[X,B]} \mu\right) = \exp\left(-\int_{[A,B]} \mu\right). \end{aligned} \tag{4.9}$$

The noteworthy consequence is that the effect of attenuation in PET depends on the line of response AB but is independent of the particular position X along the LOR where the event takes place. This is not the case in other nuclear imaging modalities such as SPECT.

Scintillators

As seen in the last section, detection of annihilation photons results in one or more *knock-on electrons*, which may be photoelectrons or recoil electrons depending on the interaction type. These electrons travel through matter and collide with other electrons, leaving them in an excited state. Each collision causes the incident electron to lose some kinetic energy, and the process continues until all kinetic energy is absorbed this way.

A *scintillator* is a material that converts impinging ionizing radiation into scintillations i.e. bursts of visible light. More specifically, scintillators have the property that the excited states induced by knock-on electrons are relaxed following a process that results in emission of photons in the optical range. It is remarkable that the wavelength λ of the optical photons, i.e. the color of the scintillation light, does not depend on the energy of the absorbed annihilation photon but rather on the scintillator, since it is determined by the gaps between the energy levels involved in the deexcitation processes. The energy of the impinging photon is related to the number of generated optical photons instead, i.e. the intensity of the scintillation;

under ideal conditions, the relationship is linear, and the proportionality constant is called the *light yield*.

Scintillator materials are typically divided into two types: organic and inorganic, with slightly different principles of operation. In *organic scintillators*, π bonds result in electrons that are bound to whole molecular structures rather than single atoms, with an energy level structure that allows relaxation transitions through emission of photons with a few eV, i.e. the optical photon range. In spite of their faster response time, organic scintillators are not commonly found in PET applications due to the large probability of *quenching*, i.e. alternate deexcitation processes through heat instead of photon emission, that result in poor efficiency and linearity ([20], pp. 220–231).

On the other hand, *inorganic scintillators* are based on crystalline insulators where the excitation of electrons causes them to rise from the valence band to the conduction band. However, a pure crystal usually results in a very ineffective scintillator since the photons generated by deexcitation contain enough energy to excite different, nearby electrons and are thus eligible to be reabsorbed, resulting in a large fraction of the light never leaving the crystal volume. In order to increase their efficiency, certain dopants are added during the crystal growth process in order to introduce a small amount of impurities called *activators*. These disturbances in the crystal lattice create additional energy levels in the forbidden band between valence and conduction, allowing more efficient transitions via these intermediate levels. They also provide the (unwanted) possibility for an electron to remain in one such intermediate level for a relatively long time until relaxation and delayed photon emission; this phenomenon is called *phosphorescence*.

The time variation of light intensity generated by a scintillator is commonly approximated by an exponential

$$I\left(t\right) = \frac{I_0}{\tau_1} e^{-t/\tau_1} \tag{4.10}$$

for $t \geq 0$, with integral I_0 related to the absorbed energy and a *decay constant* τ_1 that depends on the half-life of the excited states. For fast scintillators, the rise time needs to be considered and the more detailed approximation

$$I\left(t\right) = \frac{I_0}{\tau_1 - \tau_r} \left(e^{-t/\tau_1} - e^{-t/\tau_r}\right) \tag{4.11}$$

is employed, where τ_r models the rise time and depends on the time it takes for excited electrons to reach the activators in the crystal [199]. Moreover, several exponentials may need to be considered for the modeling of fast and slow components in order to account for phosphorescence. In this case, an approximation is given by

$$I\left(t\right) = \frac{I_1}{\tau_1} e^{-t/\tau_1} + \frac{I_2}{\tau_2} e^{-t/\tau_2} \tag{4.12}$$

where $\tau_2 \gg \tau_1$, and a finite rise time may be added to the fast component τ_1 in the same manner as in (4.11). Even for very large values of τ_2, this slow component may need to be considered as it causes a slow drift in the signal baseline that might need to be compensated against [200].

The preferred interaction mode between annihilation photons and the scintillator is photoelectric absorption, which results in a single knock-on electron emitted with an energy almost equal to the 511 keV from the impinging photon and a localized scintillation, while Compton scattered events may be composed of more than one interaction and result in several smaller scintillations. The probability that an absorbed photon undergoes photoelectric absorption before Compton scattering is called the *photofraction*. An advisable scintillator material should therefore be dense (having a high attenuation coefficient for overall sensitivity) with a high effective atomic number Z (increasing the photofraction).

In general, the desirable properties of a scintillator are [20, 188]:

- High density and high Z_{eff} for the aforementioned reasons, i.e. sensitivity and photofraction.
- High efficiency, defined as the fraction of kinetic energy that gets converted into detectable light.
- High linearity in the relationship between absorbed energy and generated scintillation intensity.
- Transparency, so the scintillation light can traverse the crystal undisturbed, and a refraction index similar to that of glass, for optimal coupling to an external photodetector.
- Short decay constant τ_1 in order to generate fast pulses.
- Ease of use and manipulation. For example, hygroscopic crystals such as NaI are problematic due to their undesired degradation with humidity.

Unfortunately, real scintillator materials cannot fulfill all these requirements simultaneously and a trade-off has to be found.

For PET, the most interesting properties in decreasing order of importance are high attenuation, a short decay constant, low cost, and high light yield ([194], pp. 25–26). A list of the most representative scintillators found in PET is given in Table 4.2 along with their key parameters. NaI was one of the first inorganic scintillators to be developed and has been widely used since the 1940s for photon detection [201]. It was largely replaced by BGO in PET applications after the appearance of the latter in the 1970s due to its increased sensitivity, although its light yield was significantly smaller. LSO is a breakthrough material introduced

Scintillator	Dopant	Z_{eff}	Density	λ	Light yield	τ_1	μ^{-1}	Photo-
			(g/cm^2)	(nm)	(photons/MeV)	(ns)	(cm)	fraction
NaI	Tl	50.8	3.67	415	38 000	230	2.6	17 %
BGO	none	75	7.13	480	8200	300	1.12	40 %
GSO	Ce	59.4	6.71	440	9000	56	1.4	25 %
LSO	Ce	66.4	7.4	420	25 000	40	1.14	32 %
LYSO	Ce	66	7.4	428	32 000	41	1.1	30 %
$\mathrm{LaBr_3}$	Ce	47	5.3	360	61 000	25	2.13	15 %

Table 4.2: Common scintillator crystals employed in PET. Adapted from [194].

in the early 1990s that combines the advantages of BGO and NaI with a much faster response time [183], and is widely regarded as the scintillator of choice for modern PET scanners, in spite of its higher price and natural radioactivity. The most common scintillators in PET today are BGO, GSO and LSO together with LSO-related materials such as LYSO.[6]

4.2.3 Photodetectors

A component that converts scintillation light into measurable electrical signals is needed at the end of the detection chain. Photodetectors usually achieve that following a two step process. The first step, called *photoconversion*, consists in the transfer of energy from impinging optical photons to electrons via photoelectric absorption. As the resulting amount of electrons may be very low, an additional amplification step that converts them into a noticeable signal is integrated in typical photodetectors.

Several photodetector types can be distinguished depending on the physical implementation of these two steps. Two of them are particularly important in PET due to widespread use: vacuum detectors like the Photomultiplier Tube (PMT), and solid-state detectors including Avalanche Photodiodes (APD) and the more recent Silicon Photomultipliers (SiPM).

Photomultiplier tubes (PMT)

Figure 4.7 shows the main components of a modern PMT. The first element is a *photocathode* that implements the photoconversion step. Optical photons impinge on the external cathode surface and generate photoelectrons that migrate towards the internal cathode surface and are liberated to the vacuum if their remaining energy is higher than the electric work function of the cathode material. The photocathode's performance is given by its *sensitivity*, or equivalently by its *quan-*

[6] LYSO and related materials are essentially modified versions of LSO that sacrifice some performance in order to circumvent the patent on LSO ([202], p. 45).

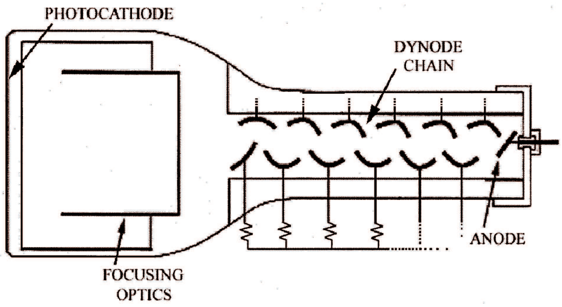


Figure 4.7: Structure of a photomultiplier tube. Reproduced from [203], © 2003 IEEE.

tum efficiency η, defined as the ratio between emitted photoelectrons and incident scintillation photons; both values depend on the impinging photon wavelength λ. Photocathodes are usually very thin in order to increase the probability that photoelectrons reach the surface; however, this decreases their absorption capabilities. As a result, PMTs tend to have rather low η, with maxima (with respect to λ) in the order of 20 %.

The amplification step is realized by a series of electrodes called *dynodes* that are biased at increasing voltage levels. Electric fields are thus induced between the photocathode and the first dynode and between successive dynodes that accelerate liberated electrons so that, once they reach the next dynode, they have gained enough kinetic energy that they are able to excite a moderately large number of electrons, a fraction of which reach the surface with enough energy to escape the dynode. These secondary electrons are, in turn, accelerated towards the next dynode, where each of them generates further electrons in the following stage. After the last dynode, an output anode collects all electrons in the avalanche, generating an electrical current signal that can be processed by external readout electronics. Each dynode stage is characterized by its multiplication factor δ, defined as the mean amount of emitted secondary electrons per impinging electron, and so the *PMT gain*, i.e. the number of electrons generated at the anode for each photoelectron, is given by

$$G = \alpha\delta^N \tag{4.13}$$

where N is the number of dynode stages and α is a loss factor that accounts for the fraction of photoelectrons that fail to reach the first dynode. Typical values of G between 10^5 to 10^6 are achieved with around $N = 10$ dynodes.

The statistical nature of the detection process in the PMT implies that energy and time measurements are subject to a positive variance. The total charge (i.e. integrated current) at the anode output is ideally equal to

$$G \cdot \eta \cdot \frac{E}{E_\lambda} = \frac{G\eta\lambda}{hc}E \tag{4.14}$$

where $E_\lambda = hc/\lambda$ is the energy of the optical photons (here, E/E_λ is the amount of photons reaching the photodetector), and thus proportional to E, the energy deposited by the annihilation photon. However, photoconversion and dynode multiplication can be understood as Poisson processes, i.e. the values of η and δ have Poisson distributions; as a consequence, the PMT gain has variance [204]

$$\sigma_G^2 = \frac{G\,(G-1)}{\delta - 1} \tag{4.15}$$

and a rather high relative variance $\sigma/G \approx (\delta - 1)^{-1/2}$ in the range of $0.5 - 1$. Gain variations above 50 % are thus to be expected.

Event timing is provided by the rising edge of the output pulse. There is a delay, called *transit time*, between the time when an optical photon strikes on the photocathode and the generation of a current pulse at the anode. A constant delay would suppose no drawback for timing; however its variation, the *transit time spread*, widens the time response of the scintillator and decreases time resolution. Additionally, the PMT response has a minimum rise time that, although typically short, is still longer than the light pulse's rise time and thus dominates the shape of the output pulse.

A "pure" PMT as described yields a signal that provides timing and energy information about the scintillation, however nothing can be inferred about the position where the scintillation has taken place. This problem is solved by *position-sensitive PMTs* (PSPMT), where photocathodes are segmented and special dynode structures with low spatial dispersion are designed in such a way that they function as several independent multipliers, each of them collecting photoelectrons from a particular cathode section and generating a separate amplified signal at its own anode output. A PSPMT is thus functionally equivalent to an array of independent PMTs with a common mechanical frame and biasing circuit, and its outputs provide a breakdown of the incident scintillation energy into space regions, from which an estimation of the event position may be obtained [205]. The segmentation of the anode system implies that single outputs no longer yield position-independent information about timing or total energy;[7] instead, PSPMTs provide a signal from the last dynode, where the liberation of electrons results in a variation of its potential and a pulse that contains the same timing and energy information as a single anode output would.

Noise in photodetectors are usually specified as a *dark current*, i.e. the electrical signal appearing at the detector output in the absence of light [203]. The main source of dark current in PMTs is thermionic noise at the photocathode and first dynodes: due to temperature, electrons may carry kinetic energies that occasionally exceed the work function and allow their spontaneous emission, triggering

[7] Anode outputs may be externally summed in order to recover this information, but this approach carries a loss of resolution due to electronic noise.

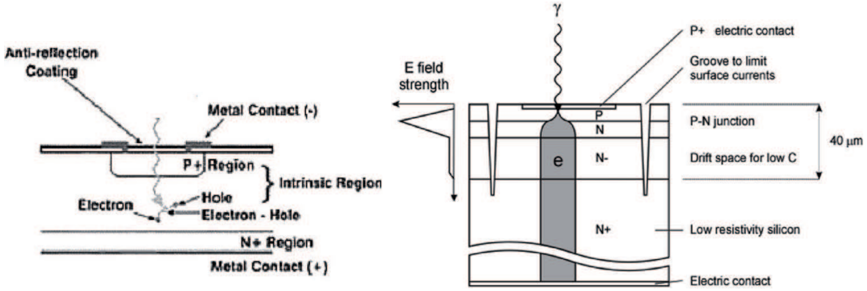


Figure 4.8: Structure of PIN (left) and APD (right) photodiodes. Reproduced from [206], © 2004 Elsevier.

an avalanche; of course, this noise contribution increases with temperature. Additional sources of dark noise can be identified such as leakage currents between different electrodes. The amplification process, however, introduces a relatively low amount of noise when compared to other detectors, rendering PMTs one of the best photodetector types in terms of SNR.

Avalanche photodiodes (APD)

In a semiconductor detector, photoconversion results in the generation of electron-hole pairs rather than electrons. The simplest such device is the *PIN photodiode*, shown in Fig. 4.8 (left). The name of the device refers to the composition of its layers: *p*-intrinsic-*n*. Electrodes are placed at the diode ends in order to reverse-bias it. The detector's sensitive volume is a thick layer made of intrinsic semiconductor, i.e. very lightly doped substrate; thin layers of strongly doped material are added close to the electrodes in order to confine the depletion region to the intrinsic layer. Electron-hole pairs generated by photoelectric absorption are driven to the electrodes by the electric field. The charge collected at the output electrode is thus proportional to the captured optical energy.

An important drawback of PIN diodes as photodetectors is the fact that they only perform photoconversion but no multiplication, resulting in weak signals that require external amplification. *Avalanche photodiodes* solve that problem by introducing an additional strongly doped region on one side of the *I* layer in order to create a *p*-*n* junction, and allowing higher reverse bias voltages to be applied between its electrodes (in the order of 100 V to 200 V for silicon). This results in a very strong electric field in the *p*-*n* junction, as shown in Fig. 4.8 (right), that accelerates photoelectrons to the point where they can excite other electrons in the region and create an avalanche of electron-hole pairs. Multiplication gain can be approximated as ([22], p. 90)

$$G \approx \frac{1}{1 - \left|\frac{V}{V_b}\right|^k} \tag{4.16}$$

where k is an empirical constant between 2 and 6, V is the bias voltage, and V_b is the device's *breakdown voltage*, above which holes contribute to the creation of new pairs, too.

APDs are biased at a voltage $|V|$ close to but below $|V_b|$ in order to maximize gain while retaining a linear response. Typical gains are in the range of 50 to 200; higher gains are difficult to manage due to instability caused by their strong dependence on temperature variations or noise in V [207]. Although the gain is much lower than a PMT's, and usually implies the need for external preamplifiers, this is offset by a quantum efficiency above 80 %. Other advantages of APDs over PMTs are a much lower bias voltage, their compactness and lack of mechanical parts, compatibility with silicon ASIC design processes leading to reduced cost and possibility of integration, and much better immunity against magnetic fields ([194], pp. 28–29).

The main disadvantages over PMTs are a much lower time resolution, sensitivity to temperature, low gain, and noise issues. On one hand, statistical noise i.e. gain variations are higher: in spite of increased quantum efficiency, the avalanche process is inherently much noisier due to the fact that each generated pair experiences a different multiplication factor [203]. On the other hand, the dark current is relatively high due to the effect of leakage currents, which get amplified by the avalanche process unlike in PMTs. The dark current values are similar to those in PMTs, however the SNR is much lower in APDs due to the reduced gain. In some applications, active cooling is needed for the APDs in order to reduce their noise and stabilize their gain.

Silicon photomultipliers (SiPM)

APDs are biased below the breakdown voltage in order to obtain a linear response. Another possibility is to operate them in *Geiger mode* by applying a bias voltage well beyond the breakdown limit, reaching very high gains above 10^5. Under this condition, a single photoelectron can trigger a self-sustained avalanche with a relatively high output current. A quenching mechanism is required in order to stop the avalanche and return the device to a stable stage; the most simple implementation consists in placing a resistor in series with the diode, so that increasing currents cause a voltage drop that eventually reduces V down below V_b and ends the process.

Geiger-mode APDs always exhibit the same output response regardless of the captured optical energy, because a single electron is enough to put the whole device in conduction, masking the signal of any further photoelectrons until the avalanche is quenched. For this reason, they are sometimes called *Single Photon Avalanche Diodes* (SPAD). SPADs are thus digital-mode sensing devices whose output merely indicates whether light was detected or not, whereas APDs have a linear response and their output levels are proportional to the amount of detected optical energy.

Single SPADs are therefore not suitable as PET photodetectors, as they cannot be used to measure the intensity of a scintillation directly. However, it is possible to arrange a very large amount of small SPADs into an array and connect them in parallel, so that each APD cell fires whenever it detects one photon at least, and the output current, which is the sum of the output currents from each cell, is proportional to the amount of fired cells. The output is therefore approximately proportional to the amount of captured photons, the non-linearity arising from cells where more than one photon is absorbed; the probability of such an event can be decreased by reducing the APD cell size. These arrays of SPADs are called *silicon photomultipliers* (SiPM) or sometimes *Multi-pixel Photon Counters* (MPPC).[8]

In a SiPM, the concept of quantum efficiency is replaced by the *photon detection efficiency* (PDE), that takes into account additional factors such as the fill factor (i.e. fraction of active area) and the cell dead time due to previous activations. The raw efficiency is thus lower than in an APD, but the output levels are much higher due to the difference in gain. Noise in SiPMs takes the form of random firing of individual cells, since the output waveform of each cell is the same regardless of the amount of original electron-hole pairs, hence it is specified as a *dark count* rather than a current.

Silicon photomultipliers, being essentially arrays of APDs, share most of the advantages of the latter with respect to PMTs, particularly near-immunity to magnetic fields and the possibility of integration in standard ASIC processes. This allows very compact devices containing all necessary biasing elements; for instance, Fig. 4.9 shows a SiPM cell structure with an integrated polysilicon quenching resistor. However, unlike APDs, their multiplication principle results in extremely fast rise times and time resolutions on par with PMTs may be achieved.

[8] MPPC is actually a trademark of Hamamatsu Photonics K. K..

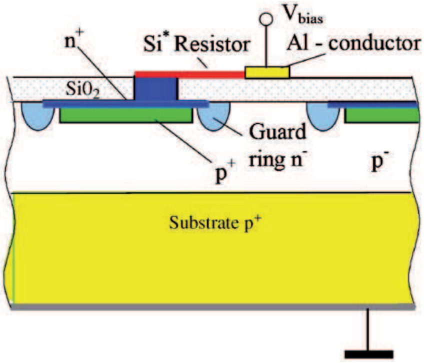


Figure 4.9: Detail of a SiPM cell. Reproduced from [208], © 2005 Elsevier.

Summary and outlook

The first photoelectric tubes that performed photoconversion were released 100 years ago, but the first true PMT that included dynodes for signal amplification came out in 1936 [206]; being the oldest photodetector technology, it was implemented in the first PET scanners [181]. APDs have been employed in PET since the 1980s [209], initially because of their smaller size that allowed one-to-one crystal to photodetector matching. Silicon photomultipliers are a more recent development [210] and, now that their cost has gone down, are progressively replacing APDs in PET applications [211] as they retain most of their advantages while overcoming disadvantages such as reduced time resolution and complexity of the readout circuitry due to reduced gain [212]. Thus, most new scanners use either PMT [213,214] or SiPM [215,216] as photodetectors, although PET systems based on APDs are still being developed [217].

Table 4.3 summarizes and compares the most important properties of all three photodetectors. PMTs are still considered to yield the best performance, particularly due to reduced noise and stability with respect to temperature variations, but cannot be used in new PET-MR designs due to sensitivity to magnetic fields. SiPMs are preferred for PET-MR or in scanners with high channel density because of their compactness, and they offer time resolutions that are on par with, and potentially better than, those obtained using PMTs [218].

The most important new advance in the field of photodetectors for PET is possibly the *digital SiPM* (dSiPM) developed by Frach at Philips [219]. These devices take the compatibility of SiPM technology with ASIC fabrication processes one step further by integrating basic processing electronics in each cell such as counters and TDCs. Integration of the timing and conversion circuitry right at the detector level enables completely digital readout and the prospect of unprecedented time

	PMT	APD	SiPM
Gain	$10^5 - 10^6$	$10^2 - 10^3$	$10^5 - 10^6$
Detection efficiency	∼20 %	∼80 %	30 % to 80 %
Rise time	1 ns	5 ns	1 ns
Noise	0.015 nA to 200 nA	0.05 nA to 30 nA	$< 1\,\mathrm{kHz/mm^2}$
Bias voltage	500 V to 3000 V	100 V to 200 V	30 V to 80 V
MR compatibility	No	Yes	Yes
Temperature sensitivity	Low	High	Moderate
Cost	High	Moderate	Low

Table 4.3: Comparison between the most usual photodetectors employed in PET. Data partially collected from [203, 212, 218].

resolution [220]. The first PET-MR scanners based on digital SiPMs are already under way [221].

4.3 Data acquisition in PET

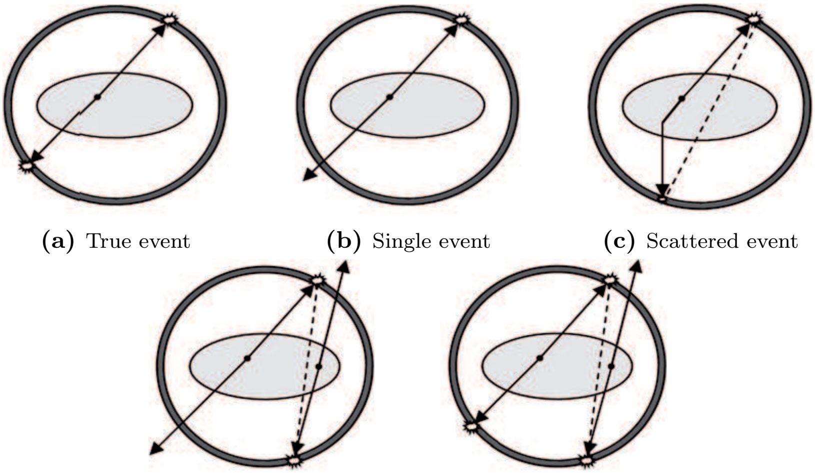
In the last section, the physical principles behind the radiation detectors employed in PET have been presented. They provide pulse signals that contain information about the location and time where annihilation photons are detected in the scanner. These raw data need to be transformed into a list of valid lines of response that can be processed by an image reconstruction algorithm.

The usual process for obtaining a valid LOR is as follows:

- A detector signal is interpreted as an annihilation subevent, i.e. the detection of a single annihilation photon, if the measured energy is within a certain range, called the *energy window*, around the ideal value $E_e = 511\,\mathrm{keV}$.
- A valid PET event is considered to have been detected when exactly two annihilation subevents occur within a time interval known as *coincidence window*.
- The resulting LOR is determined by the position where the two annihilation subevents were detected, and is valid only if it intersects the valid Field of View (FOV) that has been designated for the scanner.

Figure 4.10 illustrates the most important types of event as classified by this process:

- *True events* (Fig. 4.10a) have both annihilation photons correctly detected and yield a valid LOR.


(a) True event (b) Single event (c) Scattered event

(d) Random event (e) Multiple event

Figure 4.10: Types of event in a PET detector. Reproduced from [22] with permission of the author.

- *Single events* (Fig. 4.10b) take place when only one of the annihilation photons is detected, and the other one crosses through the scintillator without interaction. These events are rejected by the time coincidence window.

- *Scattered events* (Fig. 4.10c) occur whenever one of the annihilation photons suffers Compton scatter before arriving at the detector. The LOR determined by the position of detection of both photons is invalid as shown in the figure. However, the event is rejected because the scattered photon has a lower energy and is filtered by the energy window.[9] Compton scatter may also take place in the scintillator itself; in this situation the LOR would be valid, but the event is usually rejected anyway due to the difficulty of determining whether this is the case.

- *Random events* (Fig. 4.10d) are the result of two single events happening at the same time. Two valid photons are detected within a time coincidence window, but they belong to different radiotracer decays and the resulting LOR is not valid. This event is registered as valid anyway and contributes to noise in the reconstructed image.

[9] The event is accepted if the scattered photon's energy is within the energy window, however this implies that the energy loss due to Compton scatter is small, and hence by (4.5) the deflection angle is small too, so the detected LOR passes close to the original decay position and the resulting error is relatively small.

- *Multiple events* such as triple events (Fig. 4.10e) take place when two or more decays occur simultaneously and three or more annihilation photons are detected within a coincidence window. This event is usually rejected due to the impossibility of determining which of the photons form valid pairs.

The three physical quantities used to filter and compute LORs (energy, time difference, and position) define the three fundamental resolutions that can be used as performance parameters of the detector subsystem in a PET scanner: *energy resolution*, *time resolution* (or *coincidence resolution*), and *position resolution*.[10] These values are defined as the minimum separation that two different events (or subevents) need to display in the given physical quantity in order for them to be perceived as actual different events by the detector.

This section describes the intricacies of all three fundamental resolutions in detail. After that, the essentials of *Time-of-Flight* (ToF) PET are exposed, an acquisition and reconstruction modality where LORs are substituted by more refined estimations of decay locations given by the time difference between annihilation subevents.

4.3.1 Energy resolution

Energy resolution is the most simple of the three resolution concepts. The energy of detected photons is typically estimated as either the amplitude or the integral (i.e. charge) of the resulting photodetector pulse. This estimation is subject to errors, mainly of statistical nature: variations in the scintillator light yield, in the angular distribution of optical photons and hence of the ratio of optical energy actually reaching the photodetector surface (as opposed to being absorbed at the sides), quantum efficiency, photodetector gain variation, etc. Hence, for a given 511 keV photon, its measured energy is given by a probability distribution; energy resolution is determined by the width (typically FWHM) of the peak of that distribution, and is usually specified as a percentage of the peak value 511 keV.

More than 80 % of all decay events suffer Compton scatter in one or both annihilation photons in a typical PET scan [223]. Therefore, reduction of the energy window has a major impact on the discrimination of valid LORs. The size of the energy window establishes a trade-off between the quality of the reconstructed image (rejection of invalid events) and the scanner sensitivity (acceptance of valid events); enhancing the energy resolution thus allows a tighter energy window to be implemented for improved image quality without sacrificing sensitivity.

[10] This is different from the spatial resolution of the whole PET scanner, which would be the minimum separation between two different radiotracer samples that allows them to be told apart from each other *after image reconstruction*. This value depends on the position resolution at the detectors but also on the size of the scanner and fundamental limits like positron range [222].

4.3.2 Position estimation

The light produced by a scintillation event is typically collected at the back side of the crystal by covering its surface with arrays of photodetectors or position-sensitive detectors like PSPMTs. This results in a complex detector with an active surface that is tessellated into separate regions with an associated output signal, the energy estimation from each output current being proportional to the amount of optical photons impinging on its associated region. Hence, the set of outputs provides a bidimensional histogram of the spatial distribution of the optical energy arriving at the back of the scintillator volume. *Position estimation* in PET is the process of translating these histograms of collected photons into the position where the scintillation took place. This process is independent for each annihilation photon, and correspondence between photons in order to form valid pairs is determined using times of arrival.

Estimation of the interaction position coordinates is always handled differently for the bidimensional coordinates x, y along the photodetector plane and for the *Depth of Interaction* (DOI) z that represents the distance between the scintillation and the photodetector. In fact, many PET systems ignore DOI altogether. Estimation of bidimensional position, i.e. of the projection onto the photodetector will be considered first and the issue of DOI estimation will be handled in a later subsection.

Pixelated and continuous detectors

There are two opposing paradigms regarding the layout of scintillators that result in detectors with fundamentally different philosophies for position estimation.

The most common type of detector module is the *pixelated detector*, the classic example being Casey and Nutt's block detector [224]. It consists of a discrete matrix of scintillator crystals with a small base relative to their height that are optically isolated from each other. These may be formed by either separate crystals or a single slab of scintillator material with partial cuts on its sides. In any case, they act as a network of independent scintillators by introducing reflective material between them, so that the light produced by a scintillation in one of the crystals is confined to it. Photodetectors are arranged on the back side of the crystal matrix; in the most extreme case, photodetectors (either separate, or different PSPMT regions) are coupled to crystals on a one-to-one basis, so that each scintillation generates pulses in just one detector output.

As an alternative, *continuous detector* modules may be employed [225], consisting of a single large scintillator crystal coupled to a matrix of photodetectors. In this case, a scintillation produces light that is detected by several photodetectors. Fig. 4.11 outlines the difference between both detector philosophies by showing

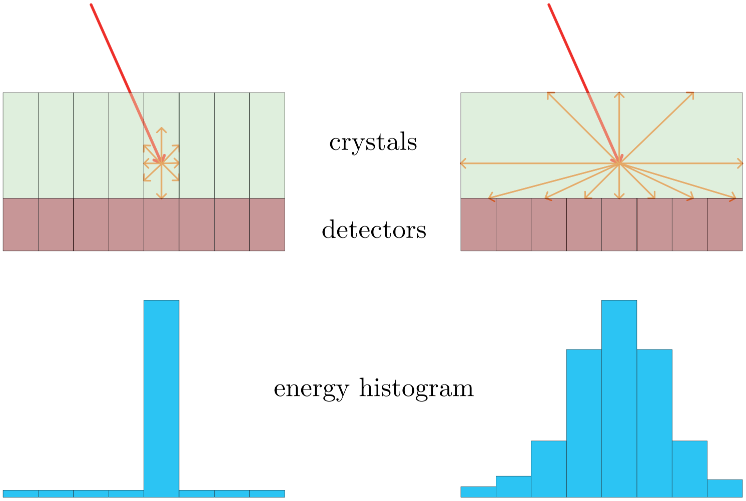


Figure 4.11: Outline of a pixelated (left) and a continuous (right) PET detector. Impinging annihilation photons are shown in red and optical photons are shown in orange.

side views of the detector blocks, an example of scintillation and the resulting one-dimensional sections of the energy histograms. Position estimation in the pixelated example is reduced to the task of checking which one of the outputs is active, and is trivial in the one-to-one case but may be more complicated when the number of photodetectors is smaller [224]. The signals generated by a continuous detector always require further processing.

In a pixelated detector, position estimation amounts to crystal identification, hence position resolution is independent of the processing front-end and equal to the crystal size. Another advantage of pixelated detectors is the reduction in spatial pile-up [226] for slow scintillators, as scintillations occurring in different crystals can be identified concurrently. There are however several major drawbacks that can be solved by using continuous detectors:

- Pixelated detectors of equivalent area are much more expensive than continuous ones due to the fabrication cost of crystal cuts.
- Detection efficiency is lower for an equivalent crystal size because of the volume loss between active crystals and the small area of contact with the photodetectors [227].
- The aspect ratio of individual crystals implies a smaller solid angle of the scintillation light reaching the photodetectors at the back side, leading to increased statistical noise and reduced energy resolution.

Continuous PET detectors are a relatively new development because they require a smaller detector pitch and more elaborate position estimation schemes for similar resolution ([189], pp. 22–24), but their aforementioned advantages make them an increasingly interesting choice as PSPMTs and SiPM arrays and digital front-end electronics become the norm.

A precise definition of position resolution in a continuous detector is given by the *Point Spread Function* (PSF), i.e. the probability distribution of the estimated positions for a given interaction position of the annihilation photon. As in the case of energy, the detector resolution is usually given by the width (typically FWHM) of the PSF, and depends on the scintillation position.[11] Systematic error, i.e. the distance between the actual position and the PSF mean, is another important parameter; it may be compensated against, but doing so increases the effective PSF width.

Channel reduction and Anger logic

The spatial histogram provided by the photodetectors contains enough information to reconstruct the coordinates of the scintillation position with a certain resolution given by physical limitations. However, these values share much redundant information, and the acquisition of a large number of signals is complex from the point of view of its electronic implementation, as it requires the routing or multiplexing of all of them, their integration and conversion to the digital domain, resulting in increased cabling, circuit area and processing requirements as well as crosstalk between detector signals. For these reasons it is desirable to reduce the number of detector signals while keeping enough information to reconstruct the interaction position with acceptable resolution.

The simplest and most extended way to reduce the number of detector signals consists in interconnecting them through a passive analog circuit that provides the target signals with minimal delay. Resistor networks are typically used that yield a reduced number of currents which are linear combinations of the individual currents generated by the photodetectors. These circuits are known as *Discrete Positioning Circuits* (DPC) or *Charge Division Circuits* (CDC).

Let an arbitrary DPC be given with N current inputs $i_j\left(t\right)$ and a number of output currents that need to be determined. Assuming that the outputs are connected to constant, real impedances, this is equivalent to the determination of certain currents $J_k\left(t\right)$ in a resistor network. The Superposition Theorem implies that

$$J_k\left(t\right) = \sum_{j=1}^{N} M_{j,k} \cdot i_j\left(t\right) \tag{4.17}$$

[11] The PSF of a pixelated detector is a discrete distribution, so its resolution cannot be measured as FWHM, but it may be expressed in terms of its standard deviation.

where $M_{j,k}$ are real constants expressing the current gain from the j-th input to the current of interest J_k. Integrating over the scintillation length (or taking the amplitude, under the assumption of homogeneous pulse shape) yields

$$Q_k = \sum_{j=1}^{N} M_{j,k} \cdot G_j \cdot E_j \tag{4.18}$$

for the charge at the k-th output, where E_j is the optical energy collected by the j-th detector and G_j is its gain in terms of charge per energy as expressed in (4.14). DPCs thus have discrete outputs that are linear combinations of the collected energy histograms, where the weights have a deterministic component $M_{j,k}$ given by the particular resistor network and a random component G_j given by the photodetectors.

The simplest example dates back to the very first gamma camera design [173] and is usually called *Anger logic* in its honor in the context of medical imaging, although it is more commonly called *Center of Gravity* algorithm (CoG) in other applications [228, 229]. This amounts to designing a DPC in such a way that the values $M_{j,k}$ turn (4.18) into formulae that can be combined to obtain the centroid (equivalently, the first moment, or the spatial average) of the energy histogram $\{E_j\}$.

Fig. 4.12 depicts the situation in a one-dimensional case where detection is carried out by N photodetectors laid out uniformly along one crystal axis, whose outputs are connected to a chain of resistors with the same value [230]. The analytical expression of output currents J^- and J^+ can be obtained easily by superposition. Assuming that only the j-th input is active, the network acts like a current divider, so the currents through the loads at both ends are

$$J_j^+ (t) = \frac{j}{N+1} i_j (t) \, , \; J_j^- (t) = \left(1 - \frac{j}{N+1} \right) i_j (t) \tag{4.19}$$

i.e. the current gains are $M_j^+ = \frac{j}{N+1}$ and $M_j^- = 1 - \frac{j}{N+1}$, and so

$$Q^+ = \frac{1}{N+1} \sum_{j=1}^{N} j \cdot G_j \cdot E_j , \; Q^- = \frac{1}{N+1} \sum_{j=1}^{N} (N+1-j) \cdot G_j \cdot E_j . \tag{4.20}$$

Assuming constant G_j, the sum of both currents is proportional to the total energy $\sum E_j$ detected for this scintillation, and its difference is proportional to the unnormalized first moment of the energy distribution. The centroid is thus given by

$$\hat{x} = \frac{Q^+ - Q^-}{Q^+ + Q^-} = \frac{2}{N+1} \frac{\sum_{j=1}^{N} \left(j - \frac{N+1}{2} \right) \cdot E_j}{\sum_{j=1}^{N} E_j} \tag{4.21}$$

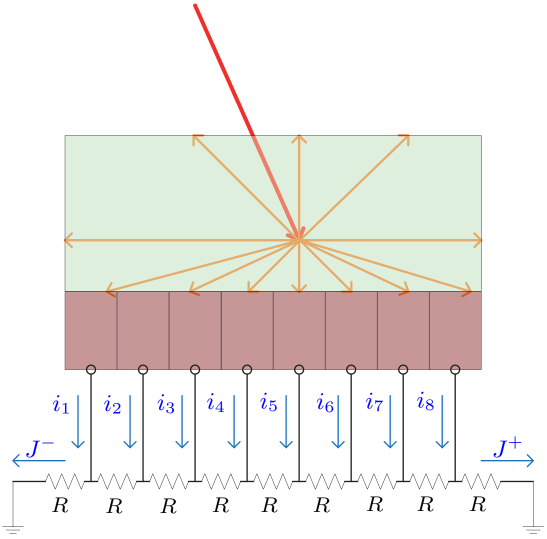


Figure 4.12: One-dimensional DPC implementation using a proportional resistor chain and connection to 8 photodetectors. The centroid is given by the formula $\left(J^+ - J^-\right) / \left(J^+ + J^-\right)$.

normalized to the interval $[-1, 1]$, with the value 0 corresponding to the center of the crystal.

The typical situation includes a two-dimensional matrix layout of photodetectors. The previous scheme can be extended to this case by connecting several horizontal chains through vertical chains at their ends [230], as depicted in Fig. 4.13. Four output currents are thus obtained at the matrix corners, whose sum is again proportional to the total energy. A detailed study found in ([194], pp. 66–69) and revised in ([174], pp. 41–46) shows that the centroid is given by

$$\hat{x} \propto \frac{Q^{++} + Q^{+-} - Q^{-+} - Q^{--}}{Q^{++} + Q^{+-} + Q^{-+} + Q^{--}}, \; \hat{y} \propto \frac{Q^{++} - Q^{+-} + Q^{-+} - Q^{--}}{Q^{++} + Q^{+-} + Q^{-+} + Q^{--}} \tag{4.22}$$

as long as the resistor values satisfy certain conditions; specifically, the resistors at the ends of the horizontal chains need to be tuned in order to linearize the equivalent impedance at the connections with the vertical chains. A particular, well-known case for an 8 × 8 matrix is the *Siegel network* described in [231].

Anger logic is an extremely simple and fast way to reduce an arbitrary number of output channels to just 4 using a discrete, entirely passive analog implementation, but it is subject to systematic errors besides the statistic errors given by variations in G_j and electronic noise. One of the error sources is the discretization of the

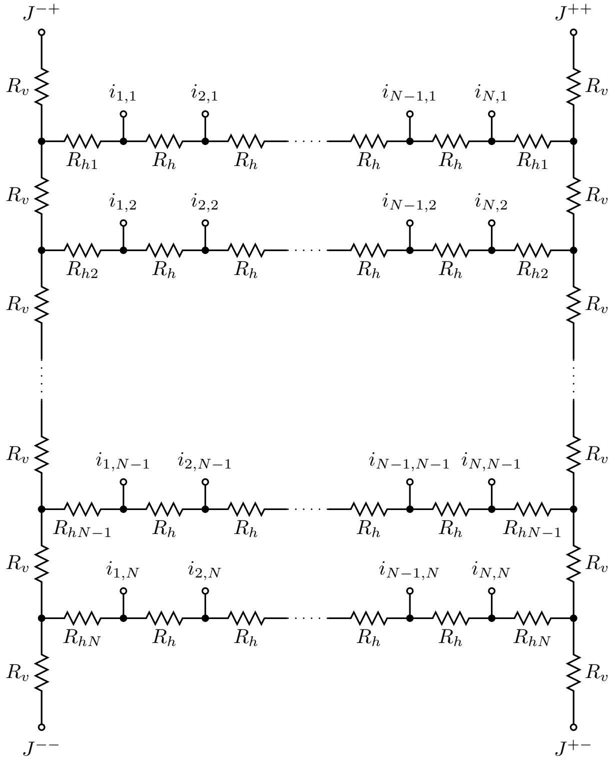


Figure 4.13: Two-dimensional DPC based on proportional resistor chains with $N \times N$ current inputs and four Anger outputs.

energy histogram $\{E_j\}$ [228, 229]. However, the major source of systematic error is the so-called *border effect* introduced by the fact that the transducers are of finite size: the centroid is a reasonable estimation of the bidimensional interaction position near the center of the crystal[12] but its exactness worsens as the scintillation point gets closer to the crystal border. The reason behind this effect is the fact that the histogram of incident energy is truncated, as it only collects the light output corresponding to the solid angle covered by the detector surface. Most of the light reaching the crystal border is lost, and only a small fraction of it is reflected, recovering some sensitivity but distorting the energy histogram.

Fig. 4.14 describes the situation, showing the complete energy histogram for an ideal, infinite detector and highlighting the actual measured part. By definition, the interaction point is between the crystal center and the closest border; hence, by dropping the energy beyond the border from the centroid formula, the resulting

[12] It is, in fact, completely accurate under the assumption of infinite crystals and detectors, isotropic light generation, and constant photodetector gain.

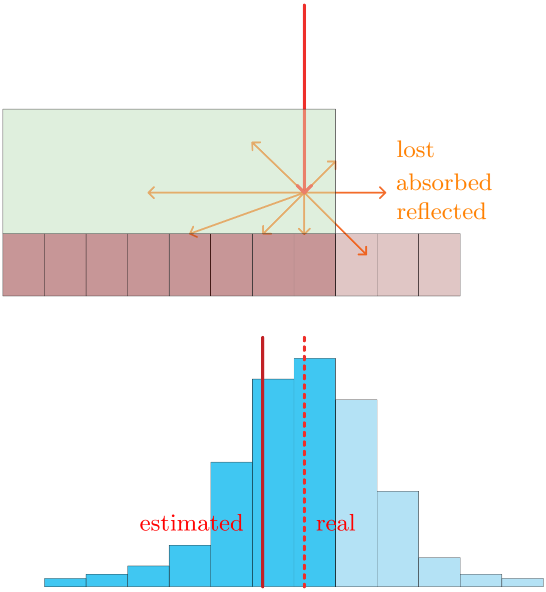


Figure 4.14: Explanation of the border effect. Shaded photodetectors and their outputs are absent in the real detector. The position estimations for the complete (dashed line) and truncated (solid line) energy histogram are superimposed on it.

estimation is always closer to the crystal center than the real interaction position. The effect thus always results in a compressed image like those shown in Fig. 4.16 (left) that is characteristic of detectors based on Anger logic without later correction. Moreover, the compression is not linear but rather becomes more severe as the interaction point gets closer to the border, because the truncation then happens closer to the interaction point. Since the compression effect is monotonic, it is possible to apply a linearity correction map in order to cancel the systematic errors, but this results in the broadening of the PSFs near the border and a vastly decreased position resolution, as outlined in Fig. 4.15. Due to this effect, the useful FOV of traditional PET detectors is limited to a reduced area around the center of the detector.

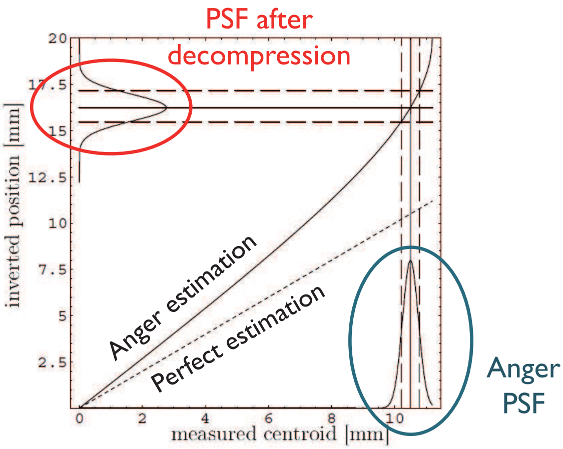


Figure 4.15: The solid line represents the Anger mapping for decompression. The ideal PSF on the bottom right is converted into the widened PSF on the top left after linearity correction. Adapted from [194] with permission of the author.

Alternative estimation methods

Several alternative methods for position estimation have been proposed and implemented in order to correct the shortcomings of Anger logic. Two of them are briefly described here: neural networks and statistics-based methods.

For the first approach, the goal is to model the interaction position as a direct function of a set of measured values, using models with a very high number of free parameters. Ideally, the whole set of photodetector outputs is not used directly but rather a smaller number of variables such as those given by a DPC that still contain enough information to be able to reconstruct the interaction point. Artificial neural networks are particularly well suited for this application due to their capability of extracting implicit relationships between variables given a discrete set of measurements. These models need to be *trained*, i.e. calibrated using a set of simulations or measurements describing correct input-output pairs (i.e. true interaction position and resulting outputs). Neural networks have been evaluated as either position estimators or direct LOR estimators by several groups [232–234] including EDNA [235–237] in the last two decades, since the popularization of FPGA technology that allows their real-time implementation. They achieve better resolution than Anger logic using the same set of 4 values and real-time hardware realizations are easy and inexpensive [235, 238].

Statistics-Based Positioning (SBP) schemes are applications of the more generic *Maximum Likelihood* (ML) method. These algorithms consider the probability distribution of the set of observable measurements i.e. detector outputs for each possible interaction point, and then, given a set of measurements, estimate position as the interaction point for which the probability of generating the observed measurements is highest. In equation form, the position estimation is given by

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p\left(\mathbf{m}|\mathbf{x}\right) \tag{4.23}$$

where $\mathbf{x}$ is the interaction position, $\mathbf{m}$ is the detector output, and $p\left(\cdot|\mathbf{x}\right)$ is the conditional probability distribution. It is necessary to define a model for p, and there are two different ways to accomplish this: either define a physical model for the *Light Response Function* (LRF) and fit the parameters through calibration [239,240], or use experimental measurements or simulations at specified interaction points and interpolation in between [241, 242]. Individual detector outputs are assumed to follow either a normal or a Poisson distribution in order to simplify the resulting expressions, and maximization is achieved using an iterative algorithm.[13] SBP methods are possibly the most precise position estimation techniques, but have two important drawbacks with heavy repercussions on system cost: on one hand, they usually require the digitization of a higher number of channels; on the other hand, iterative maximization has a high computational cost that typically forbids real time implementations.

Both methods discussed above share the property that they aim at a full detector model considering all components instead of just the resistor network, hence they naturally take into account and compensate for effects like non-uniform, non-deterministic photodetector gain, defects in scintillator crystals, or non-ideal resistor networks. As a counterpart, both require a complex calibration process that is difficult to automate. In general, neural networks provide a less precise estimation in exchange for a much more lightweight implementation. A graphic comparison between both methods is included in Fig. 4.16.

DOI estimation

The first PET detectors only considered the coordinates of the interaction position that belong to the photodetector surface plane, disregarding depth of interaction. Since a 3D point is required anyway for the determination of a LOR, this is equivalent to the assumption that all interactions take place at a fixed depth, typically at the external crystal surface, thus equating interaction position with incidence position. This assumption is reasonable for thin crystals, but such a setup has the disadvantage of severely decreased sensitivity i.e. photon stopping power. Otherwise, obliquely incident annihilation photons give rise to a *parallax error* in the computation of the LOR due to DOI variation. The situation is outlined in Fig. 4.17 (left), as the centroid, while correctly estimated, does not correspond to the

[13] More details on the ML method will be provided in §4.4.4 in the context of image reconstruction.

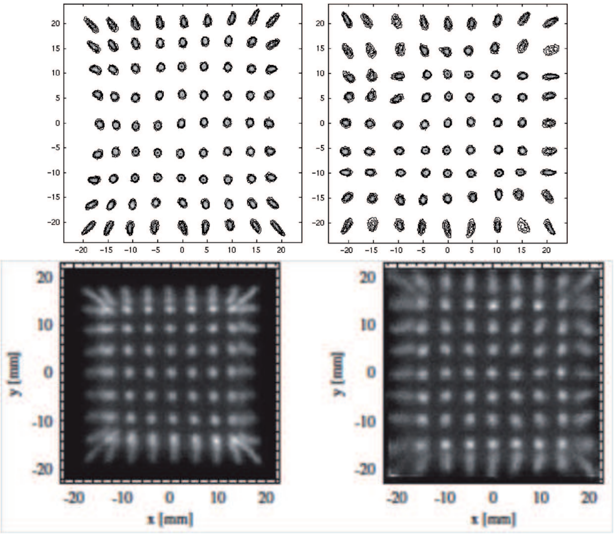


Figure 4.16: Reconstructed images i.e. 2D position histograms of a 9 × 9 grid of point sources distributed uniformly along a square scintillator. On the top row, the left image corresponds to Anger logic and the right one to neural network estimators. On the bottom row, the left image is obtained with Anger logic and the right one using SBP. Reproduced from [235] and [239], respectively. © 2006 IEEE and © 2009 Elsevier.

incidence point if the upper crystal border is taken as the constant DOI reference. Fig. 4.17 (right) shows how the LOR estimated by disregarding DOI does not intersect the true decay position.

Parallax error scales with detector thickness and with the angle of incidence of the annihilation photons. In scanners without DOI estimation, parallax error must be kept within bounds by using thinner crystals or restricting the useful FOV of the scanner, as its effect is less important near the center of the sensitive volume due to reduced variation of incidence angles. The only way to reduce parallax error without sacrificing detection efficiency is to implement some scheme for DOI measurement in the detector. Even coarse DOI estimations with a resolution in the order of 5 mm are sufficient to substantially increase the whole detector spatial resolution [243].

The most important DOI estimation techniques are discussed next. They can be classified into three broad groups:

- A number of techniques are based on the stacking of several discrete layers of scintillator crystal per detector. *Phoswich*[14] detectors (Fig. 4.18a) use several layers of various scintillator materials with different properties and

[14] A portmanteau of *phosphor* and *sandwich*.

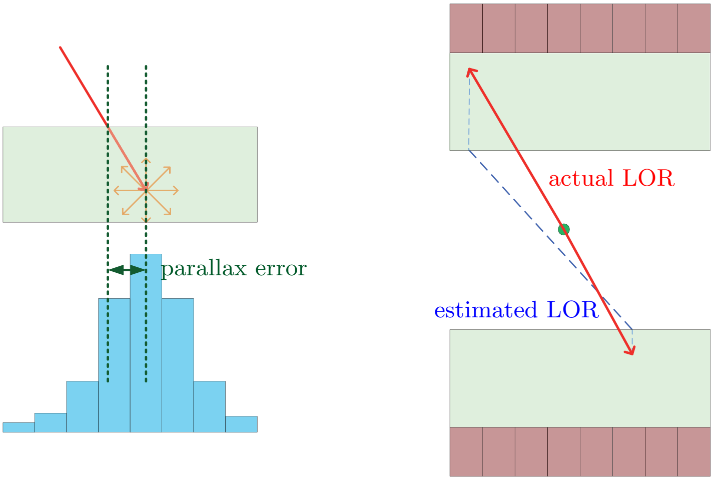


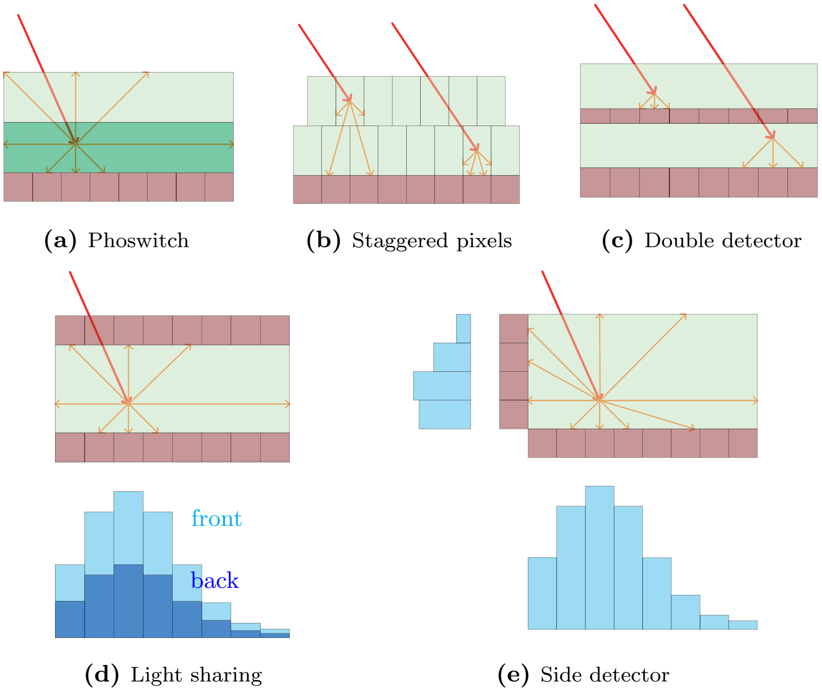
Figure 4.17: Definition of parallax error in detectors without DOI measurement (left) and impact on LOR estimation (right).

rely on pulse shape discrimination to identify the layer where the scintillation took place, e.g. measuring pulse width from scintillators with different decay time constants [244, 245].[15] Another possibility is using layers of pixelated detectors with a relative displacement (Fig. 4.18b) that allows different layers to be identified by the 2D position or the number of active pixels [247]. One further option is to stack two independent complete detectors as shown in Fig. 4.18c [248]. In all three cases, the techniques require discrete crystal layers and only allow discrimination between those layers.

- Other DOI measurement techniques employ several photodetectors in order to obtain a continuous estimation. For instance, *light sharing* detectors are based on the deployment of photodetectors on both the back and the front side of the scintillator crystal (Fig. 4.18d) and estimate DOI depending on the ratio of total energy collected in both sides [249, 250]. Also, continuous cubic detectors have been proposed with photodetectors coupled to side borders (Fig. 4.18e) in order to measure DOI using 2D position estimation techniques [232]. In these cases, DOI estimation requires the deployment of additional photodetectors and imply an increased detector cost.

- There is one final approach that has been studied thoroughly by Lerche et al. at the EDNA group and does not involve additional crystals or detectors. It is based on the observation that the shape of the collected energy histogram in a continuous detector, more specifically its relative width, de-

[15] Alternative monolithic phoswich detectors have been developed recently [246].



(a) Phoswitch **(b)** Staggered pixels **(c)** Double detector

(d) Light sharing **(e)** Side detector

Figure 4.18: Techniques for DOI estimation using several crystal layers (first row) or detectors (second row).

pends on DOI [251]. This observation is illustrated in Fig. 4.19: the energy collected by each detector is proportional to the solid angle of the scintillation that it covers; the closer the detectors are to the scintillation point, the larger the solid angle covered by the detector directly below and thus the larger the differences between adjacent solid angles will be, resulting in a narrower light distribution. An estimation of the DOI can thus be obtained from the standard deviation, or equivalently from the second moment, of the energy histogram [252]. It has been shown how modified DPCs can simultaneously yield the four Anger output currents and a fifth output that yields the width information for DOI estimation at essentially the only added cost of an additional acquisition channel [230].

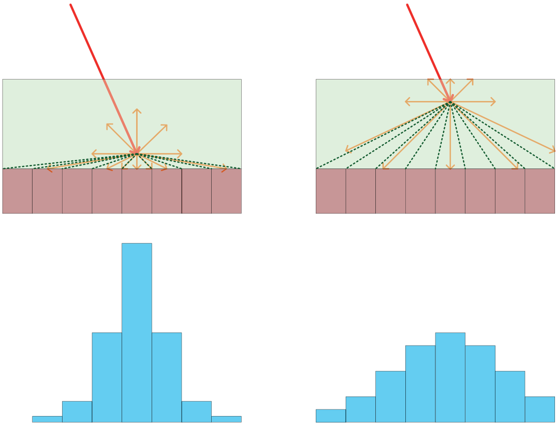


Figure 4.19: Illustration of the relationship between depth of interaction and width of the energy histogram. Angles for each detector are highlighted.

Integrated front-end for position estimation: PESIC

The typical implementation of DPCs using discrete resistors connected directly to the photodetector outputs results has one important drawback when a realistic situation is considered that takes the output capacitances of photodetectors into account. It turns out that the frequency response is not uniform for all current inputs, leading to position-dependent artifacts in the sampling of each output value unless a slow acquisition mode is implemented, either charge integration or amplitude acquisition with a slow shaping below the frequency response's bandwidth [253].

This problem may be circumvented if the bandwidth is increased by reducing the equivalent capacitance seen at the network inputs, for example by introducing preamplifiers between the photodetectors and the DPC. A discrete implementation of such a scheme is possible but impractical due to increased circuit area requirements; integration into an ASIC device is a preferable solution.

One such device is PESIC [254], an integrated circuit developed by Herrero et al. at EDNA for the readout of position sensitive detectors arranged in an 8 × 8 matrix. PESIC contains an integrated Siegel network and a preamplifier for each current input. The resistor network is modified as suggested by [230] in order to extract a fifth output for DOI estimation besides the usual Anger outputs.

Integrated front-end for position estimation: AMIC

The possibility of integrating the DPC and related electronics opens up the way for additional front-end improvements and capabilities. A resistive DPC is characterized by equation (4.17), representing the output signals as fixed linear combinations of the current inputs. A generalization of the DPC concept is therefore possible by implementing this transfer function directly by any other means; for instance, current weighting and addition may be accomplished using transistor-based current mirrors. This allows for a straightforward realization of arbitrary weighted sums without having to come up with complicated resistor network extensions for each particular case. Moreover, such a circuit may be easily adapted to provide adjustable weights, i.e. an implementation of (4.17) where the coefficients $M_{j,k}$ can be digitally programmed [255].

This generalization is the main idea behind AMIC [256, 257], a front-end ASIC designed at EDNA with 64 current inputs that generates up to 8 current outputs with 8-bit, digitally programmable positive coefficients. Programmability offers great flexibility in the usage of the circuit. For instance, an AMIC device may be programmed to implement the four Anger outputs by programming their coefficients directly; however, the Anger method may be implemented using only three signals, corresponding to both numerators and the common denominator in (4.22). Alternatively, other linear combinations may be computed that allow a partial reconstruction of the energy distribution that maintains the parameters of interest, for example the low-order statistical moments [255], or those determined by a principal component analysis. AMIC may also be used to implement the first stage of a neural network estimator. Additionally, the gain mismatch between photodetectors may calibrated and compensated against by introducing a G_j^{-1} factor in each $M_{j,k}$ coefficient [258].

AMIC maintains the extra benefit provided by PESIC of integrated preamplifiers that yield current outputs with high bandwidth even for relatively high photodetector output capacitances, making it compatible with PMTs and several families of SiPMs with low equivalent output capacitance. A second version has been designed and fabricated that expands the range of acceptable capacitances further and admits most SiPM models [259]. A third version has also been designed that allows for virtually unlimited capacitance values, but it remains unreleased.

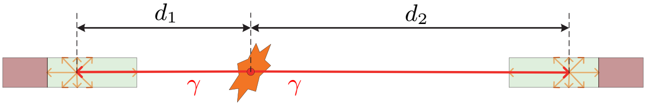


Figure 4.20: Diagram of a line of response defined by two detectors and distance from the decay location to them.

4.3.3 Coincidence detection

Valid LORs are determined by pairs of annihilation photons originating from the same disintegration. Since they are emitted at the same time, they are expected to hit the detector and be detected almost simultaneously. There is, however, a range of time differences between both scintillations that still corresponds to completely valid events. In fact, the decay event may take place anywhere along the LOR as shown in Fig. 4.20, and since the photons travel at a finite speed, each position will result in a different time difference

$$t_d = \frac{d_1 - d_2}{c} \tag{4.24}$$

where d_i are the distances between the disintegration and the interaction position at the detectors and c is the speed of light.

A non-empty *coincidence window* is therefore defined in PET scanners, such that detected scintillations are considered to belong to the same decay event if their time difference falls within the window. Because of scanner symmetry, the coincidence window is usually centered at 0, so the validity of events is given by the condition

$$|t_d| < T_W \tag{4.25}$$

for some T_W such that the coincidence window length is $2T_W$. False positives are of course possible in the form of random events (Fig. 4.10d). The worst case time difference that may result from a valid event is given by the diameter D of the useful FOV in the scanner, hence the ideal coincidence window would be given by

$$T_W = \frac{D}{c}. \tag{4.26}$$

For reference, this value is in the range of 3 ns for rings with 1 m diameter.

Time resolution and coincidence resolution

The time of interaction cannot be determined with infinite accuracy due to the physical limitations of the detector. The situation is analogous to the cases of energy and position estimation in that the processes behind time assignment in a

PET detector contain random components and the detection of different annihilation photons originating at the same point and traveling with the same direction may result in different time estimates that can be described with a probability distribution. Systematic errors in the mean of the distribution may be compensated against by way of different, more or less time consuming alignment methods [113, 260], but variance cannot be calibrated away and determines the time resolution in the system.

Two different resolution values can be introduced for timing. On one hand, *coincidence resolution* σ_c is defined as the width (either FWHM or rms) of the distribution of detected time differences between identical interactions, i.e. photon pairs with the same starting point and respective directions; the mean of such a distribution should be given by (4.24).

On the other hand, the *time resolution* σ_t of a detector is given by the distribution of the time of detection of a single annihilation photon under the same conditions, i.e. generated at a fixed time t_0 at the same position with the same direction. Due to independence between events, only relative instead of absolute timing should be considered, hence the need to fix a starting time t_0 for each event.

The relationship between both values is apparent: if the annihilation photons corresponding to an event starting at t_0 are detected at times t_1 and t_2, the measured time difference is

$$t_d = t_1 - t_2 = (t_1 - t_0) - (t_2 - t_0)\,. \tag{4.27}$$

It is reasonable to assume that the processes for time reference assignment in separate detectors are independent, hence their variances can be added and

$$\sigma_c^2 = \sigma_{t1}^2 + \sigma_{t2}^2. \tag{4.28}$$

Under the additional hypothesis that all detectors are identical and have equal time resolution σ_t, one finally obtains

$$\sigma_c = \sqrt{2}\,\sigma_t. \tag{4.29}$$

If time differences are assumed to follow a normal distribution as usual, then this relationship applies to FWHM resolution as well.

Coincidence resolution is the important parameter with regard to the coincidence window; as usual, $\sigma_{c\ \mathrm{FWHM}}$ is considered as the minimum separation between subevents (on separate detectors) that allows them to be treated as different subevents. Ideally, this value is much lower than (4.26) and the ideal coincidence window may be implemented. This may not the case in smaller PET scanners or low-performance systems, where the coincidence window is usually set to $T_W = \sigma_{c\ \mathrm{FWHM}}$ [261].

The main advantage provided by an improved coincidence resolution is a reduction in the rate of accepted random events. In fact, it is easily seen that the rates of random and multiple events are proportional to T_W while the rates of trues, singles and scattered events do not depend on the coincidence window at all ([188], pp. 29–30). Hence, for a small PET scanner with T_W greater than (4.26), improving σ_c allows a tightening of the coincidence window and a reduction in the rate of invalid LORs that are accepted. Additionally, stricter filtering may reduce the processing requirements of subsequent electronics and allow an increase in sensitivity, depending on the system architecture. Both effects imply an enhanced signal-to-noise ratio in reconstructed images. Even for σ_c below the coincidence window, an improvement in coincidence resolution can provide additional benefits as will be explained in §4.3.4.

Factors influencing time resolution

As stated above, infinite timing precision that allows the detection of true simultaneousness cannot be achieved due to physical limitations. Coincidence resolution is determined by single-detector time resolution σ_t as implied by (4.28) and (4.29) instead. The value of σ_t is in turn given by the cumulative effect of timing degradation provided by each detector component. Since the contributions from separate elements are independent, an intrinsic time resolution can be assigned to each of them that yield the combined resolution value as

$$\sigma_t = \sqrt{\sigma_{t\ \mathrm{sc}}^2 + \sigma_{t\ \mathrm{pd}}^2 + \sigma_{t\ \mathrm{el}}^2} \tag{4.30}$$

where $\sigma_{t\ \mathrm{sc}}$, $\sigma_{t\ \mathrm{pd}}$ and $\sigma_{t\ \mathrm{el}}$ are the intrinsic time resolution of the scintillator crystal, the photodetectors, and the readout and timing electronics, respectively, which are the main detector components; these resolution values may be composed of several independent contributions themselves [262].

In order to derive expressions for the intrinsic time resolution in a scintillator crystal, i.e. the best case resolution with ideal photodetectors and readout electronics, the assumption is made that the time reference being assigned corresponds to the detection of a particular optical photon, say the N-th photon out of a total of R photons being emitted. Here, R depends on the scintillator light yield and the energy of the incoming photon (511 keV) and is assumed to be Poisson distributed. This situation corresponds approximately to the detection of the time when the photodetector output pulse crosses a certain threshold value that is directly related to N. The time of detection, i.e. of arrival at the ideal photodetector, is split into two components: the time of generation of the optical photon, and its propagation time from the originating point to the detector; both are independent due to the randomness of each photon's direction.

The time distribution of photon emission in a scintillation is determined by the time variation of light intensity. The classic approach is to consider the exponential approximation (4.10) and derive the distribution of the time of emission of the N-th photon from it [263]. The variance $\sigma_t^2(N)$ of the emission time with respect to the start of the scintillation may then be given by the closed formula [264]

$$\sigma_t(N) = \tau_1 \sqrt{\sum_{k=0}^{N-1} \frac{1}{(R-k)^2}}. \tag{4.31}$$

With the assumption that $N \ll R$, the much simpler estimation

$$\sigma_t(N) \approx \frac{\tau_1}{R}\sqrt{N} \tag{4.32}$$

holds. This approximation is valid both for a fixed value of R and for Poisson distributed R, in which case its mean value is to be used in the formula.

While the exponential approximation is accurate enough for slower crystals, it has been shown to be inadequate for the prediction of photon detection statistics in faster scintillators like LSO or $\mathrm{LaBr_3}$, where an expression like (4.11) must be used for intensity that takes the rise time into account. Unfortunately, application of the above treatment to the new expression for intensity does not lead to a simple closed formula like (4.32) [265]. A simpler approach is to model the effect of rise time τ_r as a separate quadratic contribution to time resolution ([266], p. 51).

The final component in scintillator resolution is the variation in propagation time. This value depends on the speed of light in the scintillator, i.e. on its refractive index, and also on its size: variation is obviously larger in large, continuous crystals than in small, pixelated detectors. Other nontrivial factors like crystal geometry and coating apply due to the fact that several propagation modes are possible involving one or more reflections at the crystal borders. An exact expression of propagation time variation is therefore impractical. The estimation

$$\sigma_{t\ \mathrm{prop}} \approx \frac{n}{6c_0}L \tag{4.33}$$

is proposed in ([266], pp. 40–46) where c_0 is the speed of light in vacuum, n is the refractive index of the crystal, and L is the maximum possible path length, considering paths with reflections too.

The photodetector modifies the contribution due to the statistics of photon emission through its collection efficiency, since only the photons that are detected may be used for timing. On one hand, the detector surface may not collect the whole scintillation but rather a fraction $\omega = \Omega/4\pi$ of it, where Ω is the solid angle of the scintillation that is covered by the photodetector.[16] On the other hand, each photon hitting the detector has probability η of being actually detected, and this an

[16] Both directly and indirectly via reflections at the crystal borders depending on geometry and coating.

independent process for every photon, i.e. a binomial selection process. Since the binomial selection of a Poisson distributed variable is again Poisson distributed ([267], pp. 637–639), the intrinsic resolution (4.32) for detection of the N-th photon is modified as

$$\sigma_t\left(N\right) \approx \frac{\tau_1}{R\eta\omega}\sqrt{N} \tag{4.34}$$

provided that N is much smaller than $R\eta\omega$, which is itself smaller than R: η ranges from 20 % for PMT to 80 % for APD as summarized in Table 4.3; ω ranges from 20 % to 60 % in reflector-coated pixelated crystals depending on geometry [268], and varies between 10 % to 50 % in absorber-coated continuous crystals depending on position, although it can be increased by using retro-reflectors [269].

Photodetectors provide an additional resolution limit to timing through their intrinsic transit time spread, that is independent for each detected photon. For instance, in a PMT the transit time spread is dominated by the variation in the distance traveled by the photoelectron between the photocathode and the first dynode, which depends on position of emission, initial velocity, and voltage between cathode and first dynode ([266], pp. 57–59). In general, the signal at the output of a linear photodetector may be approximated as the superposition of the signals that are generated independently for each detected photon, i.e.

$$i\left(t\right) = \sum_j i_{\mathrm{SPR}}^{(j)}\left(t - t_j\right) \tag{4.35}$$

where the sum ranges over all photons j detected at times t_j [270]. Here, $i_{\mathrm{SPR}}\left(t\right)$ is the *Single Photon Response* (SPR) waveform, and it has different parameters for each instance $i_{\mathrm{SPR}}^{(j)}$ due to statistical variations like transit time spread. In any case, if the SPRs of the N first photons overlap, the fact that the sensor and photodetector are described by Poisson statistics imply that $\sigma_{t\ \mathrm{pd}}$ scales as the inverse of the square root of the initial photoelectron rate, i.e. the photodetector resolution is proportional to $1/\sqrt{R}$ [262, 271].

Typical numerical values for all these timing uncertainties are now given. The contribution due to crystal geometry and light propagation time is usually in the range of 300 ps FWHM [262, 266] for pixelated detectors. The transit time spread depends on the type of photodetector and is around 400 ps FWHM. The variation due to photon emission statistics depends on the detection level; typical values for a continuous LSO crystal coupled to a PMT yield an uncertainty in the order of $90 \cdot \sqrt{N}$ ps FWHM. It is obviously advisable to try to detect one of the first photons in order to reduce the latter contribution, however this may be unfeasible depending on the detection electronics. In any case, the dominant factor in (4.30) is usually the contribution of the timing electronics, while the sensor is assumed to impose fundamental limitations [262].

Analog timing electronics

Due to the aforementioned limitations to resolution, the exact time of scintillation can never be determined. Nevertheless, a definite criterion is needed that univocally assigns a time reference to each detected subevent from the available information, i.e. the output signals. This process for time assignment is generically referred to as *time pick-off*, or simply *timestamping* when the assigned time values are digitized.

Time pick-off methods contribute their own term $\sigma_{t\ \mathrm{el}}$ to time resolution; their uncertainty can be typically divided into two components depending on their source: *time jitter* corresponds to variations in time marks assigned to detector pulses with similar characteristics due to electronic noise, and *time walk* includes effects derived from variations in the signal parameters such as amplitude, rise time or shape [272].

The most simple and oldest time pick-off method is usually called *leading edge discrimination* (LED) and consists in the analog comparison of the output voltage signal $v(t)$ with a fixed threshold value V_{th}. The estimated time mark $t^{\star}$ is then the position where the signal crosses the threshold, i.e. $v(t^{\star}) = V_{\mathrm{th}}$. It would be desirable to keep V_{th} as low as possible in order to have its crossing correspond approximately to the detection of the N-th photon with N as small as possible and thus minimize (4.34). However, there is a limit to how low V_{th} may be due to the presence of electronic noise in the signal that may exceed the threshold even in the absence of pulse.

Time jitter in a leading edge discriminator is caused by variations in the threshold-crossing point due to instantaneous noise and is easy to analyze. If constant pulse shape and amplitude are considered, i.e.

$$v(t) = v_0(t) + n(t) \tag{4.36}$$

with a reference pulse waveform $v_0(t)$ and additive noise $n(t)$ with variance σ_v^2, then time jitter can be estimated as [272]

$$\sigma_{t\ \mathrm{j}} \approx \frac{\sigma_v}{\left.\frac{\partial v_0}{\partial t}\right|_{t=t^{\star}}} \tag{4.37}$$

by linearizing around $t^{\star}$. Fast rise times with abrupt signal slopes thus help reduce the effect of electronic noise on timing uncertainty. Time walk is however a serious problem in leading edge discriminators, as shown in Fig. 4.21. Pulses $v_i(t)$ starting at the same time t_0 lead to different assigned time marks t_i due to variations in pulse amplitude (t_1 vs t_2) or rise time (t_1 vs t_3). Leading edge is therefore an appropriate method only for cases with small variations in pulse shape, e.g. with tight energy windows that reduce the variations of pulse amplitude.

An improved and very widespread time pick-off method is the *Constant Fraction Discriminator* (CFD), a modification of leading edge that intends to cancel the

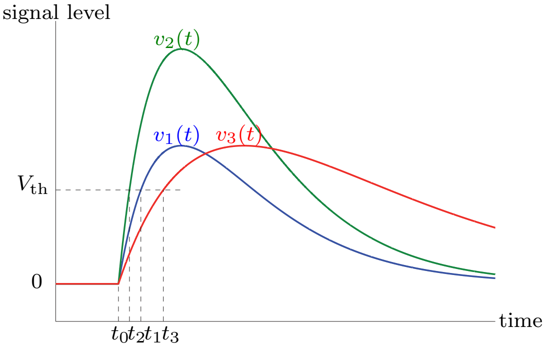


Figure 4.21: Illustration of leading edge discrimination and the walk error introduced by varying amplitudes and rise times.

time walk due to amplitude variation [273]. The main idea is that, when considering pulses with the same shape but different amplitudes, the correct threshold for each one is a constant fraction $\alpha \in [0,1]$ of said pulse's amplitude in order to obtain the same time marks, i.e. if $v(t)$ has amplitude 1 and $v(t^\star) = V_{\text{th}}$ for threshold $V_{\text{th}} = \alpha$, then $A \cdot v(t)$ has amplitude A and $A \cdot v(t^\star) = A \cdot V_{\text{th}}$ so threshold $\alpha \cdot A$ yields the same time estimation $t^\star$.

Several realizations of this idea are possible, and all of them involve some delay since the amplitude of the detected pulse cannot be known before its peak value is actually reached. The most typical implementation requires an analog delay line with delay time t_δ and an adder in order to obtain the analog signal

$$v_{\text{cfd}}(t) = v(t - t_\delta) - \alpha \cdot v(t), \tag{4.38}$$

and the time mark $t^\star$ is given by the bipolar signal $v_{\text{cfd}}(t)$ first crossing zero, i.e. $v_{\text{cfd}}(t^\star) = 0$. Its principle may be understood by noticing that $v(t^\star - t_\delta) = \alpha \cdot v(t^\star)$; hence, for a given fraction α, one should choose t_δ equal to the delay from the desired threshold crossing to the peak of $v(t)$ which, using a linear approximation of the rising signal, may be estimated as

$$t_\delta \approx (1 - \alpha)\, t_p \tag{4.39}$$

where t_p is the peaking time of $v(t)$ i.e. the time where it reaches its maximum [272]. In particular, if the shape of $v(t)$ has a slow fall, it suffices to choose t_δ longer than (4.39) so that the maximum is reached. Figure 4.22 shows how the method applied to two pulses $v_1(t)$ and $v_2(t)$ with the same shape but different amplitude yields the same time mark t_{cfd}.

A CFD mitigates the walk error due to pulse amplitude but is still vulnerable to variations in signal rise time due to (4.39). One possible solution is the so-called

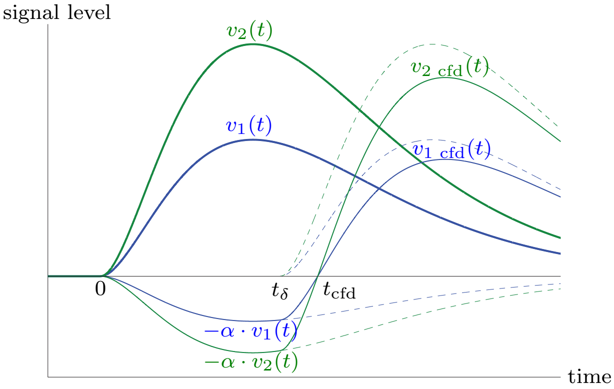


Figure 4.22: Illustration of the Constant Fraction Discrimination method. Two waveforms v_1 and v_2 with the same shape but different amplitude result in the same time estimation t_{cfd}.

Amplitude and Rise-time Compensated (ARC) discrimination [274], that looks for the zero crossing in a signal (4.38) like CFD but using a smaller delay than (4.39), so that the detection happens before the peak is reached and a constant fraction of a voltage value below the maximum is achieved. It can be proved that ARC removes the walk error due to varying rise times in pulses with exactly linear rising waveforms, and hence is expected to reduce the error in a realistic case.

In any case, both CFD and ARC have higher time jitter than the leading edge discriminator, due to the generated bipolar signal (4.38) having higher noise than $v\,(t)$. In fact, assuming uncorrelated noise, $v_{\mathrm{cfd}}\,(t)$ has noise variance $\sigma_v^2\left(1+\alpha^2\right)$ and time jitter (4.37) is multiplied by a factor $\sqrt{1+\alpha^2}$; for ARC, an additional factor $1/\left(1-\alpha\right)$ is applied because the signal slope at 0 is reduced ([193], pp. 91–94). Selection of the optimal discrimination method thus depends on a trade-off between both sources of time uncertainty.

Analog timing methods, particularly CFD and ARC, are widely regarded to yield the best coincidence resolution, with values below 500 ps FWHM [275], and have the advantage of being intrinsically real-time, unlike their digital counterparts.

Digital timing electronics

The generic trend in HEP DAQ systems of moving the digitization step as close as possible to the detectors was already discussed and justified in §2.2. The same reasoning may be applied to the case of PET data acquisition, and in particular to time pick-off electronics. Analog timing methods like CFD have demonstrated excellent performance, but their parameters (α, t_δ) are hardwired and thus difficult to calibrate. The alternative is early digitization of timing signals followed by intensive digital signal processing, with the intention of obtaining a precise timestamp for detected subevents. A crucial distinction between digital methods is whether the computation of timestamps and determination of coincidences is performed online (typically in FPGAs) or offline in a PC. Offline methods are usually more precise but require the capture of a much larger data volume.

Many proposals for digital timestamping involve direct translations of widespread analog methods to the digital domain. For instance, digital CFD or ARC (both shorted as DCFD) may be implemented from the digitized timing signal $t\,[n]$ by generating the bipolar signal

$$b\,[n] = t\,[n] - A \cdot t\,[n-k] \tag{4.40}$$

for a discrete time delay k; here, $A > 1$ is a parameter that acts as $1/\alpha$. The zero crossing point of $b\,[n]$ is then obtained by interpolation, either linear [276] or cubic [277]. Additional signal processing steps are possible in order to refine the time estimation, like signal filtering, upsampling [278], and position-dependent correction using LUTs [279]. Moreover, all of the parameters involved in these steps may be reconfigured. Alternative, purely digital approaches are also possible like estimation using neural networks [280], matched filters [281], adaptive filtering [282] or pulse shape fitting [283].

Digital time pick-off is subject to additional uncertainty due to the error introduced by the digitization process. Conversion of the timing signal to the digital domain carries the addition of quantization noise with standard deviation

$$\sigma_\mathrm{q} = \frac{V_\mathrm{LSB}}{\sqrt{12}} = \frac{V_\mathrm{FS}}{2^n\sqrt{12}} \tag{4.41}$$

where V_LSB is the quantization step size, V_FS is the full scale range and n is the number of bits per sample ([284], pp. 406-408). If n is high enough, this error is independent of all other voltage noise sources and thus can be converted to time uncertainty using (4.37) and added quadratically. Further errors are introduced by jitter in the sampling clock, that implies an uncertainty in the exact time instant that any sample belongs to.

Additionally, discretization of the signal implies loss of information if the sampling frequency is below the Nyquist rate. If the rule of thumb ([90], pp. 329–331)

$$BW \cdot t_r \approx 0.35 \tag{4.42}$$

for the relationship between a signal's 10 % to 90 % rise time t_r and its bandwidth BW is accepted as valid, then the Nyquist criterion is $t_r > 0.7\,T_\mathrm{s}$ for the sampling period T_s. While this is theoretically valid, real time timestamping algorithms cannot extract all the information contained in the discrete signal in time and thus a finer sampling is needed. For instance, the criterion

$$t_r \geq 3\,T_\mathrm{s} \tag{4.43}$$

i.e. capturing at least three samples of the rising edge, has been suggested in [285] and ([193], pp. 138–142) as an estimation for the minimum sampling requirements for acceptable resolution.

The performance of digital timing methods is typically lower than that of analog timing electronics, and is highly dependent of whether the time extraction algorithms are being applied online (in real time) or offline (after data acquisition). In the first case, coincidence resolutions around 2 ns FWHM have been reported for moderate sampling frequencies below 100 MHz [278]. Offline processing has been proven capable of pushing the resolution to 1.2 ns FWHM for slightly higher sampling frequencies [282]. Extreme variations exist using GHz-rate sampling with flash ADCs or switched capacitor arrays that achieve resolutions in the order of 10 ps [286] but these implementations seem highly impractical for a PET scanner.

Synchronization between detectors

There is one additional contribution to coincidence resolution in PET systems that is central to this dissertation: the effect of synchronization between detectors. The influence of synchronization accuracy has been traditionally disregarded and is only acknowledged in very recent work such as [287].

The issue at the core of this effect is the fact that the time pick-off processes applied to subevents from separate detector modules take place at different physical locations, but their results need to be transferred to a common location in order to obtain time difference and resolve coincidence relations. Any mismatch in the time references for both time extraction circuits is reflected in the estimated time difference. For instance, in a scanner with purely analog timing electronics, the pulses generated by CFDs need to be transported to a common component that detects coincidences using transmission lines with propagation delay t_p; fixed mismatches in t_p for different detectors are calibrable, but the variation σ_{t_p} due to drift or temperature change appears directly as a component $\sigma_{t\ \mathrm{sync}}$ in time resolution. Similarly, in scanners with digital timestamping, variations in the time reference at the sampling points (e.g. due to jitter and variance in the propagation delays of sampling clock signals) contribute to timing uncertainty.

The main reason for this uncertainty in a typical detector is the variation in the propagation delay for timing signals in the system and is usually far below 500 ps

[7]. Until recently, this was small enough compared to the contribution of timing electronics so it could be neglected. However, the improvement in coincidence resolution in recent detectors and the renewed popularity of ToF PET imply that this effect cannot be neglected anymore and care must be taken in order to keep it within acceptable levels. This issue, as well as the methods to act on it, have already been developed extensively in chapter §3 for a generic setting and will be applied to PET in chapters §5 and §6.

4.3.4 Time-of-Flight PET

It has been assumed so far that the end result of an event acquisition is a line of response with two detected endpoints, and the only information provided by such an event is the knowledge that a β^+ decay took place somewhere along that LOR. However, it is possible to extract further information from the particular value of the time difference t_d. Consider again Fig. 4.20 and (4.24), that expresses the difference between distances d_1 and d_2 in terms of t_d. Since the interaction positions are known, so is the total distance $d = d_1 + d_2$ between them. From these two values one obtains directly

$$d_1 = \frac{d}{2} + \frac{c}{2} t_d,\; d_2 = \frac{d}{2} - \frac{c}{2} t_d \tag{4.44}$$

that yield the exact decay position with a resolution[17]

$$\sigma_d = \frac{c}{2} \sigma_c. \tag{4.45}$$

This observation defines the PET modality known as *Time of Flight* (ToF) PET, in which the data given to the image reconstruction process by each event include not just a LOR but also an estimation of the disintegration location within that LOR.

Obviously, coincidence resolution σ_c needs to be small enough for this to be of any use, and in particular $\sigma_d \ll D$ is required. For instance, a ToF resolution of 1 cm requires a coincidence resolution below 70 ps which to this day remains unattainable in a complete PET scanner. A coincidence resolution of 500 ps FWHM, corresponding to 7.5 cm, is estimated as the bare minimum for a reasonable ToF detector [288].

The potential for time-of-flight imaging was already recognized in the early days of PET [289], but its principles were impossible to apply initially due to technological constraints. Several early ToF PET designs were attempted in the 1980s that fulfilled this requirement on resolution [290–292]; however, achieving these time resolutions required the choice of ultra-fast scintillators like BaF_2 that ultimately

[17] The effect of position resolution in the determination of d is being disregarded here.

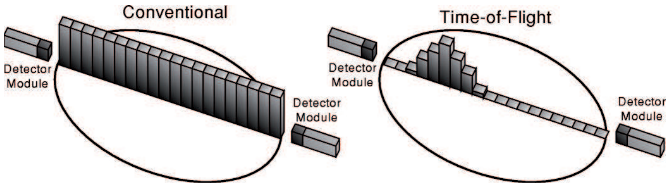


Figure 4.23: Probability density function for the location of an event located along a LOR of length D for conventional PET (left) and ToF PET (right). Reproduced from [261], © 2003 IEEE.

proved hopelessly lacking in other parameters of interest for PET imaging (particularly, light yield and density), and the idea was shelved [293]. The development of new scintillators like LSO in the 1990s opened the way for ToF PET again [261] and a number of successful commercial ToF PET scanners have been developed in the last decade as a result [214, 294, 295].

The benefits of ToF PET may only be seen at the image reconstruction level, since the detectors and acquisition electronics are identical to the case of conventional (i.e. non-ToF) PET. They can be quantified as a *ToF gain* G, defined as the ratio between the noise variances at a given point in the images obtained using ToF and non-ToF reconstruction; equivalently, it is the square root of the ratio between the signal-to-noise ratios in the images [296]. It can be estimated by considering Fig. 4.23, where the probability densities for the location of a given event are sketched in the cases of ToF and non-ToF PET. By constraining the position to a section of the LOR, it can be assured that any detected event only influences the resulting image within a neighborhood of its actual position. This argument was applied in [297] to estimate the ToF gain as[18]

$$G \approx \frac{D}{\sigma_{d\ \mathrm{FWHM}}} = \frac{2D}{c\ \sigma_{c\ \mathrm{FWHM}}}. \tag{4.46}$$

It may be proved that this variance reduction also applies to random and scattered events, leading to an improved sensitivity of the scanner [261]. Hence, ToF gain may be exploited for better image quality, but also for shorter acquisition times or lower radiotracer doses [296].

[18] This estimate corresponds to an idealized case; more exact models imply that the actual ToF gain is slightly reduced by a factor between 1.4 and 1.6 [296].

4.4 Image reconstruction

It has been detailed in §4.3 how the raw signals measured at the detectors in a PET scanner are converted into events containing useful data, namely a straight line of response that contains one radiotracer molecule, and additionally an estimation of the location of said molecule in the case of ToF PET. The mathematical principles behind the conversion of this list of events into a reconstructed image are now presented.

4.4.1 Analytical reconstruction methods

Let $f(\mathbf{x})$ be the function describing the concentration of the radiotracer in the patient at spatial position $\mathbf{x}$. The PET technique consists precisely in the estimation of this unknown distribution, and the final image is just a graphical representation of $f(\mathbf{x})$. Traditionally, an *image reconstruction method* in conventional PET is an algorithm that provides an estimation of the spatial distribution f in terms of the line integrals $\int_r f$ of f along straight lines r belonging to a set $\mathcal{R}$ of valid LORs.

The role of line integrals

In order to understand the connection between the line integrals $\int_r f$ and acquired PET data, let us fix a straight line r between two detectors in the scanner. Disregarding angular deviation and scatter, each positron annihilation generates two diametrically opposed photons in a random direction, and all directions are equally likely to be taken. Hence, the probability[19] that any given decay process generates photons along LOR r is

$$P(r) = \int_r p(\cdot, r) \propto \int_r p \propto \int_r f \tag{4.47}$$

where $p(\mathbf{x}, r)$ is the probability density for a given photon pair to be generated at position $\mathbf{x}$ along LOR r. The first proportionality in (4.47) follows from the observation that all directions are equally likely, and the second one represents the fact that the decay activity is proportional to the radiotracer concentration at any given point.

Let $N(r)$ be the number of events generated along LOR r for a whole acquisition session. The expected value of $N(r)$ is then proportional to the probability of this LOR, and it follows that

$$E[N(r)] \propto \int_r f. \tag{4.48}$$

The conclusion is that the line integral $\int_r f$ can be estimated by the total number $N(r)$ of coincidence events generated along r.

[19] Rigorously, the probability density.

Notice that the information available to the detector is not the amount $N(r)$ of generated events along r but rather the amount $D(r)$ of *detected* events. A source of error in the estimation of $N(r)$ is introduced here by the fact that emitted photons may be attenuated i.e. interact with the surrounding tissue before arriving at the detector. It was shown in §4.2 that the probability of both photons reaching the scanner is

$$P_d(r) = \exp\left(-\int_r \mu\right) \tag{4.49}$$

where $\mu(\mathbf{x})$ is the attenuation coefficient at location $\mathbf{x}$. The probability $P_d(r)$ thus only depends on the LOR but not on the particular position along the LOR. Independence between the emission and attenuation processes yields

$$E[D(r)] = P_d(r) \cdot E[N(r)] \tag{4.50}$$

i.e. $N(r)$, and therefore the line integrals, may be estimated without attenuation error as long as the probabilities of detection $P_d(r)$ are known.

Hence, it turns out that there are actually two unknown scalar fields to be determined: the radiotracer concentration $f(\mathbf{x})$ and the attenuation coefficient $\mu(\mathbf{x})$, since the effect of attenuation cannot be ignored [298]. The most typical method of obtaining the attenuation is performing a transmission scan, whereby an external positron source, typically a collimated rod source that emits photons in a single direction, is rotated around the patient; for each position, the expected amount of detected events originated at the rod source is

$$E[D_t(r)] = P_d(r) \cdot E[N_t(r)] = P_d(r) \cdot \text{constant}. \tag{4.51}$$

Several techniques exist in order to separate transmission events from emission (radiotracer) events ([188], pp. 66–68). An estimation of $P_d(r)$ is thus provided by $D_t(r)$, and (4.50) can be used to obtain an estimation of $N(r)$.

As a summary of the above considerations, the line integral $\int_r f$ can be estimated from detected data as

$$\int_r f \approx \frac{D(r)}{D_t(r)} \tag{4.52}$$

disregarding a proportionality constant. Moreover, increasing the amount of events enhances the accuracy of these estimations as statistical noise is reduced.

Although the theoretical framework for these methods is expressed in terms of an infinite, continuous amount of LORs, the set $\mathcal{R}$ is discrete and finite in practice. Assuming a typical ring topology, if the detector surface is divided into pixels with spacing Δx and a LOR is defined for each pair of different pixels, the resulting set $\mathcal{R}$ consists approximately of collections of parallel lines with uniform spacing Δx for uniformly spaced angular directions.

Differences for ToF PET

In the case of ToF PET, the data provided to the reconstruction algorithm is a set $\mathcal{E}$ of events, each of them of the form $(r, \mathbf{x})$ where $r \in \mathcal{R}$ is the LOR where said event was detected, and $\mathbf{x} \in r$ is the estimation of the annihilation position along r. After discretizing for r and $\mathbf{x}$, the remaining information is not merely $D(r)$ but rather $D(r, \mathbf{x})$, i.e. the number of events detected in location $\mathbf{x}$ along LOR r. Independence between emission and attenuation yields

$$E\left[D\left(r, \mathbf{x}\right)\right] = P_d\left(r\right) \cdot E\left[N\left(r, \mathbf{x}\right)\right] \propto D_t\left(r\right) \cdot E\left[N\left(r, \mathbf{x}\right)\right] \tag{4.53}$$

as in conventional PET, where $N(r, \mathbf{x})$ is the number of events detected as taking place at $\mathbf{x}$ and emitting their annihilation photons along LOR r.

The corresponding line integral is different in this case. Let $\mathbf{u}$ denote the *actual* location where the annihilations occur, as opposed to $\mathbf{x}$ which is an estimation with resolution σ_d given by (4.45). It is known that $\mathbf{u} \in r$. The probability that an event $(r, \mathbf{x})$ is generated is then

$$p\left(r, \mathbf{x}\right) = \int_r p\left(r, \mathbf{x}, \mathbf{u}\right) d\mathbf{u} = \int_r p\left(\mathbf{x} | r, \mathbf{u}\right) p\left(r, \mathbf{u}\right) d\mathbf{u} \tag{4.54}$$

abusing the notation in order to clarify that the integral is over $\mathbf{u} \in r$. The probability $p(\mathbf{x}|r, \mathbf{u})$ that an event generated at $\mathbf{u}$ along r is actually detected at $\mathbf{x}$ is determined by the ToF position estimation resolution; a normal distribution is usually assumed, so that

$$p\left(\mathbf{x} | r, \mathbf{u}\right) = g\left(\left\|\mathbf{x} - \mathbf{u}\right\|\right) = \frac{1}{\sqrt{2\pi}\sigma_d} \exp\left(-\frac{\left\|\mathbf{x} - \mathbf{u}\right\|^2}{2\sigma_d^2}\right) \tag{4.55}$$

and thus

$$p\left(r, \mathbf{x}\right) = \int_r p\left(r, \mathbf{u}\right) g\left(\left\|\mathbf{x} - \mathbf{u}\right\|\right) d\mathbf{u} \propto \int_r f\left(\mathbf{u}\right) g\left(\left\|\mathbf{x} - \mathbf{u}\right\|\right) d\mathbf{u} \tag{4.56}$$

using the same arguments as in (4.47). It follows that the estimations of line integrals provided by ToF PET are actually

$$\int_r f \cdot g_{\mathbf{x}} = \int_r f\left(\mathbf{u}\right) g\left(\left\|\mathbf{x} - \mathbf{u}\right\|\right) d\mathbf{u} \approx \frac{D\left(r, \mathbf{x}\right)}{D_t\left(r\right)} \tag{4.57}$$

i.e. the original line integrals modified with a kernel centered at the estimated position that accounts for the ToF resolution. Of course, by integrating or summing across $\mathbf{x} \in r$, the conventional line integrals $\int_r f$ are recovered.

2D and 3D PET

Two different data acquisition modalities can be distinguished for PET: two- and three-dimensional. In 2D PET, data for each detector ring are considered separately and LORs whose endpoints belong to different rings are discarded. Purely two-dimensional reconstruction methods are then applied for each detector ring, yielding planar sections of f that, when combined, describe the whole distribution within the active area.

In 3D PET, LORs between different rings are also taken into consideration. Hence $\mathcal{R}$ includes collections of lines contained in planes that intersect the ring plane under angles θ that are approximately uniformly spaced. The sensitivity for this modality is obviously much higher than in the case of 2D PET, suggesting that reconstruction methods should yield much better results. However, the influence of scatter is also greatly increased in this case, and it becomes necessary to apply scatter correction methods to the collected data before image reconstruction [299].

It should be noted that, in order to perform attenuation correction in 3D PET, it is not necessary to measure $P_d\left(r\right)$ explicitly for each 3D direction, but rather for the 2D directions along the ring plane for each ring in the scanner. Indeed, due to (4.49), the measurement of $P_d\left(r\right)$ is equivalent to the estimation of the line integrals $\int_r \mu$. Hence, a 2D PET reconstruction algorithm can be applied first to $\{P_d\left(r\right)\}$ for r in each plane in order to obtain an estimation of the planar section μ in that ring plane. Once $\mu\left(\mathbf{x}\right)$ is known for every location $\mathbf{x}$, the value of $P_d\left(r\right)$ for oblique LORs can be computed by integration.

4.4.2 2D analytical reconstruction

2D PET image reconstruction converts the line integrals of the scalar field f over a discrete set $\mathcal{R}$ of lines contained in planes corresponding to detector rings into an estimation of f on each of those planes. For the discussion of 2D methods, the restriction to one of these planes will be considered; in particular, a two-dimensional field $f : \mathbb{R}^2 \to \mathbb{R}$ will be assumed. The reconstruction procedures will be presented in terms of conventional PET, and the extension to ToF PET will be described afterward.

Projections

Let R_φ denote the operator for rotation in the plane by angle φ, that can be expressed in matrix form as

$$\begin{pmatrix} t \\ u \end{pmatrix} = R_\varphi \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \tag{4.58}$$

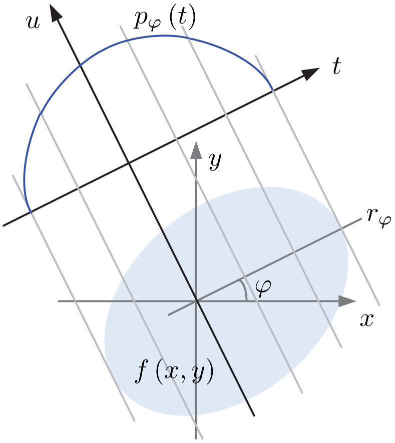


Figure 4.24: Projection p_φ of a distribution f under angle φ. Each value of $p_\varphi(t)$ is the integral of f over the oblique line passing through it.

Notice that the change of variables given by the rotation of the coordinate axes by angle φ is given by $R_{-\varphi}$, i.e. it is equivalent to a rotation of the plane in the opposite direction. Also notice that any straight line r in the plane may be described as the set $\{R_\varphi(t,u) : u \in \mathbb{R}\}$ for some t and an adequate rotation angle φ that turns the y axis parallel to r.

A *projection* of f is defined as the collection of line integrals of f over all lines parallel to a given direction, and can be described as follows: for a given angle φ, let r_φ be the line going through the origin with angle φ, that may be parameterized as

$$r_\varphi : t \mapsto (t\cos\varphi, t\sin\varphi), \; t \in \mathbb{R}. \tag{4.59}$$

For each point $\mathbf{x} = r_\varphi(t)$ in r_φ, the projection onto $\mathbf{x}$ is computed as the line integral of f over the line through $\mathbf{x}$ and orthogonal to r_φ. This defines a projection function $p_\varphi : \mathbb{R} \to \mathbb{R}$ whose analytical expression is

$$\begin{aligned} p_\varphi(t) &= \int_{-\infty}^{\infty} f(R_\varphi(t,u))\,du \\ &= \int_{-\infty}^{\infty} f(t\cos\varphi - u\sin\varphi, t\sin\varphi + u\cos\varphi)\,du \end{aligned} \tag{4.60}$$

and is equal to the line integral $\int_r f$ over the line r determined by φ and t as discussed in the last paragraph. Figure 4.24 shows an example of the computation of these projections.

The function $p(\varphi, t)$, when considered as a two-dimensional function $p : \mathbb{R}^2 \to \mathbb{R}$, is called the *Radon transform* of f. As explained in §4.4.1, the measurements taken by a 2D PET scanner yield estimations of the line integrals of f over straight lines, and hence of the Radon transform $p(\varphi, t)$, for uniformly spaced φ and t. The graphical depiction of p is usually called a *sinogramsinogram* due to the fact that the transform of a point distribution has a sine wave graph, and therefore the sinogram of a distribution concentrated on a discrete set of small zones of activity is composed of several sine plots with different amplitudes and phases [300].

Direct reconstruction

The problem of 2D image reconstruction in PET can be restated as the problem of recovering the original distribution f from its Radon transform p, i.e. finding a way to compute the inverse Radon transform. The theoretical solution was given by Radon in the early 20th century, long before it had any application in the field of medical diagnose [301]. In order to derive the inversion formula, the Fourier transforms of the different projections have to be considered, given by

$$P_\varphi(\tau) = \mathcal{F}_{1\mathrm{D}}\left[p_\varphi(t)\right] = \int_{-\infty}^{\infty} p(\varphi, t)\, e^{-2\pi i t\tau} dt \tag{4.61}$$

as well as the two-dimensional transform of f

$$F(\xi, \eta) = \mathcal{F}_{2\mathrm{D}}\left[f(x, y)\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\, e^{-2\pi i (x\xi + y\eta)} dx\, dy \tag{4.62}$$

where $\mathcal{F}_{1\mathrm{D}}$ and $\mathcal{F}_{2\mathrm{D}}$ stand for the one and two-dimensional Fourier transform operators, respectively.

By substituting the formula (4.60) into (4.61) and considering that the variable change corresponding to a rotation has a Jacobian determinant equal to 1, the expression

$$\begin{aligned} P_\varphi(\tau) &= \int_{-\infty}^{\infty} p(\varphi, t)\, e^{-2\pi i t\tau} dt \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(R_\varphi(t, u))\, e^{-2\pi i t\tau} du\, dt \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\, e^{-2\pi i (x\cos\varphi + y\sin\varphi)} dx\, dy \\ &= F(\tau\cos\varphi, \tau\sin\varphi) \end{aligned} \tag{4.63}$$

is obtained. This identity is referred to as the *Central Section Theorem* and can be rephrased in the following terms: the Fourier transform of a projection of f is just the section of the 2D transform of f along the line going through the origin with the same angle φ corresponding to the projection. It provides a method, called *direct reconstruction*, for the recovery of f from p: if all the projections p_φ

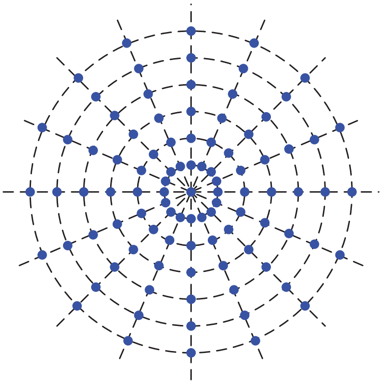


Figure 4.25: Lattice of points where the sinogram values are measured, with a non-uniform spatial distribution.

are known, then their Fourier transforms P_φ can be computed, so the value of F at each point is given by (4.63), and f can be finally obtained as the inverse 2D Fourier transform of F.

The analytical recovery of f following the aforementioned algorithm presents some practical difficulties due to the non-uniform discretization of the projections measured by the PET scanner ([302], pp. 119–128). Since $p(\varphi, t)$ is known for a uniform distribution of φ and t, the computation of P_φ yields the values of F on a two-dimensional lattice $\{(\tau\cos\varphi, \tau\sin\varphi)\}$ for uniformly distributed τ and φ, as depicted in figure 4.25. However, this lattice is not uniformly distributed along orthogonal coordinates x and y, but rather is denser close to the origin and becomes more sparse as the distance increases. Therefore, the standard algorithms for inverse 2D Fourier transform cannot be used. One possible solution is to apply an interpolation method in order to convert the polar lattice into a more regular lattice (rectangular or otherwise) before transforming [303]. A different, more extended alternative is described next.

Filtered backprojection

In order to obtain an expression for f in terms of p, a change of variables is applied to the formula describing the inverse 2D Fourier transform of F, using the semipolar coordinates given by

$$\left\{ \begin{aligned} \tau &= \xi\cos\varphi + \eta\sin\varphi \\ \tan\varphi &= \eta/\xi \end{aligned} \right. \tag{4.64}$$

where $t \in \mathbb{R}$ and φ varies between 0 and π. This change of variables has Jacobian determinant τ, therefore

$$\begin{aligned} f\left(x,y\right) &= \mathcal{F}_{\mathrm{2D}}^{-1}\left[F\left(\xi,\eta\right)\right] \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F\left(\xi,\eta\right) e^{2\pi i\left(x\xi+y\eta\right)} d\xi\, d\eta \\ &= \int_{0}^{\pi}\int_{-\infty}^{\infty} F\left(\tau\cos\varphi,\tau\sin\varphi\right) e^{2\pi i\left(x\tau\cos\varphi+y\tau\sin\varphi\right)} \left|\tau\right| d\tau\, d\varphi \\ &= \int_{0}^{\pi}\int_{-\infty}^{\infty} P_{\varphi}\left(\tau\right)\left|\tau\right| e^{2\pi i\tau\left(x\cos\varphi+y\sin\varphi\right)} d\tau\, d\varphi \end{aligned} \tag{4.65}$$

where (4.63) was applied in the last step. The expression within the integral has the form of an inverse Fourier transform, so one finally obtains[20]

$$f\left(x,y\right) = \int_{0}^{\pi} \mathcal{F}_{\mathrm{1D}}^{-1}\left[P_{\varphi}\left(\tau\right)\left|\tau\right|\right]\Big|_{t=x\cos\varphi+y\sin\varphi} d\varphi. \tag{4.66}$$

The term

$$b_{\varphi}\left(t\right) = \mathcal{F}_{\mathrm{1D}}^{-1}\left[P_{\varphi}\left(\tau\right)\left|\tau\right|\right] \tag{4.67}$$

within the integral in (4.66) is called a *filtered projection*, and is the result of filtering the projection p_{φ} with a ramp filter with frequency response $H\left(\tau\right) = \left|\tau\right|$. The *Filtered Backprojection* (FBP) algorithm consists on obtaining the distribution f directly using (4.66) [304].

In a real PET scanner, the projections of f are not available for all angles φ but rather for a finite amount K of them, separated with a constant step $\Delta\varphi$. Hence, f is estimated using the following discretized version of the integral (4.66):

$$f\left(x,y\right) \approx \sum_{k=0}^{K-1} \Delta\varphi \cdot b_{k\,\Delta\varphi}\left(x\cos\left(k\,\Delta\varphi\right) + y\sin\left(k\,\Delta\varphi\right)\right). \tag{4.68}$$

Practical implementations of FBP compute the filtered projections $b_{k\,\Delta\varphi}$ one by one and accumulate the values of f instead of computing all filtered projections at once, thus saving a considerable amount of storage resources.

The FBP algorithm as described above still presents some drawbacks due to the way that the projections are acquired:

- On one hand, the projections that are used as a starting point for reconstruction are not available in full form but sampled with spacing Δt. Hence, by the Nyquist sampling theorem, frequencies higher than $\tau_{\max} = 1/2\Delta t$ should

[20] In these computations, a multiplicative factor of 2π is being neglected for simplicity, as it has no impact on image reconstruction.

not be considered in their Fourier transform in order to avoid aliasing. The bandwidth of distributions in tomography is usually high, hence this axial sampling imposes a bound on the resolution which the reconstructed image can be obtained with.

- On the other hand, since the projections are measured using statistical methods, they are subject to statistical noise that is dominant at high frequencies. Moreover, projections are modified with a ramp filter, whose frequency response is increasing and unbounded and thus has an amplifying effect on the statistical noise.

In order to mitigate these effects, the projections are *windowed* i.e. filtered with a low-pass filter $W(\tau)$ prior to reconstruction [304]. Usually, a Hamming window is used

$$W(\tau) = \begin{cases} \alpha + (1-\alpha)\cos(\pi\tau/\tau_{\max}) & , \ |\tau| < \tau_{\max} \\ 0 & , \ |\tau| \geq \tau_{\max} \end{cases} \tag{4.69}$$

that filters all frequencies above $\tau_{\max}$. This has a smoothing effect on the image, reducing the statistical and sampling noise. The filtered projections are computed as

$$b_\varphi(t) = \mathcal{F}_{1\text{D}}^{-1}\left[P_\varphi(\tau)\,W(\tau)\,|\tau|\right] \tag{4.70}$$

and the reconstructed distribution as $f(x,y) = \int_0^\pi b_\varphi(x\cos\varphi + y\sin\varphi)\,d\varphi$. The performance of this method is largely controlled by the choice of $\tau_{\max}$ [302].

FBP with time-of-flight data

The operations involved in the FBP algorithm as described are all linear, and hence the algorithm itself may be regarded as a linear system, and the result corresponding to the set $\mathcal{E}$ of acquired events is the sum of the responses corresponding to individual events in $\mathcal{E}$, as hinted by (4.68).

In the case of conventional PET, events are described by a LOR r_i that may be represented as a pair (φ_i, t_i), and the response to such an event can be expressed in terms of a 2D impulse response function h corresponding to event $(0,0)$ which has the y axis as LOR. The response h must be invariant in y, i.e. $h(x,y) = \tilde{h}(x)$ for a one-dimensional filter $\tilde{h}$. The reconstructed image is then the sum

$$\begin{aligned} f(x,y) &= \sum_{(\varphi_i,t_i)\in\mathcal{E}} \delta\left(R_{-\varphi_i}(x,y) - (t_i,0)\right) * h(x,y) \\ &= \sum_{(\varphi_i,t_i)\in\mathcal{E}} h\left(R_{-\varphi_i}(x,y) - (t_i,0)\right) \\ &= \sum_{(\varphi_i,t_i)\in\mathcal{E}} \tilde{h}\left(\pi_1 \circ R_{-\varphi_i}(x,y) - t_i\right) \end{aligned} \tag{4.71}$$

where δ is the Dirac delta and π_1 is the projection onto the first coordinate. The impulse response $\tilde{h}$ is essentially the inverse transform of the filter $W(\tau)|\tau|$ used to compute the filtered projections.

The extension to ToF PET was first described in [305] and was shown to be equivalent to an additional filter that accounts for the position information within the LOR.[21] In the spatial domain, this amounts to convolving the response h with the inverse of the Gaussian kernel as given by (4.55); the resulting $h(x, y)$ then includes variation across y, such that values around 0, i.e. the ToF-estimated position, are higher. An event in $\mathcal{E}$ is now represented as (φ_i, t_i, u_i) where the estimated position is at ordinate u_i in LOR r_i, and the reconstructed image is obtained as the sum

$$f(x, y) = \sum_{(\varphi_i, t_i, u_i) \in \mathcal{E}} h\left(R_{-\varphi_i}(x, y) - (t_i, u_i)\right) \tag{4.72}$$

which reduces to (4.71) if h is invariant in y i.e. the ToF estimation u_i is not used.

4.4.3 3D analytical reconstruction

In the case of 3D PET, a scalar field $f : \mathbb{R}^3 \to \mathbb{R}$ is to be found from its line integrals over a certain set $\mathcal{R}$ of straight lines in three-dimensional space. Image reconstruction algorithms for 3D PET are conceptually very similar to those described for the bidimensional setting; however, the set of acquired data has very different properties and so it requires special handling, either by additional processing before applying an algorithm similar to the 2D case, or by modifying the algorithm itself. The following are the main differences in the measurements:

- Acquired data in 3D PET are highly redundant. In the 2D case, the set of projections determines the underlying distribution f univocally; hence, in a 3D setting, it would suffice to obtain the set of projections in planes that are parallel to a given direction, e.g. to the scanner rings, to univocally determine the sections of f along such planes, and therefore at any point in space. The availability of projection data contained in planes with different directions thus implies data redundancy, although these data are not consistent due to the presence of statistical noise. A “good” 3D reconstruction algorithm will take this into account in order to obtain a better signal to noise ratio than in the 2D case.

- In 2D PET, the detector ring completely surrounds the region under study, so measurements are possible for any LOR intersecting the distribution. In 3D PET, the scanner has a finite axial dimension and so there will be directions along which the acquired projections are only partially available, or even completely unavailable.

[21] In the original description, the effect of positron range is taken into account simultaneously.

- The influence of scattered events is much higher than in the 2D case. In 2D reconstruction, the 3D solid angle covered by each detector ring is relatively small, and detection of a scattered event as a coincident pair is only possible if the scatter angle is small.[22] However, in 3D PET the scanner covers a much larger solid angle from each detector point and most scattered photons are detected and registered as valid events on wrong LORs. Hence, a scatter correction method is mandatory in 3D tomography, whereas the effect of scattered events in 2D PET can largely be ignored.

Projections

Projections in 3D space are defined analogously as in 2D as collections of integrals along all LORs parallel to a given direction. In this case, the direction is specified by two angles that define a unit vector in polar coordinates

$$\mathbf{u}_{\theta,\varphi} = (\cos\theta\cos\varphi, \cos\theta\sin\varphi, \sin\theta) \tag{4.73}$$

and the projection is a function $p_{\theta,\varphi} : \mathbb{R}^2 \to \mathbb{R}$ that yields the integral of f over the line that is parallel to $\mathbf{u}_{\theta,\varphi}$ at each point on its normal plane. It can be expressed as

$$p_{\theta,\varphi}(s,t) = \int_{-\infty}^{\infty} f\left(R_{\theta,\varphi}(s,t,u)\right) du \tag{4.74}$$

where $R_{\theta,\varphi} : (s,t,u) \mapsto (x,y,z)$ is the rotation by angles θ and φ that maps vector $\vec{\mathbf{k}} = (0,0,1)$ into $\mathbf{u}_{\theta,\varphi}$.

Considering the detector geometry, there are bounds on the values of θ where it is possible to obtain useful projections. As depicted in Fig. 4.26, there are limit angles θ_F and θ_D, whose values depend on the detector geometry and the size of interaction region, with the following properties:

- All projections with $|\theta| < \theta_F$ are complete.
- Projections for $|\theta| > \theta_F$ are truncated i.e. incomplete, as some lines going through the region under study intersect the detector cylinder at only one point, while others intersect at two points and define a valid LOR.
- If $|\theta| > \theta_D$, then the associated projection cannot be evaluated at any point, as lines only intersect the detector cylinder at one point at most.

For most practical applications, the object under study is longer than the scanner's axial dimension and so $\theta_F = 0$. All projections are therefore incomplete,

[22] Or in the unlikely case where both photons are scattered in such a way that they end up hitting the same detector ring.

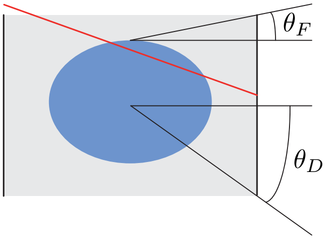


Figure 4.26: Limit angles θ_F and θ_D for a cylindrical PET scanner. The red line is an example with $\theta_F < \theta < \theta_D$ that intersects the detector ring at only one point and thus yields an incomplete projection.

except those corresponding to $\theta = 0$, i.e. those employed in 2D PET. Usually, a maximum angle $\theta_{\max} < \theta_D$ is chosen such that all projections with $|\theta| > \theta_{\max}$ are discarded, and those with $|\theta| < \theta_{\max}$ are completed prior to 3D reconstruction, i.e. their unknown values are estimated using a different method. The value of $\theta_{\max}$ establishes a tradeoff between the reconstructed image quality (as more measurements are used) and the increased computational effort for the estimation of partial projections. Typical values for $\theta_{\max}$ are around 10° to 15° [304].

Direct reconstruction

The three-dimensional version of the Central Section Theorem is obtained in a similar fashion as the 2D case by considering the Fourier transform of the projections

$$P_{\theta,\varphi}(\sigma,\tau) = \mathcal{F}_{2\mathrm{D}}\left[p_{\theta,\varphi}(s,t)\right] = \iint p_{\theta,\varphi}(s,t)\, e^{-2\pi i(s\sigma+t\tau)}\, ds\, dt \tag{4.75}$$

and of the distribution f

$$F(\xi,\eta,\zeta) = \mathcal{F}_{3\mathrm{D}}\left[f(x,y,z)\right] = \iiint f(x,y,z)\, e^{-2\pi i(x\xi+y\eta+z\zeta)}\, dx\, dy\, dz \tag{4.76}$$

and then substituting (4.74) into (4.75) to obtain

$$\begin{aligned}
P_{\theta,\varphi}(\sigma,\tau) &= \iint p_{\theta,\varphi}(s,t)\, e^{-2\pi i(s\sigma+t\tau)}\, ds\, dt \\
&= \iiint f\left(R_{\theta,\varphi}(s,t,u)\right) e^{-2\pi i\langle(s,t,u),(\sigma,\tau,0)\rangle}\, ds\, dt\, du \\
&= \iiint f(x,y,z)\, e^{-2\pi i\left\langle R^{-1}_{\theta,\varphi}(x,y,z),(\sigma,\tau,0)\right\rangle}\, dx\, dy\, dz \\
&= \iiint f(x,y,z)\, e^{-2\pi i\langle(x,y,z),R_{\theta,\varphi}(\sigma,\tau,0)\rangle}\, dx\, dy\, dz \\
&= F\left(R_{\theta,\varphi}(\sigma,\tau,0)\right). \qquad (4.77)
\end{aligned}$$

Thus, the 3D Central Section Theorem can be rephrased as follows: the Fourier transform of the projection of f along direction $\mathbf{u}_{\theta,\varphi}$ is the planar section of the transform of f across the plane normal to $\mathbf{u}_{\theta,\varphi}$. In the derivation of this formula, the symbol $\langle\cdot,\cdot\rangle$ has been used for the scalar product in $\mathbb{R}^3$. The third equality is a change of variable given by the rotation $R_{\theta,\varphi}$ with Jacobian determinant equal to 1, and the fourth follows from the fact that rotations are isometries i.e. the scalar product is invariant under rotations.

Eq. (4.77) provides a way to reconstruct the Fourier transform F of f from the projections $p_{\theta,\varphi}$ in a similar way as (4.63). However, the data given by 3D projections are redundant and the value of F at a given point (ξ,η,ζ) can be estimated using different projections, yielding inconsistent results. It is usually assumed that the optimal value in terms of signal to noise ratio is obtained by averaging all possible estimations [304].

The projections $p_{\theta,\varphi}$ that can be used to compute $F(\xi,\eta,\zeta)$ are precisely the projections on planes that contain the origin and the point (ξ,η,ζ), i.e. such that $(\xi,\eta,\zeta)\perp\mathbf{u}_{\theta,\varphi}$. Representing $(\xi,\eta,\zeta)=\rho\cdot\mathbf{u}_{\vartheta,\psi}$ in polar coordinates, the orthogonality condition is equivalent to

$$\cos(\varphi-\psi) = -\operatorname{tg}\theta\operatorname{tg}\vartheta \qquad (4.78)$$

that can be fulfilled only if the absolute value of the right term does not exceed 1. Eq. (4.78) provides a bound on the values of θ that may be used to recover F in the specified point, the other bound being θ_{max}. Thus, valid estimations can be obtained using projections with $|\theta|\le\theta_{\mathrm{lim}}$ where

$$\theta_{\mathrm{lim}}(\vartheta,\theta_{\mathrm{max}}) = \begin{cases} \theta_{\mathrm{max}} & \text{, if } \cos\vartheta > \sin\theta_{\mathrm{max}} \\ \pi/2-\vartheta & \text{, if } \cos\vartheta \le \sin\theta_{\mathrm{max}} \end{cases}. \qquad (4.79)$$

Using these limits, the optimal estimation of $F(\xi,\eta,\zeta)$ is obtained by averaging all possible values uniformly over the valid solid angle

$$F(\rho\cdot\mathbf{u}_{\vartheta,\psi}) = \frac{1}{N(\vartheta,\theta_{\mathrm{max}})}\int_{-\theta_{\mathrm{lim}}}^{\theta_{\mathrm{lim}}} \frac{2\cos\theta}{\sqrt{\cos^2\vartheta-\sin^2\theta}} P_{\theta,\varphi}(\sigma,\tau)\, d\theta \qquad (4.80)$$

where the variables inside the integral depend on θ as follows: the value of φ is the one that fulfills (4.78), and σ and τ are given by $(\sigma, \tau, 0) = R^{-1}_{\theta,\varphi}(\xi, \eta, \zeta)$ as implied by the Central Section Theorem. The weighting factor for $P_{\theta,\varphi}$ inside the integral is there to ensure uniformity over solid angles, and is compensated by the normalization factor

$$\begin{aligned} N(\vartheta, \theta_{\max}) &= \int_{-\theta_{\lim}}^{\theta_{\lim}} \frac{2\cos\theta}{\sqrt{\cos^2\vartheta - \sin^2\theta}} d\theta \\ &= \begin{cases} 4 \arcsin \dfrac{\sin\theta_{\max}}{\cos\vartheta} & \text{, if } \cos\vartheta > \sin\theta_{\max} \\ 2\pi & \text{, if } \cos\vartheta \le \sin\theta_{\max} \end{cases} . \end{aligned} \tag{4.81}$$

Direct reconstruction thus consists in using (4.80) to obtain the values of F at each point from the completed projections $p_{\theta,\varphi}$ and is the 3D equivalent to direct 2D reconstruction using (4.63). However, this method suffers from the same drawbacks as the 2D version due to discretization, i.e. the lattice of points where $P_{\theta,\varphi}$ is known is not adequate for the computation of F. A 3D version of FBP is thus a preferred choice.

Filtered backprojection

The formulae for 3D FBP can be obtained following the same steps as in the derivation of the 2D case, but the computations are much more complicated. The resulting expression for the distribution f is

$$f(x, y, z) = \int_{-\theta_{\max}}^{\theta_{\max}} \int_0^{\pi} b_{\theta,\varphi}(s, t) \cos\theta \, d\varphi \, d\theta \tag{4.82}$$

where s and t are such that the LOR associated to $p_{\theta,\varphi}(s, t)$ contains the point (x, y, z), and the filtered projection is given by

$$b_{\theta,\varphi}(s, t) = \mathcal{F}^{-1}_{2\text{D}} \left[P_{\theta,\varphi}(\sigma, \tau) \, H_C(\sigma, \tau, \theta) \right] . \tag{4.83}$$

The filter H_C, known as the *Colsher filter* [306], is the 3D equivalent of the one-dimensional ramp filter but includes an angular correction factor in order to account for the averaging effect over all possible estimations of F. It can be expressed as

$$H_C(\sigma, \tau, \theta) = \frac{\sqrt{\sigma^2 + \tau^2}}{N(\vartheta, \theta_{\max})} \tag{4.84}$$

where the angle ϑ is the result of a change to polar coordinates and given by

$$\vartheta = \operatorname{arctg} \sqrt{\frac{\tau^2 \cos^2\theta}{\sigma^2 + \tau^2 \sin^2\theta}} . \tag{4.85}$$

The same considerations apply here as in the 2D case regarding the weighting of high frequencies by H_C and the resulting amplification of aliasing and statistical noise. The typical solution is also the same: a window filter $W_{2\mathrm{D}}$ is applied as

$$b_{\theta,\varphi}(s,t) = \mathcal{F}_{2\mathrm{D}}^{-1}\left[P_{\theta,\varphi}(\sigma,\tau)\, H_C(\sigma,\tau,\theta)\, W_{2\mathrm{D}}(\sigma,\tau)\right]. \tag{4.86}$$

The two-dimensional window filter may be an adaptation of the standard 1D Hamming window $W_{1\mathrm{D}}$ in (4.69) given by either $W_{2\mathrm{D}}(\sigma,\tau) = W_{1\mathrm{D}}(\sigma)\, W_{1\mathrm{D}}(\tau)$ or $W_{2\mathrm{D}}(\sigma,\tau) = W_{1\mathrm{D}}\left(\sqrt{\sigma^2+\tau^2}\right)$.

Practical implementations of 3D FBP estimate f using a discretized formula similar to (4.68), accumulating the values from filtered projections one by one. The main difference with the 2D case is that the number of LORs, and hence of filtered projections, is significantly higher. The extension to ToF PET is also similar and amounts to convolution with a kernel representing the Gaussian estimation of the position along the LOR [307].

3D reprojection

The FBP algorithm needs a set of complete projections in order to obtain an estimation of the distribution f, but the only projections whose values can be measured at any point are those corresponding to $|\theta| \leq \theta_F$, and typically $\theta_F = 0$. Applying the algorithm directly on truncated projections has been shown to result in very noisy images, particularly at positions far away from the center of the scanner ([302], pp. 158–166). These projections are thus completed prior to FBP by estimating their value on the missing points.

The most common completion algorithm is *3D reprojection* (3DRP) introduced in [308]. The starting point for this method is a crude estimation f^0 of the field f. f^0 can be obtained by using only the projections with $\theta = 0$ (i.e. computing the 2D sections across each ring plane); a better estimation is obtained by applying FBP to the set of projections with small values of θ, exploiting redundancy in order to reduce noise. The projections $p^0_{\theta,\varphi}$ of f^0 are then computed directly as line integrals. Finally, the truncated projections $p_{\theta,\varphi}$ are completed with the values of $p^0_{\theta,\varphi}$ at all points where they could not be measured.

Reduction to the 2D case

A separate collection of 3D reconstruction methods, generically called *rebinning* algorithms, is based on a completely different idea. The goal is to reduce the dimensionality of the problem; specifically, 3D reconstruction using 3D data is to be simplified into the 2D case. The motivation behind these methods is to avoid the main drawbacks of 3D reconstruction, i.e. computational and data storage requirements, by sacrificing some image quality. In order to do so, a procedure is

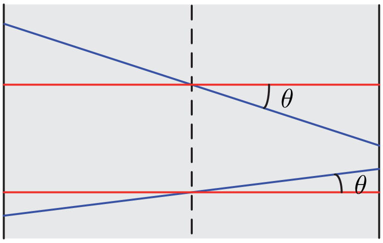


Figure 4.27: Axial view of the PET scanner and illustration of the SSRB algorithm. 3D LORs (blue) are converted into 2D LORs (red) by ignoring angle θ formed with the ring planes.

sought for condensing the full set of 3D measures indicating the activity measured along each 3D LOR into an equivalent set $\left\{p_{\varphi}^{z}\right\}$ of 2D projections, containing only LORs that are parallel to the detector rings at different positions z along the central scanner axis. A 2D reconstruction method is then applied to the projections p_{θ}^{z} for each value of z in order to obtain slices of f.

The simplest form of rebinning is *Single Slice Rebinning* (SSRB) [309]. This method consists on ignoring the angle θ altogether, transforming each oblique LOR into its projection on the z plane passing through its center point as shown in Figure 4.27. The most important advantage of SSRB is that it can be performed online and assimilated into the data acquisition process by taking measurements directly as in 2D PET. This ensures vastly reduced computational load and storage resources. However, disregarding θ implies large estimation errors in positions far away from the central axis, hence this method can only be applied to small, relatively symmetric objects or for small $\theta_{\max}$.

A more elaborate algorithm is *Fourier Rebinning* (FORE), where the dimensionality reduction is applied on the frequency domain [310]. This method is based on the notion that each value of the projection transforms $P_{\theta,\varphi}$ depends only on the values of the distribution f within a certain, bounded distance. Applying this principle of locality, a value of z can be assigned for each point in the transform domain, and the original transforms $P_{\theta,\varphi}$ can then be recombined in order to obtain 2D projections P_{φ}^{z} for each z. This method yields better results than SSRB, but requires the previous acquisition of all 3D data in order to compute the original Fourier transforms, so the system storage requirements are not avoided.

4.4.4 Iterative reconstruction methods

The previous sections have described the most extended analytical reconstruction methods, that return an estimation of the image f by applying what is essentially a (possibly very complex) closed formula to the set of acquired data. In stark contrast to them, iterative methods start with an initial estimation of f and progressively refine it by applying the same processing step to it over and over until a satisfactory solution is reached. Several iterations are required before convergence, however it is expected that each iteration step carries a smaller computational load than a full analytical reconstruction, so as to balance the execution time between both alternatives.

Iterative algorithms are currently recognized to possess several advantages over the traditional direct methods:

- Iterative methods are based on models of data acquisition by the scanner that are much more flexible than the line integral approximation that has been considered in the previous sections. This allows a more exact modeling of the relationship between the activity measured on the LORs and the underlying radioactivity.

- In the same way, models of statistical noise inherent to the measurements are taken into account; specifically, they are modeled as Poisson distributions. This allows a more selective correction against statistical noise than that applied by FBP, i.e. a window filtering of projections that succeeds in removing statistical noise for the most part but does so at the expense of increased image distortion, causing a smoothing effect that ultimately results in a loss of resolution [311].

- Many iterative methods are able to incorporate *a priori* information about the image to be reconstructed, either for faster convergence of for better results. This information may be as complex as the results of previous reconstructions or as simple as the fact that the field f is non-negative. This results in much more reliable images. As a simple example of this, consider the fact that an analytic method such as FBP applied to noisy or incomplete data might conceivably yield a solution f with negative values at some points, something that clearly has no physical meaning. Similarly, the resulting image f could describe a structure that makes no sense from a medical or anatomical point of view. The use of previous information in iterative methods prevents or minimizes the probability of such anomalies.

- Analytical methods often yield poor results when the amount of acquired data is small, due to the strong effect of statistical noise. Hence, in PET scanners or other situations with a relatively small acquisition time, it is often preferable to resort to statistics-based iterative methods, which have been shown to provide better image quality under conditions of low activity [312] and incomplete data sets [313]. A different way of exploiting this advantage is through the reduction of acquisition time and of exposition of the patient to radiation.

Most of the iterative methods employed in medical imaging assume a linear model for data acquisition, whereby the relationship between the variables being sought and the measured values, i.e. the radiotracer concentration and the photon pairs along each LOR, can be expressed as a linear operator and a noise component with zero mean. Both are explicitly discretized from the start: the concentration f is to be determined on a finite grid of pixels or voxels and so its variable is represented as a finite vector $\mathbf{f}$ with length N equal to the total number of voxels. Similarly, the set of measurements is represented as a vector $\mathbf{p}$ whose length is the size of the set $\mathcal{R}$ of LORs where incidence data are being collected, or of discrete LOR segments in the case of ToF reconstruction.

Let $\mathbf{n}$ be the vector of length N that contains the amount of photon pairs actually generated at each voxel. Then $\mathbf{n}$ is a random variable whose components have Poisson distributions and its expectation is $E[\mathbf{n}] = \mathbf{f}$ (up to a proportionality constant). The hypothesis of a linear generation method means that

$$\begin{aligned} \mathbf{p} &= \mathbf{A} \cdot \mathbf{n} \\ E[\mathbf{p}] &= \mathbf{A} \cdot \mathbf{f} \end{aligned} \tag{4.87}$$

for a certain constant matrix $\mathbf{A}$, called the *transition matrix*. Its coefficients a_{ij} represent the conditional probability that a given event is detected on the j-th LOR or segment, with the knowledge that it was generated at the i-th voxel. The transition matrix depends on the geometry of the PET scanner and may convey a very exact modeling of the acquisition system by including information about non-ideal scanner response like positron range, angular deviation and non-uniformity in the detectors. Correction for attenuation and scattering may also be automatically incorporated into the process [314]. It should be noted that this approach is independent of the dimensionality of the reconstruction problem i.e. the procedure is the same for 2D and 3D PET, the only difference being the size of the problem.

The ML-EM algorithm

Any numerical method of iterative optimization is based on the evaluation of the space of admissible solutions through a *cost function*, i.e. a scalar function that is used as an estimation of the goodness of the selected solution, and its optimization (i.e. search of its maximum or minimum) following an iterative algorithm. The classic statistical method of Maximum Likelihood (ML) can be applied directly in this case, and consists of using the *likelihood function* L as a cost function and then searching for its maximum. The likelihood is defined as

$$L\left(\mathbf{f}\right) = p\left(\mathbf{p}\left|\mathbf{f}\right.\right) \tag{4.88}$$

i.e. the conditional probability of measuring the data $\mathbf{p}$ that were actually measured, if the radiotracer distribution were $\mathbf{f}$. The distribution $\mathbf{f}$ that maximizes $L\left(\mathbf{f}\right)$ is the one that has the highest probability of generating the data $\mathbf{p}$ acquired by the scanner.

Practical expressions of the likelihood function can be obtained by considering the particular statistical distribution in the measurements. In this case, each element of vector $\mathbf{p}$ is a linear combination of independent Poisson distributed variables and hence Poisson distributed itself ([32], pp. 57–58), so that

$$p\left(p_j = k\left|\mathbf{f}\right.\right) = e^{-E\left[p_j\right]}\frac{E\left[p_j\right]^k}{k!} \tag{4.89}$$

where $E\left[p_j\right] = E\left[p_j\left|\mathbf{f}\right.\right]$ is the mean value of the j-th element in vector $\mathbf{p}$, whose relationship with $\mathbf{f}$ is given by the transition matrix as described in (4.87), specifically

$$E\left[p_j\left|\mathbf{f}\right.\right] = \sum_i a_{ij} f_i. \tag{4.90}$$

The likelihood can then be obtained as the product of the probabilities for each individual component of $\mathbf{p}$, i.e.

$$L\left(\mathbf{f}\right) = p\left(p\left|\mathbf{f}\right.\right) = \prod_j p\left(p_j\left|\mathbf{f}\right.\right) = \prod_j e^{-E\left[p_j\left|\mathbf{f}\right.\right]}\frac{E\left[p_j\left|\mathbf{f}\right.\right]^{p_j}}{p_j!}. \tag{4.91}$$

In order to reduce this to a simpler formula, the *log-likelihood* is usually used instead of the likelihood, since their maximization is mathematically equivalent:

$$\log L\left(\mathbf{f}\right) = \sum_j \left(-E\left[p_j\left|\mathbf{f}\right.\right] + p_j \log E\left[p_j\left|\mathbf{f}\right.\right] - \log p_j!\right). \tag{4.92}$$

Finally, the last term $\log p_j!$ summed over all components j is a constant value so it can be discarded when searching for the maximum. Substituting in (4.90), the ML cost function is obtained:

$$\begin{aligned} L_{\mathrm{ML}}\left(\mathbf{f}\right) &= \sum_j \left(p_j \log E\left[p_j\left|\mathbf{f}\right.\right] - E\left[p_j\left|\mathbf{f}\right.\right]\right) \\ &= \sum_j \left(p_j \log\left(\sum_i a_{ij} f_i\right) - \sum_i a_{ij} f_i\right). \end{aligned} \tag{4.93}$$

Maximum likelihood reconstruction methods will strive to find an image estimation $\mathbf{f}$ that maximizes this value. The most common approach is *Maximum Likelihood-Expectation Maximization* (ML-EM) [315]. This method starts with an initial estimation $\mathbf{f}^{(0)}$ of the solution that may be obtained, for instance, by slicewise 2D FBP as in the case of 3D reprojection. It then searches for a maximum of L_{ML} according to the following rule: in the k-th iteration, the last estimation $\mathbf{f}^{(k)}$ is projected on the space of measurements i.e. the vector $\mathbf{p}^{(k)} = \mathbf{A} \cdot \mathbf{f}^{(k)}$ is obtained, corresponding to the expected measurements that would have been acquired if the radiotracer distribution were really equal to the current estimation $\mathbf{f}^{(k)}$. This vector is then used to correct the estimation on a component by component basis as

$$f_i^{(k+1)} = f_i^{(k)} \frac{\sum_j a_{ij} \frac{p_j}{p_j^{(k)}}}{\sum_j a_{ij}}. \tag{4.94}$$

It can be proven that the ML-EM algorithm converges to the maximum likelihood solution eventually [315]. One of the advantages of this algorithm is that the iterative step does not change the sign of the f_i, hence each estimation $\mathbf{f}^{(k)}$ is non-negative if the initial estimation $\mathbf{f}^{(0)}$ was. Additionally, the images obtained with ML-EM are smoother than those obtained with FBP. However, there are some disadvantages to ML-EM, at least in its initial form as presented above. The algorithm suffers from extremely slow convergence, requiring around 30 to 50 iterations in order to reach acceptable solution, and each iteration has a computational load that is comparable to that of the FBP algorithm on the same set of data. Moreover, the estimated images start losing quality after a certain iteration, as will be discussed in the next subsection.

Several modifications of the original ML-EM algorithm have been developed in order to overcome its main drawback, the extremely high execution time. Most of them are direct applications of generic convergence acceleration methods for iterative algorithms. For instance, the SAGE method (*Space-Alternating Generalized EM*, [316]) consists in updating the projections $\mathbf{p}^{(k)}$ within each iteration, i.e. after each component of group of components $f_i^{(k+1)}$ is obtained using (4.94), the projection $\mathbf{p}^{(k)}$ is corrected so that it includes the effect of the estimated components. This results in a higher computational load per step, but reduces the number of required iterations.

The most popular variation of ML-EM is OSEM (*Ordered Subsets EM*, [317]), that focuses on the reduction of execution time for each iteration by incorporating some physical information in the formulae. This is accomplished by substituting the sums over all j in (4.94) with sums over reduced subsets of LORs or segments that are assumed to have the highest influence on the i-th voxel that is currently being estimated. The amount of computations for the estimation of each $\mathbf{f}^{(k)}$ is considerably reduced: if the set $\mathcal{R}$ is split into K non-overlapping subsets, then OSEM is approximately K times faster than the original ML-EM. While OSEM does not always converge to the maximum likelihood solution [317], it has been empirically validated that it yields acceptable solutions.

Regularization

Iterative methods in general, and derivatives of ML-EM in particular, are prone to an over-specialization problem that causes the obtained estimations to start degrading in quality after a certain iteration. The reason behind this effect is the fact that the solutions that maximize the likelihood and the ones with the best image quality do not necessarily match. Such a problem arises whenever the cost function does not describe the desired characteristics of the result exactly, because they are too complex to be modeled by a scalar function that needs to have a relatively simple expression for computational reasons.

During optimization of the cost function, the image estimation improves initially after each iteration, but after a certain point, the estimations start to specialize with respect to the cost function and, as a result, their quality starts to drop. This usually takes the form of speckles, oscillations and "checkerboard" effects in the images. One possible solution is to establish a criterion for early termination of the algorithm, that takes image quality parameters into consideration and stops just before specialization. Unfortunately, it is very difficult to define robust stopping criteria [318].

Another solution is to include a *regularization* constraint in the iterative method that enforces smoothness in the image. Several regularization approaches are possible. For instance, the cost function may be modified as

$$L\left(\mathbf{f}\right) = L_{\mathrm{ML}}\left(\mathbf{f}; \mathbf{p}, \mathbf{A}\right) + R\left(\mathbf{f}\right) \tag{4.95}$$

where $R\left(\mathbf{f}\right)$ is a penalty function that is independent of the measurements $\mathbf{p}$ or the system parameterization $\mathbf{A}$ but rather depends only on the image estimation $\mathbf{f}$, and is used as a measure of its inherent fitness as a solution to the problem. This is one possible way to incorporate *a priori* information into the iterative method; $R\left(\mathbf{f}\right)$ may only measure the smoothness of $\mathbf{f}$ or it can model more complex parameters. In any case, the iterative update rule must be modified in order to include the term $R\left(\mathbf{f}\right)$ into its maximization [319].

Other regularization methods involve smoothing the estimated images locally as they are obtained. For instance, the Median Root Prior (MRP) algorithm [320] modifies the ML-EM iteration step as

$$f_i^{(k+1)} = f_i^{(k)} \frac{\sum_j a_{ij} \frac{p_j}{p_j^{(k)}}}{\sum_j a_{ij}} \cdot \left(1 + \beta \frac{f_i^{(k)} - \operatorname{med}\left(\mathbf{f}^{(k)}, i\right)}{\operatorname{med}\left(\mathbf{f}^{(k)}, i\right)}\right)^{-1} \tag{4.96}$$

where $\operatorname{med}\left(\mathbf{f}^{(k)}, i\right)$ is the median of the voxel values of $\mathbf{f}^{(k)}$ in a neighborhood of the i-th voxel.[23] The smoothing process is controlled by coefficient β. This modified

[23] This *median filtering* is a very common operation for noise reduction in digital image processing ([321], pp. 104–105).

method does not converge to the maximum likelihood solution but succeeds in removing a large part of the noise in the image while preserving its most important features.

Chapter 5

DAQ systems for PET

Last chapter has been devoted to the description of PET imaging together with its physical, electronic and mathematical principles from the generation of events within the patient up until the display of a reconstructed image on a PC. In this chapter, PET is identified as a particular case of HEP setups, and the general study of DAQ systems performed in chapter §2 is applied in order to analyze the architecture of PET DAQs, identify their key components, and comment on the existing systems. A specific DAQ architecture is then proposed as a central element of this thesis, based on the trends and conclusions obtained in §2.2.2, that takes advantage of the technological advances on synchronization exposed in chapter §3.

5.1 DAQ system architecture

In terms of the generic description of DAQ systems for experimental physics setups given in §2.1, PET scanners are a particular case whose data flow is completely characterized by the following features:

- The variable under study is the concentration of a radiotracer inside a given target sample. The target may be assumed to be stationary for short periods of time.

- The events are β^+ decays that result in two almost diametrically opposed annihilation photons with 511 keV each, ejected along a line that intersects the radiotracer distribution.

- Events take place spontaneously without an accelerator. Their times of occurrence do not match a fixed bunch pattern, but follow a Poisson process instead.
- The interaction region i.e. the FOV has a relatively small size (diameter under 1 m) and is surrounded by one or more rings of identical detectors, consisting of a scintillator calorimeter that is sensed by photodetectors.[1]
- Each annihilation photon triggers a subevent at a detector by interacting with its scintillator. As a result, the photodetectors generate pulses with durations in the order of 50 ns to 1000 ns whose amplitude or charge is proportional to the deposited energy.
- Local triggers are issued at each detector whenever timing pulses cross a voltage threshold.
- For each accepted subevent, data is extracted from the captured pulse samples that contain information about photon energy, scintillation position, and a time reference.
- The complete event trigger consists in finding two subevents from valid detector pairs, each within a given energy window, happening within a time coincidence window. The event builder generates the corresponding LOR, and possibly an estimation of location along the LOR in the case of ToF PET.
- The high-level computing stage applies an image reconstruction algorithm in order to convert the list of LORs (and LOR locations in ToF PET) into an estimation of the radiotracer distribution. This is usually accomplished offline with software or GPU implementations.

Other PET system specifications and requirements are listed here:

- Radiation hardness is not required for PET instrumentation, as the nature of the annihilation radiation potentially reaching the electronics after failing to be absorbed by scintillators is unlikely to damage it or even cause an observable effect.
- Time resolutions below a few ns for standard PET and 500 ps for ToF PET are required, implying that the synchronization accuracy between timing elements in the system needs to be much lower than that.
- Errors such as the acceptance of invalid events or poor estimation of event features are not critical and only result in added noise in the reconstructed

[1] For dual-modality scanners such as PET/MR, only the PET section is being considered here.

image. Their contribution is merely statistical and one only needs to keep their rate within bounds, rather than implement sophisticated algorithms to completely filter them out. This may not be the case in HEP experiments where new particles are expected to be discovered in only a few total events and thus a large degree of certainty is needed.

- While PET detectors are smaller in scope than large experimental setups, they consist largely of replicated, identical modular elements. Also, PET scanner designs are usually meant for continued production, as opposed to large, specific setups that are only meant to be assembled once. This has an impact on economic considerations, e.g. it may be a good idea to develop custom ASICs as their fixed development cost is less relevant with large production sizes.

The main elements and architectural choices in the DAQ system of PET scanners are described next, comprising all functions from the detection of radiation to the generation of LORs and their transmission to the image reconstruction device; subevent detection and the trigger subsystem are considered separately. Finally, a small review of existing PET DAQ systems is presented.

5.1.1 Subevent processing

Recall that each PET sensor block typically consists of an array of small scintillator crystals, or alternatively a large monolithic crystal covering the same incidence area, with attached photodetectors (PMT, APD or SiPM) that generate current signals whose total charge is proportional to the collected scintillation light. These signals are usually processed by preamplifiers in order to guarantee a reasonable SNR and isolation from the readout electronics. Subevent features (energy, position and time) are then extracted from the preamplified signals using various methods, possibly including digital processing techniques.

Front-end and readout

The first processing step applied to the detector signals, either before or after preamplifying them, is usually a reduction in the number of channels, compressing the information from each physical source into a smaller number of signals in such a way that there is no significant loss of information. The main goal behind this operation is to simplify the electronics and reduce their cost by lowering the amount of signals to be tracked. This technique is usually called *multiplexing*. At the very least, it has the effect of reducing the number of channels that need to be digitized and therefore the amount of ADCs in the system; this results in reduced cost, area, and power consumption. As a downside, this may increase the need in digital processing resources as more intensive processing may be needed to decompress and interpret the information contained in the multiplexed signals.

Multiplexing is introduced in the optical domain between a pixelated crystal and the photodetector plane whenever there is not a one-to-one correspondence between pixels and detectors. This technique makes use of light guides to distribute and transport optical photons to the active photodetector area. For instance, introducing a common, continuous transparent medium after the crystals allows for position reconstruction using fewer detectors, e.g. four PMTs using the equivalent of two-dimensional Anger logic as introduced by classic block detectors [224]. A different approach involves the distribution of pixel outputs using optical fibers; specific combinations may then be hardwired, such as row-column encoding i.e. summing all rows and all columns of a $n \times n$ crystal matrix in parallel, obtaining $2n$ optical outputs that need to be sensed [322].

Electrical multiplexing is also possible for the photodetector output pulses. The paradigmatic examples of this technique are the DPC resistor networks introduced in §4.3.2. These schemes may be applied either before or after the preamplifying stage. In the first case, the amount of preamplifiers is also reduced, but this carries a subtle drawback. Photodetector outputs can always be used to drive several passive combination circuits at the same time in order to generate several multiplexed outputs as long as the input impedances are balanced between them; however, if these impedances are not equal for all photodetector signals, the frequency response is not uniform for all of them and position-dependent artifacts are introduced unless a slow shaping acquisition mode is implemented [253]. The only way to avoid this problem in generic multiplexing schemes is by preamplifying all photodetector outputs.

Some specific examples of electrical multiplexing include Anger logic circuits such as the Siegel network in Fig. 4.13 [231], or row-column encoding [234, 323], which may in turn be compressed into four signals using two one-dimensional Anger networks [324]. For pixelated detectors, particularly those with no optical multiplexing step where only one photodetector is expected to activate per subevent, optimal reduction may be achieved by binary encoding schemes where n photodetector outputs are combined into $\log_2 n$ signals whose activation yields the binary address of the corresponding pixel [325]. A generalized version of this idea is given by the *compressed sensing* method [326], where more than $\log_2 n$ signals are generated in such a way that it is possible to recover multi-pixel subevents. All of these linear multiplexing schemes are in turn generalized by the AMIC device family developed at EDNA, that allows the generation of up to 8 arbitrary programmable linear combinations of 64 photodetector outputs [256].

It is of course possible to employ no multiplexing technique at all. This results in a bulky, expensive system, but it is the approach taken in some cases such as [227, 323] where all detector signals need to be digitized for complex position estimation algorithms.

Feature extraction

As explained in §4.3, there are three key pieces of information that need to be extracted from the analog signals for each subevent: energy, position, and time of the interaction. From the point of view of the acquisition, the first two behave similarly and are fundamentally different from the last one, in that the relevant information is essentially contained in the total signal charge whereas timing information is given by the signal shape, as already outlined in §2.1.1. Consequently, the impact of the electronic readout system on energy resolution is dominated by SNR, whereas the impact on time resolution is determined by jitter and rise time [287].

DAQ systems may be classified independently by the type of readout used for each of those kinds of signal. For position and energy, the possible approaches are determined by either slow or fast sampling:

- In the first case, a slow shaping with large time constant τ is applied to the signal and its peak value is captured, which is proportional to the total charge and thus to the magnitude under consideration. This is the procedure followed by older systems such as [196], and has the drawback of a large dead time given by the shaped pulse length, that limits the supported count rate.
- The other case usually involves ADCs in FRS mode followed by digital processing where the magnitude is extracted from several samples using more elaborate algorithms. This allows fast shapings and removes dead time, particularly if digital pile-up recovery mechanisms are implemented [327], as well as providing a more flexible and update-friendly environment, but the resulting circuitry is more expensive and power hungry.

In the case of timing signals, there are also two approaches that more or less correspond to those described above for charge signals:

- The first one involves digitization by way of a comparator applied to the timing signal, which may itself be a raw photodetector pulse or a CFD output, such that a digital pulse is generated whose rising edge identifies the time reference for the subevent. This pulse may then be transformed into a digital timestamp with a TDC or used directly for gated coincidence detection without assigning a digital value to it, as will be described in §5.1.2. This approach may be generalized to the Time over Threshold (ToT) [324] or Multi-Voltage Threshold (MVT) [328] techniques by using either several edges or several comparator thresholds, respectively. In general, this approach results in better time resolution, but it breaks the pipelined DAQ flow and therefore always introduces dead time [329].

- The second method consists in sampling of the timing signal and digital processing for timestamp extraction. In this case, it is possible to use free-running sampling with relatively low sampling frequencies, in the order of 100 MHz, or transient digitizers such as analog memories working in the GHz range [213]. The latter may improve coincidence resolutions by one order of magnitude [271], but this comes at the cost of vastly increased dead time, whereas FRS approaches usually introduce no dead time at all.

Postprocessing techniques on digitized data, be it energy correction, position estimation or timestamping from sampled waveforms, may be implemented online at the subevent detector or left to a later stage, possibly offline. In the latter case, all necessary data need to be transmitted per subevent, vastly increasing output bandwidth requirements for a fixed rate of singles. In the former case, shortened subevent descriptions are transmitted, but the computational load at the detector modules increases and thus its circuit complexity and power consumption.

5.1.2 Trigger and coincidence detection

In a PET DAQ system, the complete event trigger consists in finding a pair of subevents, each within the valid energy window, happening within a time coincidence window, and such that the corresponding LOR intersects the FOV. Energy estimation occurs at the subevent stage, and LOR validation usually amounts to discarding random coincidences between invalid pairs of block detectors, e.g. those located along the same radial direction. The critical element in the trigger subsystem is thus the coincidence detection engine.

Coincidence detection schemes

There are two possible, fundamentally different ways to implement coincidence detection in a PET scanner: gated and timestamp-based approaches. They correspond roughly to readout schemes with and without L1 trigger as depicted in Fig. 2.5, but not exactly and some variations are possible. In a gated approach, discriminators in each detector generate timing pulses whenever a signal threshold is crossed and a subevent is detected. These pulses are then transported to a common coincidence engine that looks for pairs of signals that are active at the same time. On the other hand, timestamp-based coincidence detection relies on the assignment of a digital timestamp to all detected subevents independently, and a later comparison of timestamps in the digital domain in order to determine the presence of coincident pairs.

In its simplest implementation [330], gated coincidence detection amounts to an OR-AND or AND-OR plane of gates comparing pairs of incoming pulses followed by the generation of signals for each detector indicating the identification of a coincidence involving it. The coincidence window is then equal to twice the length T_W of the timing pulses. Simple refinements of this architecture are possible. For instance, LOR validation may be hardwired for a fixed detector topology by removing the AND gates corresponding to invalid detector pairs. This may result in a dramatic reduction of logic complexity in cases such as dual-head PEM scanners [331]. Other improvements include detection and rejection of multiple events [332]. Gated coincidence detection may be realized using asynchronous circuitry [333] or digitally using synchronous pulses [331,334]. In the second case, the coincidence window is necessarily a multiple of the clock period T_{clk} and, in particular, cannot be too small or be programmed in very fine steps.

Timestamp-based schemes [335] increase circuit complexity by requiring the generation of a digital timestamp, either using a TDC or by digital processing of sampled timing pulses. Therefore, they result in increased cost, area and power consumption. However, these drawbacks are offset by the benefits listed below. Some of them correspond to generic early digitization approaches and are adaptations of those mentioned in §2.2.2:

- The single most important advantage of timestamping schemes is the improvement in time resolution, particularly in the granularity of the coincidence window. Implementing very small windows in a gated system proves problematic due to the need to generate narrow pulses whose active states need to overlap in time; this problem is exacerbated if the gating pulses are generated at the front-end and transported to the back-end for coincidence, as the transmission process degrades them due to noise and bandwidth limitations. Moreover, the implementation of finely programmable coincidence windows is straightforward in a timestamp-based approach.

- In relation to the last item, accurate measurements of time difference in gated systems are difficult and not easily scalable, as they need to provide a means to compare any two incoming pulses using e.g. a TDC. In particular, for ToF PET systems that require estimations of time differences with subnanosecond resolutions, it is mandatory to use timestamp-based schemes.

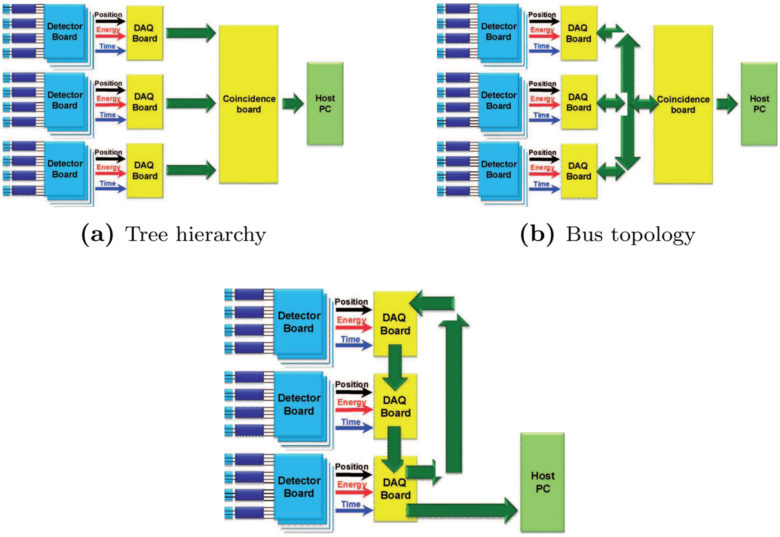
- Individual timestamping in each detector module offers a higher degree of flexibility in deciding how subevent data is handled at higher hierarchy levels. For instance, dynamic decisions to capture single or multiple events are easier to implement. One particularly important example is the capture of *delayed coincidences* [336]. This is a common method for the estimation of the rate of random coincidences between two given detectors whereby a coincidence window is established between them that is centered far away from zero, at a time difference value where no true coincidence is possible; therefore, any coincidences detected within that window correspond to random events. The acquisition of delayed coincidences is possible in gated systems but usually requires replication of the coincidence circuitry with added delay lines for any possible detector pair [331]; in a timestamp-based system, their implementation is much more simple.

- In §2.2.2, the convenience of triggerless readout schemes was established in the context of generic DAQ systems as technological advances make them possible. This argument also applies to PET systems. In this context, a triggerless readout necessarily involves timestamp-based coincidence detection, as the capture of subevents from different detectors is required to be completely independent.

One further difference between gated and timestamp-based coincidence detection is given by the synchronization between detector modules. In a gated approach, the difference between arrival times of timing pulses to a common location is considered, hence synchronization needs to be established at the physical level [334,337]. This corresponds to level 1 synchronization between detectors, in the terminology used in Fig. 3.10, and includes calibration of cable latencies and carries degradation of the coincidence resolution if they drift. For timestamp-based coincidence with early digitization, it is necessary to establish level 3 synchronization between detector modules, which is not necessary in a centralized gated engine. It was already established in chapter §3 that this is a more convenient and scalable approach, leads to improved compensation of delay drift, and allows the implementation of synchronization at the logical level, e.g. by LUT-based delay correction [332,338].

While all of the above considerations are focused on the DAQ triggering on events of multiplicity exactly equal to 2, it is possible to implement finer, more complicated trigger schemes that allow the recovery of some scattered events with higher multiplicity. Consider the following example: if two scintillations are detected almost simultaneously in different crystals of the same pixelated detector, then the segment joining the interaction locations lies entirely inside the scintillator and therefore cannot intersect the sample under study, so the event is necessarily multiple or scattered and would ordinarily be dropped as invalid. However, if the total amount of energy deposited by both of them equals 511 keV, then it can be assumed that both correspond to the same annihilation photon, which was first Compton scattered and later absorbed; it may then be possible to determine which of both local subevents corresponds to the first interaction and thus recover the LOR [339].

In any case, the trigger may be implemented in two levels [340], using L1 as a coarse filter with a large acceptance window based on gated coincidence detection using leading edge discriminators and later adding a second, fine coincidence engine based on more accurate timestamping methods. As with any other L1 trigger



(a) Tree hierarchy

(b) Bus topology

(c) Ring topology

Figure 5.1: Interconnection topologies for PET DAQ systems. Reproduced from [287], © 2013 IEEE.

scheme, subevents need to be either delayed analogically [341] or stored in queues at the subdetectors and the latency of the trigger signal generation needs to be controlled in order for these queues not to fill up. In the case of gated coincidence, the trigger logic is simple and latency can be kept very low [331], but in systems with timestamp-based L1 trigger e.g. those implementing distributed coincidence schemes, sufficient queue space is needed per node [342].

Communication architecture

In any PET DAQ, the coincidence detection subsystem is a middle layer between the detection of subevents at each detector module and image reconstruction. Subevent detection is a distributed task but the connection to the highest-level DAQ section where image reconstruction is performed is usually unique, hence there is a single top node in the hierarchy of coincidence boards. The communication architecture between elements of a PET scanner is strongly connected to the implementation of coincidence detection, as this is equivalent to the trigger subsystem and therefore controls the flow and amount of data. In [287], three different interconnection topologies are identified and analyzed in the context of communication bandwidth usage. They are depicted in Fig. 5.1 and described below:

- *Tree hierarchy* (Fig. 5.1a): This is the most common structure in PET systems as well as in generic HEP DAQs, and consists in all time detection modules sending their subevent data upwards to a centralized coincidence detection unit where events are identified. This is the only possible communication scheme in scanners with gated coincidence, as detector pulses need to be sent up to a common location. In timestamp-based systems, the simplest case corresponds to a single coincidence module where all timestamping nodes are directly connected [330]. This may not be possible in large scanners with a high number of timing modules, and intermediate hierarchy levels may be needed where data from lower level nodes are concentrated [343] and partial coincidence detection for their child nodes can be implemented in order to reduce their upstream bandwidth requirements.

  Event building is implemented at the top coincidence module, either in a single step for triggerless systems [335] or in two steps for systems with L1 trigger where timestamps are sent first for coincidence detection and other data such as position are retrieved later for actual coincidences only [213].

  In [287], this architecture is criticized as being extremely inefficient in its usage of communication bandwidth, since all subevent data need to be transmitted before singles are discarded. However, this is not necessarily true. For L1-type coincidence, only timestamps and source identifiers need to be sent to the coincidence engine, in particular they are transmitted only once if there is a single coincidence board; most communication architectures require timestamps from every subevent to be transmitted at least once so there is no loss. In the triggerless case, it is true that the bandwidth taken up by additional data from discarded singles is wasted; however, this increase in bandwidth was already established in §2.2.2 as a drawback of triggerless schemes in general that is nevertheless compensated by their benefits.

- *Bus topology* (Fig. 5.1b): In this case, the communication interface between the subevent timestamping nodes and the event builder is a shared transmission medium such as a data bus [344–346]. This allows a relatively cheap implementation of distributed coincidence detection with low latency, as timestamping nodes may broadcast the timestamps corresponding to each detected subevent so every module can detect the coincidences it is involved in, build the associated event and send it to the top node. The downsides of this approach are essentially the same of every bus-based system, namely its poor scalability due to increased arbitration overhead as the number of nodes grows. Another disadvantage is the fact that it is unable to benefit from most developments in faster communication technology, as they are focused on point-to-point links rather than shared buses, in line with the trends described in §2.2.1.

- *Ring topology* (Fig. 5.1c): In this situation, data acquisition nodes are connected following a circular ring topology, usually one-directional [347, 348]. Announcements of newly detected subevents are transmitted along the ring from one node to the next, where temporal coincidence with local subevents may be checked, until reaching the source node (or the one behind it) where they can be dropped. Such a scheme for coincidence detection does not need a top hierarchy node for coincidence, although one is necessary for communication with the image reconstruction subsystem anyway. Variations of this architecture have also been proposed such as toroidal topologies i.e. two-dimensional rings [349].

  This architecture requires a small amount of physical connection overhead (i.e. cables and boards), and it also adapts naturally to PET scanners since they are typically arranged as one or more rings of detectors. However, it also implies long coincidence detection latencies, as subevent announcements need to travel through the whole ring before they can be discarded as single events. Since this scheme is incompatible with triggerless readout and forces a L1 trigger, a suitably large subevent buffer queue is required per node. One further disadvantage is that subevent notices, including at least a timestamp, have to be retransmitted many times between consecutive nodes, so the aggregate bandwidth usage is much larger than in the other cases.

Some electronic board designs for PET DAQ support arrangement in several or all of the aforementioned communication topologies, such as [323].

The most important difference between the possible DAQ topologies are given by bandwidth usage and coincidence detection latency, which is highest for ring topologies and lowest for hierarchic trees. Regarding system scalability, it is poorest for bus topologies and highest for tree-based systems, provided that coincidence units are designed in such a way that several layers of them may be implemented, so that the number of detectors is not constrained by the number of connections in a coincidence board. While ring topologies may physically accept an arbitrary number of detectors, their coincidence detection latency increases linearly with it (rather than logarithmically as in tree topologies) so their scalability is compromised.

In all connections, it is necessary to guarantee enough throughput capability to support the expected data rate, determined by either the subevent rate in links between detector modules and coincidence units, or by the valid event rate in the uplink connection from the top node to the external processor that handles image reconstruction. As has been mentioned, early online processing reduces the necessary bandwidth by producing compressed descriptions of subevents and events instead of relaying raw data for later processing.

PET project reference	Front-end description				DAQ description					Resolution		Comments
	Crystal	DOI estimation	Detector	Multiplexing scheme	Readout scheme	Coincidence detection	Sync scheme	Timing method	Sampling	Coinc. (ns)	Energy (%)	
Small animal PET												
microPET [350,351]	LSO	No	PSPMT	Anger logic	Triggerless	Timestamp		CFD+TDC	FRS	3.2	23	
ClearPET [325,352]	LSO+LuYAP	Phoswich	PSPMT	Binary encoder	Triggerless	Software	Clock tree	Software	FRS	2.0	15–23	
MADPET-II [353,354]	LSO	Dual layer	APD	None	Triggerless	Software	Clock tree	CFD+TDC	Peak detector	4.5–10.2	20.7	
RatCAP [355]	LSO	No	APD	None	Triggerless	Timestamp		Shaping+TDC	TDC	6.7	18.7	
LabPET [356]	LYSO+LGSO	Phoswich	APD	None	Triggerless	Timestamp	Backplane	DCFD	FRS	5.4–9.8	25	
LabPET II [357]	LYSO+LGSO	Phoswich	APD	None	Triggerless	Timestamp	Backplane	LED+TDC	ToT			
BNL/Penn [217]	LYSO	No	APD	None	Triggerless	Timestamp	Reset tree	Shaping+TDC	TDC	8.7	30	MR compatible
miniPET-III [324,358]	LYSO	None	SiPM	Row-column	Triggerless	Software	Clock tree	DCFD	FRS	2.0		MR compatible
Trans-PET [359]	LYSO	None	PSPMT	Anger logic	Triggerless	Software	Clock tree	MVT+TDC	FRS	1.4	14.2	
cMiCE [323,360]	LYSO (cont.)	SBP	PSPMT	None	L1+L2	Gated+TS	Clock tree	Fitting	FRS	1.05	18.4	
dMiCE [323,361]	LYSO	Light sharing	SiPM	Row-column	L1+L2	Gated+TS	Clock tree	Fitting	FRS		12.3	
Fysikopoulos et al. [362]	BGO	None	PSPMT	Anger logic	L1+L2	Gated+TS	Single PCB	DCFD	FRS	17.2	18.2	Dual-head
Brain PET												
Kim et al. [363]		None	SiPM	Compr. sensing	Triggerless	Timestamp	Data link	Digital	FRS		16.2	MR compatible
Omura et al. [343,364]	LYSO	Four layers	SiPM	Anger logic	Triggerless	Software		LED+TDC		2.3	24.5	
Others, or adaptable geometry												
Sportelli et al. [330]	LYSO	None	PSPMT	Row-column	L1	Gated	Asynchronous	CFD	Peak detector	9.1	20	PEM
PhytoPET [365]	LYSO	None	PSPMT	Anger logic	L1	Timestamp		Digital	FRS			Plant, flex. geometry
LaPET [213]	LaBr$_3$	None	PMT	Optical	L1+L2	Timestamp	Clock tree	LED+DCFD	Analog memory	0.37–0.42	5.3	Whole body, ToF
SPADnet [220,342,366]	LYSO	None	dSiPM	None	L1	Timestamp	Clock tree	TDC	Counter	0.4	10.9	ToF, MR compatible
OpenPET [367]	LSO	None	PMT	Optical Anger	Triggerless	Timestamp	Backplane	LED+TDC	FRS	0.26	12	Also SiPM
Developed at EDNA												
Martínez et al. [189,329]	LSO (cont.)	Distr. width	PSPMT	Anger logic	Triggerless	Timestamp	Single PCB	DCFD	FRS	5.1	20	PEM
Esteve et al. [1]	LSO (cont.)	Distr. width	PSPMT	Anger logic	Triggerless	Timestamp	Backplane	DCFD	FRS	5.3	19	Brain PET
This work [8]	LSO (cont.)	Distr. width	PSPMT	AMIC	Triggerless	Timestamp	Data link	DCFD	FRS	2.0	19	Also SiPM

Table 5.1: Summary of existing PET systems and their main features with respect to the items discussed in this dissertation.

5.1.3 Review of PET systems

The first PET scanner, from 1972 [176], featured a ring of 32 large NaI crystals with energy windowing, gated coincidence detection and a minicomputer reading LORs and implementing a very simple reconstruction method. From that moment, the evolution of PET DAQ systems is parallel to that of generic DAQ systems as presented in §2.2.1. The technological advances that had the most impact were those related to increase in digital processing capability (particularly in microcomputers for image reconstruction) and programmable logic for the implementation of fast custom logic circuitry. In particular, the issue of parallel versus serial communication was not very important with long scintillator decay times and the prevalence of gated coincidence detection. The most important advances specific to PET are the advent of the block detector in 1986 [224] where optical multiplexing and the possibility of small crystals were introduced, the development of fast scintillators like LSO in the 1990s [183], and later the resurgence of ToF PET in the wake of the 21st century [261].

The current trends in PET DAQ are mostly the same as in HEP DAQ:

- Digitization of detector signals as early as possible [357, 359, 366].
- Shift to triggerless acquisition [335, 354, 367].
- Increasingly complex local digital processing in FPGA [323, 327, 356] or, lately, integrated with the photodetector [220, 368].
- Attempts at modular, generic, scalable electronic design [338, 358, 366, 367].

Availability is not a very important concern in the case of PET since the electronic assembly is accessible and individual scans rarely take more than half an hour.

The current state of the art in data acquisition for PET is outlined in Table 5.1, where some of the most important or innovative PET systems are described. In particular, details are given on the arrangement of their front-end readout circuitry, trigger subsystem, subevent building methods, and synchronization scheme for their timing nodes. A few items were left blank because appropriate references could not be found. Several systems such as NanoPET/CT [369], eXplore VISTA [245] or HOTPET [332] were excluded from the table altogether for the same reason. Resolution values are included for energy and timing but not for position, since most examples correspond to pixelated detectors and therefore their position resolution is fundamentally determined by the pixel size; the few cases that involve continuous crystals are labeled explicitly.

It is noteworthy that most of the systems correspond to small animal imaging; since they are typically smaller-scope systems than full-body human PET scanners, and used primarily for biological research, a large part of the innovation efforts in

PET instrumentation is devoted to them. Some system descriptions correspond to variable configurations, such as OpenPET and SPADnet. Others support flexible geometries independently of their intended application, such as PhytoPET and the architecture presented in this thesis.

In the “synchronization scheme” column of Table 5.1, a distinction is made between system-synchronous clock trees that are implemented through controlled PCB traces and backplane connections (*backplane*) and those reliant on matched-length cabling (*clock tree*). Unfortunately, synchronization accuracy is not usually reported except for very specific cases such as [281, 350, 351, 370]; in those cases, it remains below 500 ps. However, for ToF PET these figures are not restrictive enough.

5.2 Proposed architecture

The DAQ architecture for PET scanners that is proposed as the object of this thesis can now be described. The following paragraphs are dedicated to summarizing all observations made previously throughout this dissertation and deriving the architectural proposal from them.

It has been established in chapters §2 and §5 that early digitization is a desirable feature in PET DAQ systems, in order to limit the loss of quality in detector signals that lead to reduced energy and time resolution due to added noise and bandwidth reduction, respectively. In particular, a triggerless scheme is the optimal option provided that the output bandwidth of each detector node can support it. This can be achieved by integrating early digital processing stages that implement all feature extraction algorithms online and result in completely summarized, digital subevent descriptions that are as short as possible.

The ideal DAQ therefore consists of completely independent detector modules with a single digital output each that contain a stream of detected subevents. Advances in FPGAs already make this a viable choice with small circuit footprints. An even stronger integration effort has been launched with the advent of digital SiPMs, where parts of the digital feature extraction circuitry are embedded in the same silicon die as the photodetectors. It is not unreasonable to expect fully integrated *PET-on-Chip* concepts to appear in the near future where the entirety of the front-end electronics is condensed into a monolithic ASIC that collects scintillation light and outputs digital frames containing subevent descriptions.

The superiority of timestamp-based coincidence detection over gated implementations has also been established in §5.1.2. This, together with the considerations above, implies the necessity of level 3 synchronization between detector modules, using the terminology from Fig. 3.10, i.e. the local timestamping logic in each module needs to share a common time reference. The renewed interest in ToF

PET, where coincidence resolutions below 500 ps are expected, forces in turn synchronization accuracies in the order of 100 ps. It is thus mandatory to establish a precise synchronization scheme between all detector modules in the PET system.

Optimal synchronization accuracy is obtained when no cable or fiber is involved, and transmission media consist exclusively in controlled PCB traces and connectors; in this case, that goal is achieved when there is a motherboard or backplane with slots where all detector modules are inserted. Examples of this approach are given by [217, 357]. However, resorting to backplanes and eliminating cables completely carries the following limitations to the PET system:

- Scalability and system expandability are hindered by the use of backplanes with mechanical slots for detector module insertion, as the number of detectors becomes limited by the amount of connection slots. Adding spare slots merely pushes the limit upward, but hardware redesign is needed whenever the new limit is to be surpassed.

- The modularity and potential for reuse of modules is also limited by mechanical constraints such as form factors, board-to-board connector specifications, etc.

- Variability in scanner geometry or topology are constrained by the size and form of backplanes. For example, scanner designs with a fixed number of detectors that allow a certain degree of mobility such as [365,371] are useful for certain medical applications [372]. Similarly, insert [373] or camera modules [372] cannot be added to a given scanner setup using a fixed frame and cabling is required instead. The same is often the case for the connection between subsystems in a dual modality scanner [374].

These disadvantages may not be really very significant in the case of certain PET scanner designs intended for mass production, but they definitely are for small research setups which are expected to evolve over their life time.

Precise synchronization of detector modules over cables is thus a better choice. There are two possible approaches to this: either using a cable-based clock tree, or implementing a pairwise synchronization scheme over the data link. Cable-based clock trees are employed usually, as exemplified by Table 5.1. In this dissertation, however, a case is made against them and in favor of pairwise synchronization over the data link. Some of the reasons have already been presented in chapter §3, namely: self-calibration capability, compensation of drifts in delay, and adaptability to arbitrary cabling choices. An additional advantage can be identified in PET scanners in that the reduction in the number of cables notably simplifies the gantry assembly, especially for rotating gantries [340]. By integrating the clock distribution with the digital data link, detector modules can be defined whose only connections are two cables: one for the data link and another one for the power supply.

In light of these considerations, a PET DAQ architecture is proposed whose elements are outlined in Fig. 5.2. The system is divided logically into a front-end section and a back-end section, and each of them is formed by an arbitrary number of respectively identical modules with no constraints on physical location with respect to each other, i.e. the only placement restriction is given by the particle detectors at the front-end. Two kinds of modules are defined that completely make up the DAQ circuitry:

- *Acquisition modules:* These are the detector modules located at the front-end section. They contain the scintillators and photodetectors and perform analog conditioning of detector signals, digitization, subevent detection and digital extraction of its features: timestamp, energy and position.
- *Concentrator modules:* These modules conform the system back-end and are responsible for the collection and aggregation of subevent data from acquisition modules or other concentrator modules, detection of coincidences, event building, and communication with the external processor that handles the image reconstruction.

For the communication between modules, an Inter-Module Link (IML) is defined. This is a purely digital, full-duplex data link based on self-synchronous signaling with embedded clock and the capability of precise synchronization and self-calibration. It corresponds to the link and synchronization scheme proposed in §3.5. In particular, for every IML, its ends are regarded as master or slave according to the global system hierarchy. The links serve three purposes:

- *Data transmission:* Subevent and possibly event data are transmitted upward (from slave to master) and configuration commands are sent downward (from master to slave).
- *Syntonization:* The slave is responsible for recovering the clock frequency from the data link and uses it for its own downlink transmissions, and its digitizing circuitry in the case of acquisition modules. Thus, the whole system becomes syntonized with the master oscillator which is located at the top concentrator module.
- *Synchronization:* In each link, the slave node is capable of synchronizing its time reference with that of the master using the pairwise synchronization algorithm described in chapter §3, and does so independently of any other module connections. Clock phase alignment between the end nodes is not necessary.

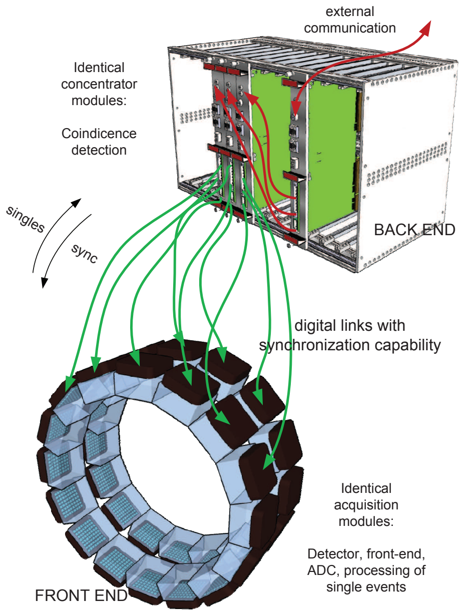


Figure 5.2: Logical architecture of the DAQ system. Arrows represent digital links: cables between back-end and front-end, and either cables or backplane connections between back-end modules. Frequency and synchronization are propagated in the direction of the arrows.

The only connection present in acquisition modules is an IML implemented on a single composite cable, where the acquisition module acts as the slave. Concentrator modules host a number of IMLs to lower level nodes, which may be acquisition modules or other concentrator modules, and act as masters over them; they also contain an upward IML where they function as slaves to be synchronized with a higher level node, and a network interface to an external processor using a standard protocol such as Gigabit Ethernet. IMLs at the back-end may be implemented using either cables or backplane connections.

This architecture is arbitrarily scalable: no hard limit is imposed on the number of detectors or modules, not just at the design stage but also after the modules have actually been built. Several complete sub-DAQs may be merged into a single one that retains acceptable coincidence resolution just by connecting their two top nodes to a new concentrator module. Synchronization between dual-modality subsystems or between the main ring scanner and an insert module is therefore not hindered.

Expandability and compatibility with other subsystems is thus guaranteed. Since all modules are identical copies of one of two different hardware designs and physically independent of each other, all PCBs are fully reusable in the case of system expansion or any other topology change. Moreover, detector modules are connected with minimal cabling (a data link and a power connection) so they are almost completely mobile and variable geometries may be implemented.

Although there are no hard limits on system size, the existence of soft limits needs to be acknowledged. There are two main parameters that get degraded as the DAQ grows. The most obvious one is the bandwidth of communication links, which has to be large enough to support the transmission of all subevent or coincidence data at the highest levels; if it is not, subevents start to get lost to full queues and the accepted event rate suffers. A number of recent studies on the scalability of PET DAQ architectures [287, 348, 349, 375] focus on this limitation as they identify coincidence management as the bottleneck to DAQ scalability and propose distributed coincidence schemes based on ring topologies and variations thereof as the solution, suggesting that tree topologies are too inefficient in terms of link bandwidth usage.

However, synchronization is another soft limit that needs to be taken into account. The synchronization accuracy of the system will invariably decrease as the height of the DAQ hierarchy increases and the number of intermediate nodes increases together with the aggregated propagation delay variance in the Lundelius-Lynch bound (3.14). For systems with very stringent time resolution requirements such as ToF PET, this issue emerges as the main limitation to scalability. In fact, for a clocking scheme such as the one proposed here, ring topologies yield no advantage: independently of the choice of data communication architecture, a substructure for syntonization and synchronization needs to be established anyway, and this structure achieves optimal resolution when its height is *minimized* i.e. when it corresponds to a tree topology, as opposed to a ring topology where the height is *maximized*. This does not prevent the use of additional links between concentrator modules for increased coincidence detection performance, but such a study remains outside the scope of this thesis.

Chapter 6

Implementation of the proposed DAQ architecture

The previous chapters have described and analyzed the structure and desirable features of a scalable, mobile architecture for a PET data acquisition system, culminating in a proposal for a DAQ architecture in chapter §5. In this chapter, a particular implementation of the proposed architecture is described, as well as its design process. It corresponds to the contents of paper [8] but with many more details and results.

The implementation presented here has a very specific target environment, namely the two-detector experimental setup located at the EDNA facilities. Consequently, only a partial implementation of the proposed architecture is needed. In particular, full-fledged concentrator modules are not necessary. Combined acquisition-concentrator module boards have been developed instead capable of performing both functions, and the DAQ system consists of two such modules, one of them functioning as a pure acquisition module and the other one carrying out the acquisition role and also acting as the top level concentrator, processing coincidences and communicating with a readout PC. Nevertheless, it will be seen that these boards may be reused as acquisition modules in a larger-scale DAQ system by developing additional concentrator modules with an appropriate number of downlinks.

6.1 Circuit board design

As stated above, the purpose of the work described in this chapter is to develop and evaluate a single circuit board that is able to act either as an acquisition module or as a mixed acquisition-concentrator module for a two-detector scanner. More specifically, the objectives to be fulfilled are the following:

- The modules have to function as a direct replacement of the previous generation DAQ [1], accepting analog input signals from the AMIC-based front-end boards and communicating with an external PC through a standard interface.
- They also need to provide the means to validate the proposed PET DAQ architecture. In particular, it must be possible to measure the synchronization accuracy properly, and to test the stability and performance of the clock distribution and synchronization scheme, at the very least for a two-level hierarchy i.e. a single concentrator node with acquisition leaves hanging directly from it.
- Finally, the mid-term research goals at EDNA have to be taken into account, as the board has to be able to accommodate any foreseen experiments. In particular, fully digital feature extraction is to be implemented based on free-running sampling ADCs, increasing the sampling frequency over 100 MHz, in order to support ongoing research on online positioning and timing algorithms.

This section focuses on the translation of these general goals into definite circuit specifications and on the design of the circuit board, both at the schematic and the physical level.

6.1.1 Board specifications

The specifications and main design choices for the board are detailed in this section, grouped by generic board function.

Interface with the detectors

Each board has to provide acquisition capabilities for a single block detector. In particular, the analog interface is defined by the AMIC front-end test boards that were developed previously. The AMIC devices have already been presented in §4.3.2. Each of them is used to convert 64 photodetector output currents into up to 8 current signals that retain the subevent position and energy information. Moreover, both PMT and SiPM are accepted as photodetectors.

Two different variations of AMIC boards are available. One of them contains a single AMIC device and can handle detectors with 64 outputs. The other one can process the signals from a detector with 256 outputs by using four AMICs and wiring the corresponding current output signals together in order to implement their addition [259]. In both cases, the output signals are converted to voltage on-board using commercial operational amplifiers. The resulting signals have a bandwidth between 25 MHz and 50 MHz and a dynamic range around 1 V peak-to-peak.

In order to contribute to the general theme of modularity, the AMIC boards feature connectors for two mezzanine boards:

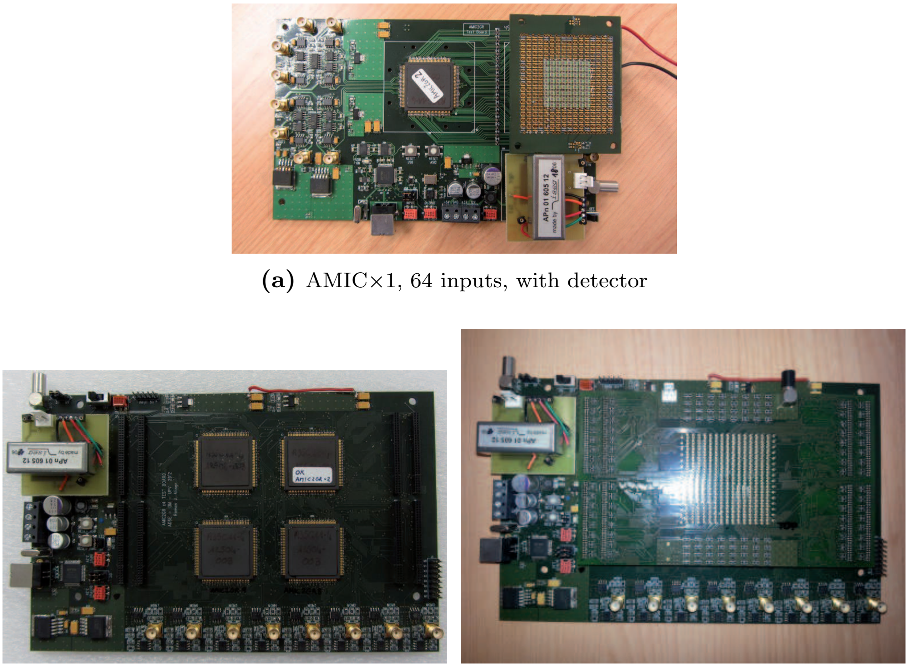
- One for the photodetectors, with either 64 or 256 signals. For the 64-channel version, the connection interface matches that of the Hamamatsu H8500 PMT [376] so that it can be connected directly; a 64-SiPM mezzanine has also been designed that is adapted to the same interface. For the 256-channel version, a custom interface has been used and only the SiPM version has been assembled.
- A high-voltage source module that provides the bias voltage for the photodetectors. Two slightly different module designs exist, containing a −1500 V and a −100 V voltage source for PMTs and SiPMs, respectively. The connection interface is the same for both and allows fine tuning of the bias voltage through a digital interface.

The whole set of front-end boards is pictured in Fig. 6.1.

A common interface with the data acquisition system is defined for all variations of the AMIC submodule, given by the following connections:

- Eight AMIC outputs, converted to voltage, from SMA connectors adapted to 50 Ω.
- A ninth output, connected directly to the last PMT dynode (unused for SiPM configurations), using the same connector type.
- An I^{2}C interface bus for the configuration of AMIC devices, high-voltage sources for the photodetectors, and other board functions, using a MicroMatch connector.

Hence, the acquisition module needs to provide at least nine ADC channels with appropriate shaping circuitry and input using a SMA connector. Eight of them have to be identical and intended for energy signals, while the ninth one has to include specific shaping for AC-coupled timing signals. A MicroMatch connector routed to the control FPGA is also needed for the I^{2}C configuration interface.



(a) AMIC×1, 64 inputs, with detector

(b) AMIC×4, 256 inputs, without detector

(c) AMIC×4, 256 inputs, with detector

Figure 6.1: AMIC front-end boards used to interface the photodetectors and the acquisition modules. The boards include a small mezzanine with the high-voltage source, and another one with the appropriate number of SiPM.

Digitization

Selection of the ADC devices for the acquisition board is severely constrained by the sampling frequency, which needs to be higher than $f_s = 100\,\text{MHz}$ while retaining at least a 12 bit resolution for compatibility with previous setups. Since at least 9 digitization channels need to be implemented, it is impractical to employ ADC devices with parallel bus readout i.e. using a dedicated wire for each bit of the output sample with a common f_s clock; the reason is that the 108 resulting output signals require a large amount of dedicated routing area, with skew issues likely to arise due to unavoidable trace length mismatches [377], and signal integrity issues caused by Simultaneous Switching Noise (SSN) on a high number of digital traces ([378], pp. 391–403).

As an alternative, ADCs with serial outputs need to be considered. In particular, ADCs with serial source-synchronous interface are a popular option for f_s in the 10 MHz to 100 MHz range [379–382]. For higher frequencies, however, the limitations of source-synchronous signaling described in §3.1.1 arise as the bit

	AD9239 [112]	LTC2274 [384]
Resolution	12 bits	16 bits
Maximum sampling rate	170 MHz to 250 MHz	105 MHz
Channels per device	4	1
Size per device	10 mm × 10 mm	6 mm × 6 mm
Power consumption per channel	285 mW	1300 mW
Physical level coding	Scrambling	8B/10B

Table 6.1: Comparison between the AD9239 and LTC2274 ADCs.

clock frequency exceeds 1 GHz and one must resort to either multilane source-synchronous buses [383] or self-synchronous interfaces, which necessarily provide some serial clock recovery mechanism that must be generated at the source i.e. the ADC. At the time of design of the acquisition board this was still a rare feature for commercial ADCs in the intended frequency range, and only two devices were found that satisfied all requirements: the LTC2274 by Linear Technologies [384] and the AD9239 by Analog Devices [112].

Table 6.1 compares the most significant differences between both device choices. Although the LTC2274 was initially considered and tested, the AD9239 was selected in the end due to its lower power consumption and PCB footprint for the intended number of channels, as well as its higher sampling frequency; its lower resolution was not considered a problem as it is still enough for the application. The only possible drawback is the scheme for clock recovery, as it relies on scrambling rather than the more reliable 8B/10B encoding, but this has not been found to be an issue in subsequent tests. All in all, 3 AD9239 devices are necessary in order to cover all 9 input channels.

Level 2 synchronization

In section §3.1.3, the latency between the ADC and the FPGA was identified as one of the synchronization levels that needs to be considered during design for precise timing. In this case, communication between both devices takes place over a self-synchronous link, implying the use of complex transmitters and receivers at the ADC and FPGA, respectively, that involve framing and clock embedding and extraction operations with additional latency components. These components are not necessarily fixed or even deterministic.

The problem of non-deterministic ADC to FPGA latency has been recognized since the mid-2000s. For instance, [377] describes a way to equalize the receiver latency for multichannel ADCs in a single FPGA but the resulting uncertainty is in the order of one sampling clock cycle. It is also possible to employ transceivers with deterministic latency configurations, but the total latency, given by (3.33), necessarily includes a phase term in a multichannel situation since all sample streams need to be converted to a common local clock domain at some point, so phase measurements and their associated errors need to be involved. Even in this case, the transmitter i.e. ADC latency is not guaranteed to be deterministic; the recent JESD204B standard for ADC output links acknowledges and attempts to solve this problem [385].[1]

For the ADC to FPGA latency to be completely neglected, it is necessary to equalize all latency terms not just across a single board but rather across *all* acquisition nodes. Alternatively, equalization may be limited to a board-per-board basis if an additional latency measurement step is carried out in order to compensate for differences between acquisition modules. The latter approach is the one followed in this design, and corresponds to individual level 2 synchronization in each acquisition node using the terminology of §3.1.3.

The proposed latency compensation scheme is outlined in Fig. 6.2 and takes advantage of the fact that multichannel ADCs are being employed. An analog reference signal generator is included on board, whose start can be triggered by FPGA logic, and one of the ADC channels is dedicated to sampling it. The time delay values t_{FG}, t_{GA} and t_{AF} are shown, corresponding to the latency from the FPGA latency estimation logic to the reference signal generator, to the ADC input, and from there to the latency estimation logic in the FPGA again, respectively.

It is assumed that the receiver logic includes a method for alignment of all receiver channels so that the ADC to FPGA latency is identical for all of them.[2] The procedure begins after alignment has been completed, so that t_{AF} is equal for all ADC channels. The reference signal is triggered and its start time is recovered from the read samples. In this way, the time difference between the start of the signal and its detection can be measured, i.e. $t_{\mathrm{FG}} + t_{\mathrm{GA}} + t_{\mathrm{AF}}$. It is assumed that the components t_{FG} and t_{GA} are constant between identical circuit boards. The measurement is thus of the form constant $+\,t_{\mathrm{AF}}$ where t_{AF} corresponds to the ADC channels with detector signals. By subtracting this value from timestamps assigned in this node, synchronization at the ADC level is achieved.

A few comments on the proposed method are in order:

- The reference signal needs to have an easily recognizable landmark indicating its digitally triggered start so that it can be recovered online with sub-clock cycle resolution. A simple analog linear ramp generator is proposed, as the start time can be computed by fitting a line to the recovered samples and finding its intersection with the baseline. The circuit will be described in §6.1.2 and the latency estimation logic in §6.2.4.

[1] The issue had been brought up at [5] and the boards had been designed before the appearance of the standard in 2011.

[2] The chosen method for ADC channel alignment will be described in §6.2.1. Other options such as [377] are possible.

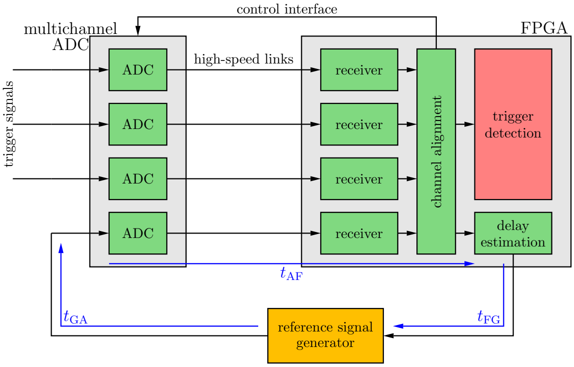


Figure 6.2: Latency measurement method for level 2 synchronization in acquisition boards. The ADC control interface is used by the channel alignment procedure.

- The latency estimation will invariably have errors due to noise in the measurement process. The procedure thus needs to be repeated periodically in order to obtain an averaged value.

- A reasonable assumption is that $t_{\text{FG}} + t_{\text{GA}}$ are not equal in all boards, but remain constant within a given board. This results in a fixed time offset that can be compensated with a one-time calibration.

- It is reasonable to assume that the alignment process results in identical t_{AF} for all channels within the same ADC device, but it is unclear whether this necessarily holds across all multichannel ADCs in the same board, due to possible differences in the ADCs' t_{TX} term or in the phases of the clock signals entering the ADCs. It was thus decided to locate the reference signal channel in the same ADC as the signal intended for time pick-off i.e. the dynode.

Communication interfaces

Since the board is to be used as an acquisition module, it must contain at least one IML as defined in §5.2, for communication and synchronization with a parent concentrator module. A single IML is also sufficient for the mixed acquisition-concentration module in a two-detector setup, as it only needs to connect with a single acquisition slave. Nevertheless, the decision was made to include two IMLs in the board. This allows for a test of its functionality as a concentrator-only module by using three boards: two for acquisition and one only for concentration, with an IML towards each of the other ones. More importantly, this is the minimum configuration that allows for the arrangement of an arbitrarily large DAQ using only instances of the designed board, as many as detectors in the system, by

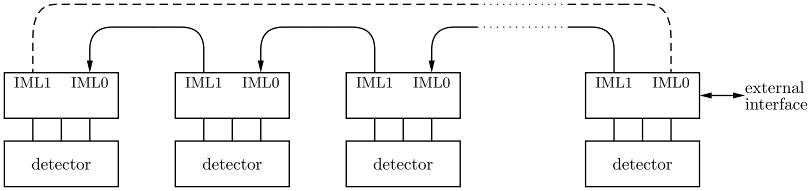


Figure 6.3: Suboptimal DAQ architecture using mixed acquisition-concentration modules with two IMLs each. Arrows indicate the direction of frequency propagation. The dashed link is optional and distinguishes between a tree and a ring topology.

daisy-chaining them as outlined in Fig. 6.3 and forming either a long tree or a ring topology. Of course, this configuration results in the worst possible synchronization performance compared to one that uses intermediate concentrators, because the synchronization tree height and thus the synchronization error are maximized.

Additionally, an external interface needs to be implemented as part of the top concentrator functionality, in order to communicate with an external PC for slow control and coincidence result transmission. A Gigabit Ethernet link was selected for this task due to its simplicity and widespread support. More specifically, the decision was made to use a pluggable SFP transceiver and install sockets on all boards; in this way, a single SFP module is needed for the DAQ system and any board can be used and tested as a concentrator by plugging the SFP into it.

Finally, it was decided to include two small expansion headers connected to the FPGA in order to support extra digital communication lines or the addition of unforeseen features in small mezzanine boards.

Clocking scheme

The main clocking subsystem for the acquisition board encompasses the generation and distribution of the local clock that is to be used for local FPGA logic, IML transmission, and signal digitization. Considering the ADC specifications on maximum sampling rate, the value $f_{\mathrm{clk}} = 156.25\,\mathrm{MHz}$ has been chosen as the base frequency for the whole system due to the high availability of compatible commercial components.[3]

Recall from §3.5.1 that the main local clock in a slave node such as an acquisition module has to be derived but different the recovered IML clock. Moreover, at least six copies of the main local clock need to be generated: one for each of the three ADCs, two for the FPGA, and one final clock signal routed directly to connectors,

[3] 156.25 MHz oscillators are employed extensively e.g. in 10 Gigabit Ethernet applications as they match the parallel clock frequency of the $m = 64$ bit wide words in single data rate implementations [386].

intended for test purposes or additional functionality; two copies are required for the FPGA in order to cover the transceiver reference clock inputs in both sides of the device due to its internal architecture. The clocking scheme described in §3.5.1 cannot be used directly in this case due to the need for clock replication. However, the lifting of the constraints imposed by the evaluation board allow a superior variant to be implemented where the all PLL components are integrated in a single, external device with a linear phase detector for vastly reduced jitter.

The resulting circuit is outlined in Fig. 6.4 and is based around a National Semiconductor LMK02000 PLL and clock distribution device [387] that generates up to 8 clock copies. Its crucial feature is that the VCO is external to the device and its input i.e. connection to the phase detector output can be switched open (tristate output) or closed on demand. The same 156.25 MHz VCXO is employed and the clocking scheme is modified in the following way: at the top node, the switch is kept open so that the VCXO control input stays at a constant bias value and the circuit works as a fixed oscillator and clock distributor, feeding the transceiver's reference clock input. At slave nodes, the PLL is initially configured in the same way; however, once the recovered clock is stable, the loop switch is closed and the VCXO output eventually converges to a jitter-filtered copy of the recovered clock. Once again, transceiver operation and clock recovery are not affected during the PLL transient period because its reference clock suffers only very small variations while maintaining its nominal value; in particular, clock phase remains continuous.

The values of the components in the external PLL loop filter were determined with help of the National Semiconductor Clock Design Tool. The phase noise profile for the VCXO was obtained from its datasheet; for the recovered clock, the measurements reported in [95] were used as an estimation. The estimated random jitter with this configuration is around 800 fs rms.

Besides the ADC and IML clock, two additional oscillators are needed on board: a 125 MHz clock for the transceiver implementing the Gigabit Ethernet interface, and a simple oscillator for the local control clock. A 10 MHz value was selected for the latter in order to generate 40 MHz and 100 MHz control clocks inside the FPGA. Note that all of these oscillators need to be independent in order to ensure reliable slow control operation in case of link failure, and to support pure acquisition boards that lack the Gigabit Ethernet circuitry.

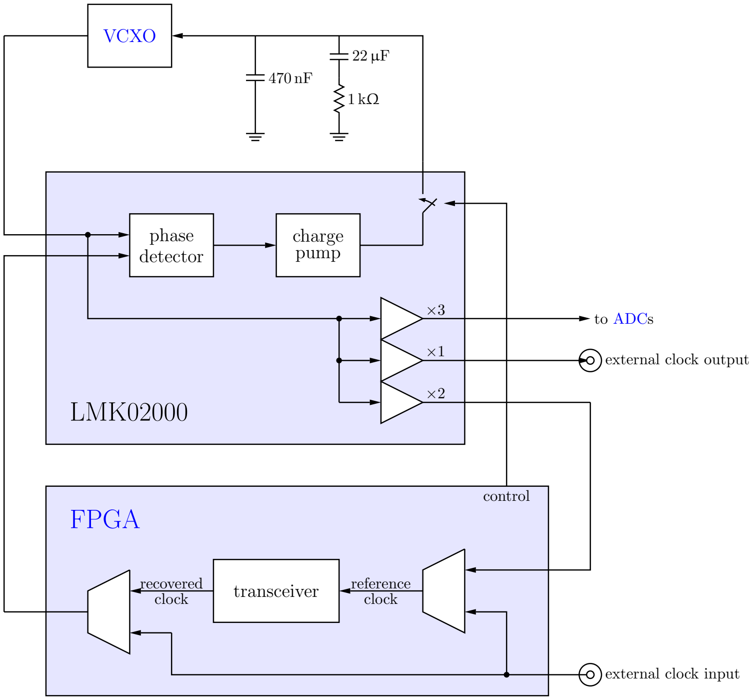


Figure 6.4: Schematic of the clock recovery and distribution subsystem. The PLL loop starts open and gets closed after the recovered clock from the IML becomes available. FPGA configuration allows the use of an external clock as either the transceiver reference or the local logic and sampling clock.

FPGA selection

There are three main constraints for the selection of the FPGA:[4]

- The included transceivers have to support deterministic latency configurations in order to implement the IMLs as described in §5.2.
- At least 13 transceivers are needed: 10 for ADC channels (including 8 AMIC outputs, one dynode and one reference signal), 2 for both IMLs, and one for the Gigabit Ethernet SFP interface.
- It needs to contain enough internal resources (logic elements, embedded memory, PLLs...) for the intended firmware.

[4] The use of a single FPGA is implicitly assumed. In fact, designs with two FPGAs were also considered, and the cheapest option actually consisted in the combination of a large Arria device and a small Virtex-5 for the IMLs. This choice was finally discarded.

	XC5VLX110T [388]	EP4SGX110F [389]	EP2AGX190F [390]
Transceivers	16	16	16
PLLs	6 PLLs, 12 DLLs	4 PLLs	6 PLLs
Registers	69120	84480	152240
Embedded memory	6.5 Mbit	9.6 Mbit	9.9 Mbit
Footprint	$35 \times 35\,\mathrm{mm}^2$, 1136 pin	$35 \times 35\,\mathrm{mm}^2$, 1152 pin	$35 \times 35\,\mathrm{mm}^2$, 1152 pin
Price	1823 $	2609 $	1400 $

Table 6.2: Comparison between the smallest eligible Virtex-5, Stratix IV and Arria II FPGAs. Prices were retrieved from www.digikey.com in August 2015.

By the analysis in §3.4.1, the first constraint implies that the FPGA family must be either a Virtex-5 LXT or better, or a Stratix IV GX (or Arria II GX) or better, depending on whether Xilinx or Altera devices are used. The other constraints merely indicate which of the devices in the given family can be used. The smallest, cheapest devices that can be used are the XC5VLX110T, EP4SGX110F and EP2AGX190F, respectively, and their main features are compared in Table 6.2. The devices are roughly equivalent, except for the Arria II, which has increased logic capacity in exchange for worse speed grade.

In terms of synchronization performance, it was shown in §3.4.1 that the synchronization error is expected to be 40 % higher in Altera devices, suggesting that the optimal choice would be the Virtex-5. Nevertheless, the Stratix IV solution was adopted in the end, thanks to a generous donation of FPGA devices by the Altera University Program.

Circuit block diagram

The previous considerations are enough to completely define the specifications of the circuit board. Its resulting block diagram is shown in Fig. 6.5 and includes all items mentioned previously: the 9 analog inputs, three ADC devices, the FPGAs, clocking circuitry and clock and data connectors. A few additional items are included:

- A 8 MB RAM chip. While the internal memory is enough to implement the high speed data path corresponding to detector data, additional memory is needed for the operating system software, particularly the management of the Gigabit Ethernet protocol stack.
- A 8 MB flash memory chip meant to hold the FPGA configuration file. This leaves approximately 1 MB unused non-volatile memory that can be potentially used to store detector configuration data such as calibration coefficients.

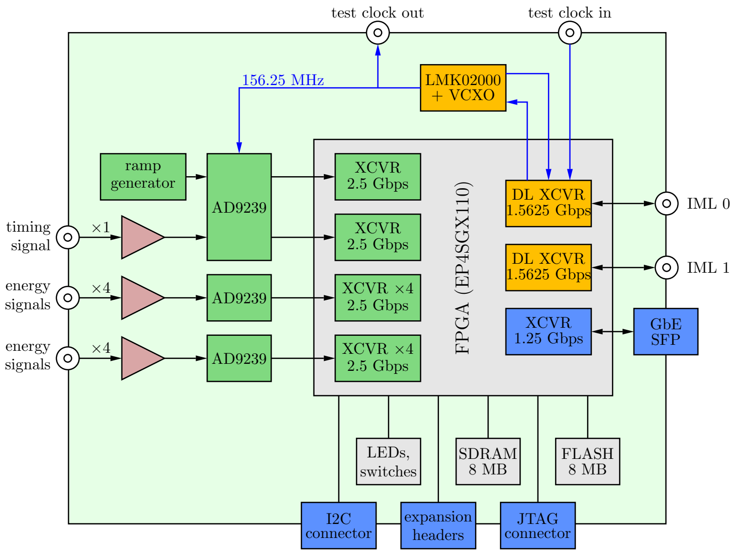


Figure 6.5: Block diagram of the contents of the circuit board, indicating the FPGA transceivers with data rates and the clock distribution path.

- A JTAG connector used to program and debug the FPGA.
- A number of LEDs and switches for status display and debug purposes.

This hardware board contains enough hardware to implement all foreseen module functionalities by using the appropriate FPGA firmware. Figure 6.6 shows the complete schematic of the pure acquisition module and the mixed acquisition and concentration module that will be used for the evaluation of the proposed architecture. Note that the modules include both the board under discussion and the AMIC front-end boards, connected with cables (9 for the detector signals plus one for the I^{2}C control bus). This is suboptimal but necessary in this prototype version for compatibility with the existing setup; a final version would integrate the whole module, including the AMIC and related circuitry and the acquisition electronics, on the same circuit board.

The main functional blocks inside the FPGA are depicted in Fig. 6.6, using a different color for each of the main FPGA functions for clarity: sampled signal reception and delay compensation, subevent detection and processing, synchronization and communication, and configuration and control; additionally, coincidence processing has to be included at the concentrator module. The implementation of all blocks present in the figure will be detailed in §6.2.

The main differences between both module schematics are the lack of coincidence processing and Gigabit Ethernet link at the slave node, and the VCXO configuration as a fixed oscillator at the concentrator board. It should be clear how to modify them in order to implement either an intermediate acquisition-concentration module such as those in Fig. 6.3, or a pure concentration board: the former would imply moving the uplink from the SFP to the second IML port, whereas the latter would need both IMLs, coincidence processing and the SFP link, removing all ADC reception and subevent processing blocks as well as the front-end submodule.

6.1.2 Analog channels

The PCB digitizes ten different analog channels: eight energy and position signals, one timing signal, and one calibration signal. The latter is generated locally in the PCB, but the others come from external boards and are fed to the PCB using short cables. Analog conditioning stages are required for all of them in order to adapt their properties to the ADC specifications. This circuitry is different for each analog channel type although it shares a number of common blocks.

Energy and position signals

The schematic for each energy signal conditioning circuit is depicted in Fig. 6.7 (top). It is divided into three sequential stages that are described below:

- *Input amplifier*: This stage amplifies and terminates the input signal and provides a programmable DC offset. It consists in an inverting amplifier with fixed gain

  $$G_{\text{in}} = -\frac{R_{g2}}{R_{g1}} \tag{6.1}$$

  and an input resistor R_t intended as parallel termination for the coaxial cable transporting the signal to the PCB. The input impedance of the circuit is

  $$Z_{\text{in}} = R_t \parallel R_{g1} \tag{6.2}$$

  which must be equal to 50 Ω. The operational amplifier is biased at a variable DC level that is programmed using a *Resistive DAC* (RDAC), or *digital potentiometer*, i.e. a potentiometer whose tap position may be programmed through a digital interface. A large potentiometer value R_{RDAC} is chosen and its ends are biased by a resistive divisor at symmetric levels

  $$V^{\pm} = \pm V_{CC}\left(1 - \frac{R_{b1}}{2R_{b1} + R_{b2} \parallel R_{\text{eq}}}\right) \approx \pm V_{CC}\left(1 - \frac{R_{b1}}{2R_{b1} + R_{b2}}\right) \tag{6.3}$$

  provided that R_{b2} is much smaller than the equivalent parallel resistance of the RDACs, R_{eq}; the biasing divisor is common to all eight energy channels, hence $R_{\text{eq}} = R_{\text{RDAC}}/8$. The potentiometer tap is thus biased at a constant

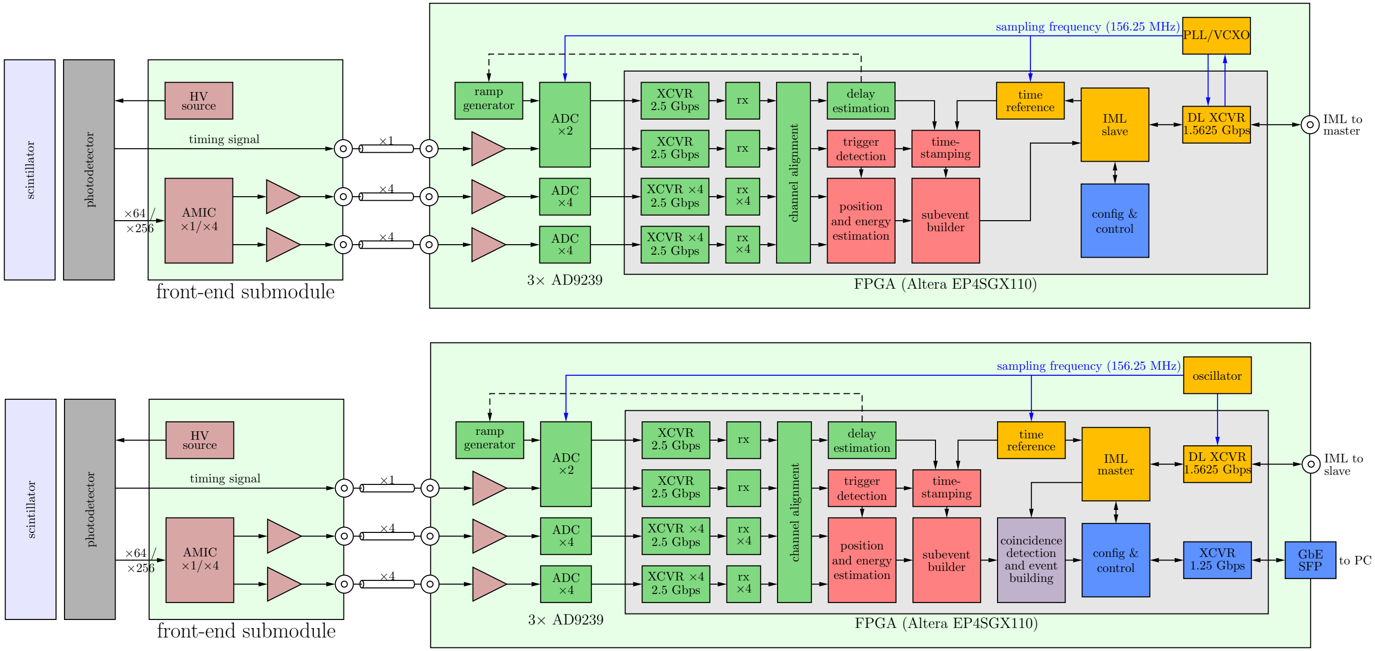


Figure 6.6: Schematic diagram of a full acquisition module (top) and a mixed acquisition-concentration module (bottom) using the designed boards and the existing AMIC modules. Logic blocks within the FPGA are color coded depending on the subsystem they belong to (green: sampling; red: subevent processing; violet: coincidence processing; orange: synchronization; blue: configuration).

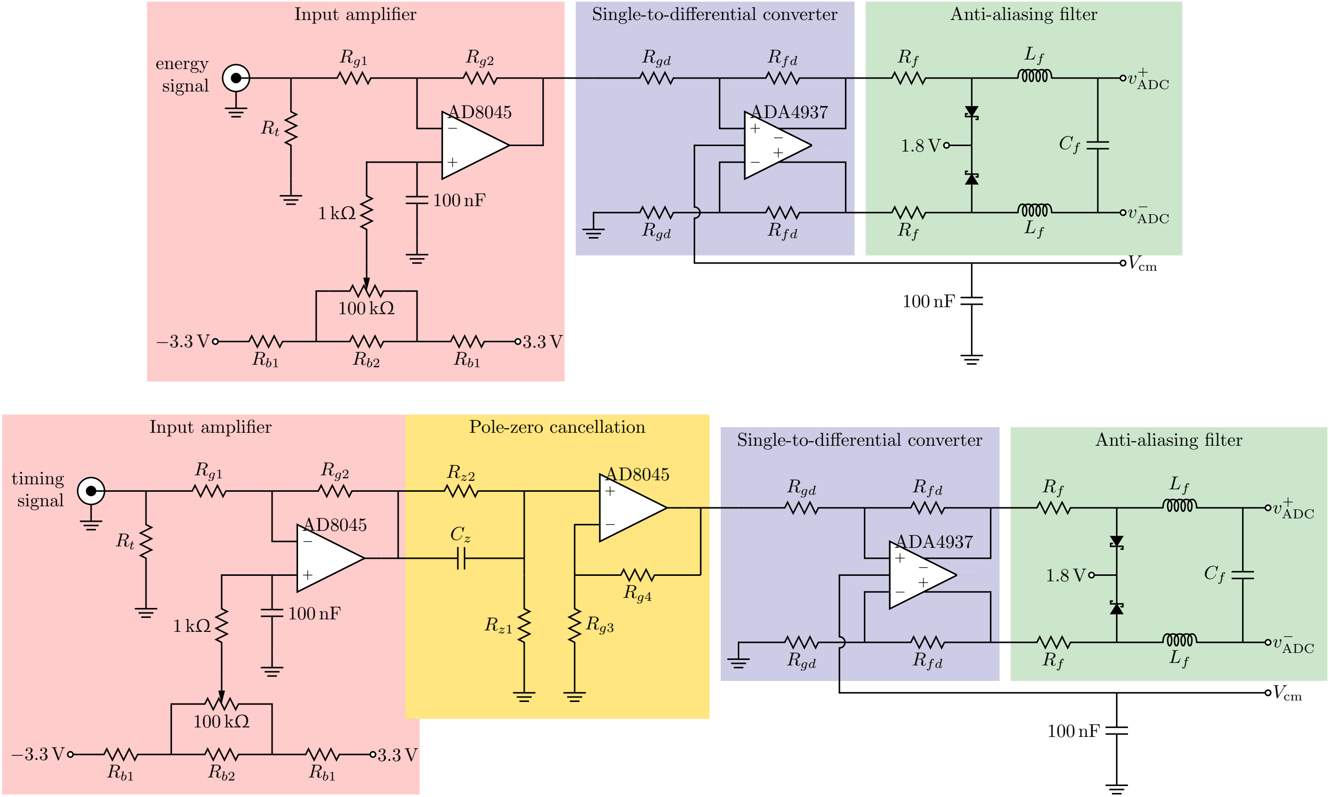


Figure 6.7: Schematic of the analog processing stages for energy signals (top) and timing signals (bottom).

voltage that can vary between V^- and V^+. An RC filter is included in order to reject thermal noise caused by the large potentiometer resistance.

This variable offset is used to push the resulting signal baseline as close as possible to the ADC full scale value: since the expected signals are unipolar, shifting their baseline to one end of the ADC voltage acquisition window instead of the simpler solution of keeping it at ground allows doubling the channel gain so that the expected signal range covers the full ADC input range, increasing SNR by a factor of two. Besides reconfigurability, the other reason to make the offset voltage programmable is the fact that energy signals are DC coupled, so baseline shifts are expected during operation that can be compensated against by online tuning of the RDACs.

- *Single-to-differential converter*: A second stage transforms the single ended input signal into a differential one using a differential amplifier, in order to adapt it to the ADC input specifications. The reference voltage for the conversion is fixed at ground. This stage has gain

$$G_d = \frac{R_{fd}}{R_{gd}} \tag{6.4}$$

and an input impedance equal to

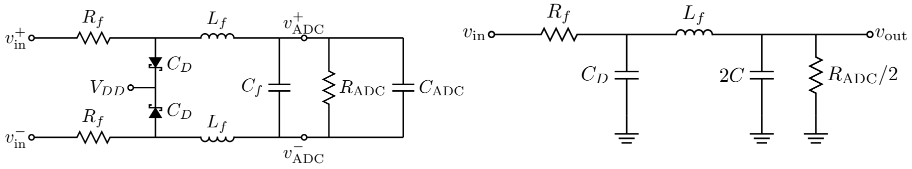
$$Z_{\mathrm{in}} = \frac{R_{gd}}{1 - \dfrac{R_{fd}}{2\left(R_{fd} + R_{gd}\right)}}. \tag{6.5}$$

The output common-mode voltage V_{cm} is provided by the ADC and filtered using a capacitor.

- *Anti-aliasing filter*: This stage implements two fundamental functions of analog front-ends in DAQ systems with sampling. The first one is the *anti-aliasing filtering* that is recommended at the input of any fast ADC in order to remove the coupling of high-frequency noise in the digitized signal. Recall that, by virtue of the Nyquist sampling theorem, a sampling frequency f_{s} only guarantees perfect reconstruction of low-pass signals with a bandwidth below $f_{\mathrm{s}}/2$; any signal component at a higher frequency f is reflected in the frequency domain and appears at the sampled output as indistinguishable from the low-pass frequency

$$f_{\mathrm{eq}} = \min\left\{f \bmod f_{\mathrm{s}}, f_{\mathrm{s}} - \left(f \bmod f_{\mathrm{s}}\right)\right\}. \tag{6.6}$$

Even if the signal bandwidth is below $f_{\mathrm{s}}/2$, the conditioning circuitry adds noise beyond that frequency that can be aliased into the passband. For this reason, an anti-aliasing low-pass filter is included as the last stage in front of the ADC with a cutoff frequency below $f_{\mathrm{s}}/2$. A simple filter is used here,



(a) Anti-aliasing filter model

(b) Equivalent differential small-signal circuit

Figure 6.8: Schematic models for analysis of the RLC anti-aliasing filter.

and the filtering is later enhanced by including a digital low-pass filter inside the FPGA.

The other purpose of this block is the shaping of the detector signal, so that their final properties are better suited to the later procedure of feature extraction, be it timing or energy information. Different shapings are possible depending on the desired effect on the sampled signals. This aspect is more important for timing signals than energy signals, so it will be discussed afterward.

In this case, a single passive, differential filter is implemented to perform both functions. It is a second order passive RLC filter that takes advantage of the differential encoding of the signal. Its roll-off of $-40\,\text{dB}$ per decade is not particularly high but adequate for noise rejection provided that the margin between the cutoff and Nyquist frequencies is large enough and a digital anti-aliasing filter is included in the FPGA.

The filter has been enhanced with the additional function of signal level clipping in order to keep the ADC input voltage levels within its absolute ratings; this is necessary as the ADC is powered at a lower supply voltage ($V_{DD} = 1.8\,\text{V}$) as the rest of the analog conditioning circuitry ($V_{CC} = 3.3\,\text{V}$). This function is accomplished by connecting a pair of diodes between the internal filter nodes and the corresponding ADC supply; Schottky diodes are chosen because of their low forward voltage drop $V_\gamma \approx 0.3\,\text{V}$, so that the ADC inputs always remain below $V_{DD}+V_\gamma$. Notice that this imposes a lower bound on R_f in the order of $10\,\Omega$ to $20\,\Omega$, as it must be able to withstand the DC current flowing from the differential amplifier to V_{DD} when saturated.

Analysis of the filter response may take into account the input impedance of the ADC, which is modeled as $R_{\text{ADC}} \approx 4\,\text{k}\Omega$ in parallel with $C_{\text{ADC}} \approx 2\,\text{pF}$ [112], and the equivalent capacitance $C_D \approx 30\,\text{pF}$ of the diodes, as outlined in Fig. 6.8a. By the Bartlett bisection theorem, the analysis is reduced to that of the equivalent circuit in Fig. 6.8b for differential mode, where $C = C_f + C_{\text{ADC}}$. In order to simplify the design, the diode capacitance C_D is neglected first in order to obtain approximate analytical expressions. In that case, the differential transfer function becomes

$$\begin{aligned}\frac{v_{\text{out}}}{v_{\text{in}}} &= \frac{\frac{R_{\text{ADC}}}{2} \parallel \frac{1}{2Cs}}{R_f + L_f s + \frac{R_{\text{ADC}}}{2} \parallel \frac{1}{2Cs}} \\ &= \frac{R_{\text{ADC}}}{R_{\text{ADC}} + 2R_f} \frac{1}{1 + 2\frac{R_f R_{\text{ADC}} C + L_f}{R_{\text{ADC}} + 2R_f} s + \frac{2R_{\text{ADC}} L_f C}{R_{\text{ADC}} + 2R_f} s^2}\end{aligned} \tag{6.7}$$

i.e. that of a second-order low-pass filter with parameters (passband gain, cutoff frequency and damping factor, respectively) given by

$$A_0 = \frac{R_{\text{ADC}}}{R_{\text{ADC}} + 2R_f} \approx 1 \tag{6.8}$$

$$\omega_0 = \sqrt{\frac{A_0}{2L_f C}} \approx \frac{1}{\sqrt{2L_f C_f}} \tag{6.9}$$

$$\zeta = \frac{A_0^{3/2}}{\sqrt{2L_f C}} \left(R_f C + \frac{L_f}{R_{\text{ADC}}} \right) \approx \frac{1}{\sqrt{2}} \left(R_f \sqrt{\frac{C_f}{L_f}} + \frac{1}{R_{\text{ADC}}} \sqrt{\frac{L_f}{C_f}} \right) \tag{6.10}$$

under the assumptions $R_f \ll R_{\text{ADC}}$, $C_f \gg C_{\text{ADC}}$.

In order to obtain useful design equations, let us rewrite (6.10) in the form

$$R_f \left(\frac{C_f}{L_f} \right) - \zeta\sqrt{2} \sqrt{\frac{C_f}{L_f}} + \frac{1}{R_{\text{ADC}}} = 0 \tag{6.11}$$

which is a quadratic equation on $\sqrt{C_f/L_f}$ whose solutions are given by

$$\sqrt{\frac{C_f}{L_f}} = \frac{\zeta\sqrt{2} \pm \sqrt{2\zeta^2 - 4R_f/R_{\text{ADC}}}}{2R_f}. \tag{6.12}$$

Taking $R_f \ll R_{\text{ADC}}$, the positive sign gives the root $\sqrt{2}\zeta/R_f$, whereas the negative sign yields 0. A better approximation is obtained by using a first order Taylor expansion, giving the root $\left(\sqrt{2}\zeta R_{\text{ADC}}\right)^{-1}$. Hence

$$\sqrt{\frac{L_f}{C_f}} = \frac{R_f}{\zeta\sqrt{2}} \text{ or } R_{\text{ADC}} \cdot \zeta\sqrt{2}. \tag{6.13}$$

Between these two values, it is preferable to use the first one for design, primarily because it yields an additional degree of freedom by choosing R_f appropriately as opposed to R_{ADC} which is fixed by the ADC and, moreover, can only be estimated. Substituting in (6.9), one finally obtains

$$C_f = \frac{\zeta}{\omega_0 R_f} \ , \ L_f = \frac{R_f}{2\zeta\omega_0}. \tag{6.14}$$

Incorporating the diode capacitance into the analysis yields a third-order low-pass transfer function whose poles are not given by a simple formula. The second-order design equations (6.14) are used instead to obtain preliminary component values, and the circuit is then simulated in order to ascertain that C_D does not cause a significant degradation of the filter response.

Timing signal

The analog processing circuit for the timing signal, depicted in Fig. 6.7 (bottom), is similar to that of energy signals except for the inclusion of an additional *pole-zero cancellation* stage after the input amplifier. This stage consists of a passive filter formed by components R_{z1}, R_{z2} and C_z followed by a non-inverting amplifier that compensates for the attenuation; the complete transfer function is

$$H_{\mathrm{pz}}\left(s\right) = G_{\mathrm{pz}} \frac{1 + R_{z2} C_z s}{1 + \left(R_{z1} \parallel R_{z2}\right) C_z s} \tag{6.15}$$

with DC gain

$$G_{\mathrm{pz}} = \frac{R_{z1}}{R_{z1} + R_{z2}} \left(1 + \frac{R_{g4}}{R_{g3}}\right). \tag{6.16}$$

The reasoning behind this shaping stage lies in its ability to modify the decay constant of incoming exponential pulses without adding undershoot artifacts to the signal. Consider the simple exponential model (4.10) for a photodetector pulse,[5] whose Laplace transform is equal to

$$I\left(s\right) = \frac{I_0}{1 + \tau_1 s}. \tag{6.17}$$

By applying the pole-zero filter to it, the resulting signal is of the form

$$I\left(s\right) H_{\mathrm{pz}}\left(s\right) = \frac{I_0 G_{\mathrm{pz}}}{1 + \tau_1 s} \frac{1 + R_{z2} C_z s}{1 + \left(R_{z1} \parallel R_{z2}\right) C_z s} \tag{6.18}$$

in the Laplace domain. The idea is to choose component values such that

$$\tau_1 = R_{z2} C_z \tag{6.19}$$

$$\tau_1' = \left(R_{z1} \parallel R_{z2}\right) C_z \tag{6.20}$$

so that the τ_1 terms in the numerator and denominator cancel each other and the resulting signal has transform

$$I\left(s\right) H_{\mathrm{pz}}\left(s\right) = \frac{I_0 G_{\mathrm{pz}}}{1 + \tau_1' s} \tag{6.21}$$

[5] The referenced equation corresponds to the scintillator output; however, a linear photodetector response maintains the shape in the received pulse.

i.e. that of an exponential pulse with decay constant $\tau_1' < \tau_1$. Hence, the filter has a narrowing effect on exponential pulses, reducing pile-up.

The exact value of τ_1 depends on the detector, and so (6.19) will either hold only approximately or force component value tuning. It can be shown ([20], pp. 593–595) that $R_{z2}C_z > \tau_1$ causes undershoot in the filtered exponential pulses, whereas $R_{z2}C_z < \tau_1$ results in no undershoot but a decay constant higher than τ_1'. Avoiding the appearance of undershoot is of prime importance, as it lasts for a relatively long time during which it has the effect of shifting the signal baseline and therefore introducing errors in amplitude and time estimation algorithms. Tuning the circuit so that (6.19) holds exactly yields the narrowest possible pulses while avoiding undershoot.

The shaping properties of the anti-aliasing filter were disregarded for energy signals but have a significant effect in the case of timing signals. Because time pick-off is implemented digitally in this particular DAQ, a minimum timing signal rise time has to be guaranteed so that enough rising edge samples can be captured for each valid subevent, according e.g. to the criterion established in (4.43). The low-pass filter provides this functionality, resulting in a longer peaking time the smaller the cutoff frequency (6.9) becomes.

Ramp generator

The schematic for the reference signal generation circuit is shown in 6.9. Its main block is the linear ramp generator located on the left, controlled by digital signals V_{charge} and $V_{\text{discharge}}$ connected directly to FPGA pins. The circuit has two separate working phases. In the charge phase, V_{charge} is high and the capacitor C_c is charged with a constant current V_{charge}/R_c provided by the FPGA, so that the output of the operational amplifier at time t after V_{charge} goes high has the value

$$V_{\text{output}}(t) = -\frac{V_{\text{charge}}}{R_c C_c}\, t \tag{6.22}$$

as long as it stays above saturation. In the discharge phase, $V_{\text{discharge}}$ goes high, closing the nFET switch and allowing C_c to be discharged through R_d.

This topology has been selected because the uncertainty in the ramp start time is determined exclusively by the FPGA switching characteristics instead of relying on a discrete transistor, with the hope that this benefits the matching between different boards. The timing of the capacitor discharge phase is not critical and therefore an external switch may be used, with an RC filter for the control signal in order to soften the transitions.

The rest of the circuits consists of the usual single-to-differential converter and filter blocks. However, two modifications have been made to the first of them.

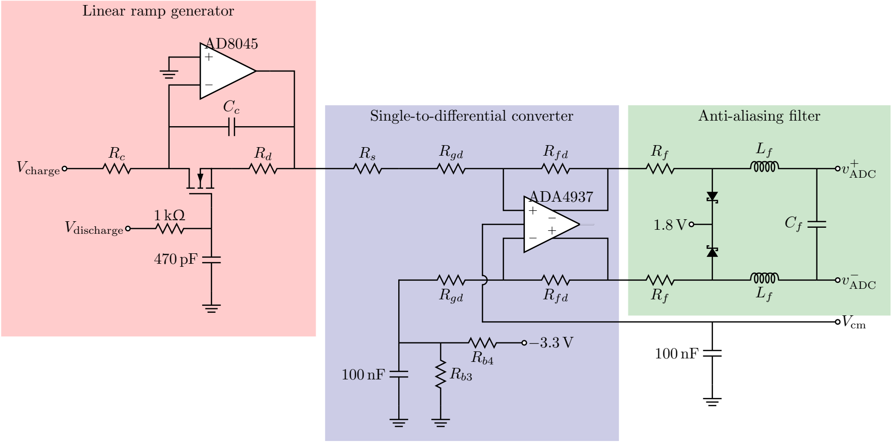


Figure 6.9: Schematic of the ramp generator circuit.

On one hand, a series resistor R_s is added at the input, so that it forms a voltage divisor with the equivalent input impedance Z_{in} given by (6.5) and lowers the input voltage level.[6] On the other hand, the negative amplifier input is not referred to zero but to a filtered negative bias voltage in order to account for the expected voltage range of the ramp; while the offset is not necessarily constant, it forces a linear relationship between the input and output signals that maintains the properties of the linear ramp.

Component values

The final choice of analog component values for all ten channels are listed in Table 6.3. For the amplifier stages, the total gain is determined by the expected input signal range and the dynamic range of the ADC (nominal 1.25 V), whereas the splitting of the total gain into the several amplifier stages has been designed in such a way as to maintain a balance between the general recommendation to apply the highest gains first as implied by the Friis Formula for noise ([391], pp. 280–283) and the limitation of output bandwidth of a given stage resulting from high gains due to finite gain-bandwidth product. Values of the pole-zero cancellation circuitry were obtained experimentally.

The same anti-aliasing and shaping filter has been used for all channels for simplicity, choosing R_f to be as small as possible while satisfying the power restriction, and a Butterworth response (i.e. $\zeta = 1/\sqrt{2}$) for compatibility with previous electronics. Other possibilities include RC^2 shaping ($\zeta = 1$) which is widely used for

[6] In the actual implementation, R_s and the top R_{gd} resistor are merged into a single physical component.

Component	Energy channel	Timing channel	Ramp channel
R_{g1}	750 Ω	750 Ω	
R_{g2}	1.6 kΩ	3 kΩ	
R_t	53.6 Ω	53.6 Ω	
R_{b1}	330 Ω	3.6 kΩ	
R_{b2}	1.2 kΩ	1 kΩ	
R_{z1}		100 Ω	
R_{z2}		330 Ω	
C_z		100 pF	
R_{g3}		100 Ω	
R_{g4}		300 Ω	
R_c			3 kΩ
C_c			1 nF
R_d			200 Ω
R_s			460 Ω
R_{b3}			680 Ω
R_{b4}			1.5 kΩ
R_{gd}	200 Ω	200 Ω	220 Ω
R_{fd}	200 Ω	680 Ω	220 Ω
R_f	39 Ω	39 Ω	39 Ω
L_f	150 nH	150 nH	150 nH
C_f	100 pF	100 pF	100 pF

Table 6.3: Analog component values on the acquisition boards.

radiation detector shaping ([20], pp. 588–590), and Bessel filters ($\zeta = \sqrt{3}/2$) that result in the lowest possible signal distortion. The latter is possibly better suited for the ramp channel but this has not been tested; to do so, the values $L_f = 120\,\text{nH}$ and $C_f = 120\,\text{pF}$ should be used.

6.1.3 Power distribution

The Power Distribution Network (PDN) is a complex subsystem of the acquisition board, due to the high number of different power voltage values required to feed the variety of on-board components, as well as the need to provide separate paths for certain blocks with the same nominal supply voltage but strong requirements on crosstalk noise coupling. PDN design has been traditionally neglected or considered trivial, but this situation has been changing for the past fifteen years as lower supply voltages with higher current peaks become prevalent, as its proper planning turns out to be critical for the performance of high-speed digital links and for noise coupling in analog subcircuits. In this section, the PDN design methodology and final result are described.

The first step towards the complete PDN design involves the identification of all different power domains, and the requirements of each of them as well as the existence of special restrictions on the dependency between them. For each domain, the following data are needed: nominal voltage V_{CC}, static supply current I_{CC}, dy-

Power domain	V_{CC}	I_{CC}	ΔI	ΔV	Z_{target}	L_{reg}	$f_{\min}$	C_{LF}
Analog +3.3V	3.3 V	800 mA	400 mA	50 mV	0.12 Ω	10 nH	1.9 MHz	0.7 µF
Analog -3.3V	−3.3 V	400 mA	200 mA	50 mV	0.25 Ω	10 nH	3.9 MHz	0.16 µF
Digital 3V3	3.3 V	800 mA	500 mA	16 mV	32 mΩ	1.5 µH	3.4 kHz	1.5 mF
Clocking	3.3 V	400 mA	200 mA	110 mV	0.55 Ω	10 nH	8.7 MHz	33 nF
ADC analog (×3)	1.8 V	600 mA	300 mA	80 mV	0.26 Ω	10 nH	4.1 MHz	0.15 µF
ADC digital	1.8 V	320 mA	160 mA	80 mV	0.5 Ω	10 nH	7.9 MHz	40 nF
FPGA core	0.9 V	1.5 A	750 mA	18 mV	25 mΩ	1.5 µH	2.6 kHz	2.4 mF
FPGA V_{CCDPLL}	0.9 V	23 mA	13 mA	18 mV	1.4 Ω	2 µH	110 kHz	1 µF
FPGA V_{CCA}	2.5 V	80 mA	30 mA	100 mV	3.3 Ω	10 nH	52 MHz	1 nF
FPGA V_{CCAUX}	2.5 V	8 mA	4 mA	100 mV	25 Ω	500 nH	8 MHz	1 nF
FPGA V_{CCAPLL}	2.5 V	20 mA	5 mA	100 mV	20 Ω	500 nH	6.3 MHz	1.3 nF
FPGA V_{CCH}	1.5 V	200 mA	100 mA	65 mV	0.65 Ω	10 nH	10 MHz	24 nF
FPGA V_{CCPT}	1.5 V	35 mA	10 mA	40 mV	4 Ω	500 nH	1.2 MHz	31 nF
FPGA 1.1V transceivers	1.1 V	600 mA	180 mA	37 mV	0.2 Ω	10 nH	3.1 MHz	0.25 µF
FPGA 3.0V I/O banks	3.0 V	900 mA	26 mA	15 mV	0.57 Ω	1.5 µH	60 kHz	36 µF
FPGA 2.5V I/O banks	2.5 V	1.1 A	105 mA	7 mV	65 mΩ	1.5 µH	7 kHz	3.6 µF
FPGA 1.8V I/O banks	1.8 V	100 mA	230 mA	35 mV	0.15 Ω	1.5 µH	16 kHz	67 µF

Table 6.4: Different power domains in the acquisition board. All initial specifications and computed design values are shown, except for the final list of decoupling capacitors.

namic supply current ΔI and admissible supply voltage variation ΔV. The static current is needed for the dimensioning of supply regulators and the estimation and validation of power dissipation. The dynamic current consumption, on the other hand, is directly related to ripple in the power supply and therefore to noise. The maximum ripple ΔV is usually given as a specification in device datasheets. When the two latter are not given, the estimation $\Delta I = I_{CC}/2$ is typically used, whereas ΔV is taken to be between 1 % to 5 % of the nominal supply voltage. Table 6.4 contains all power domains in this design and these initial specifications.

As a second step, the nominal voltage values and static consumptions are used to design a *power tree* that describes exactly how each of these power supply voltages is derived from the main supply input to the PCB, i.e. it defines the power inputs and the regulators used to convert them into the desired voltages. Due to the large expected total power consumption and the need for a negative supply, it was decided to use a special AC/DC converter to power the board that provides three supplies: 12 V/2 A, 5 V/5 A, and −5 V/0.8 A. The complete power tree is shown in Fig. 6.10. The first tree stage is composed of switching regulators, that have a high power conversion efficiency and can provide a higher output current at the expense of increased ripple at the output with frequency peaks; their output is thus fit for the supply of digital subsystems, but not for analog circuits where switching noise can couple to sensitive signals. A second stage of low-dropout (LDO) linear regulators is included for this purpose. These regulators merely cause a voltage drop and generate a stable output voltage, with

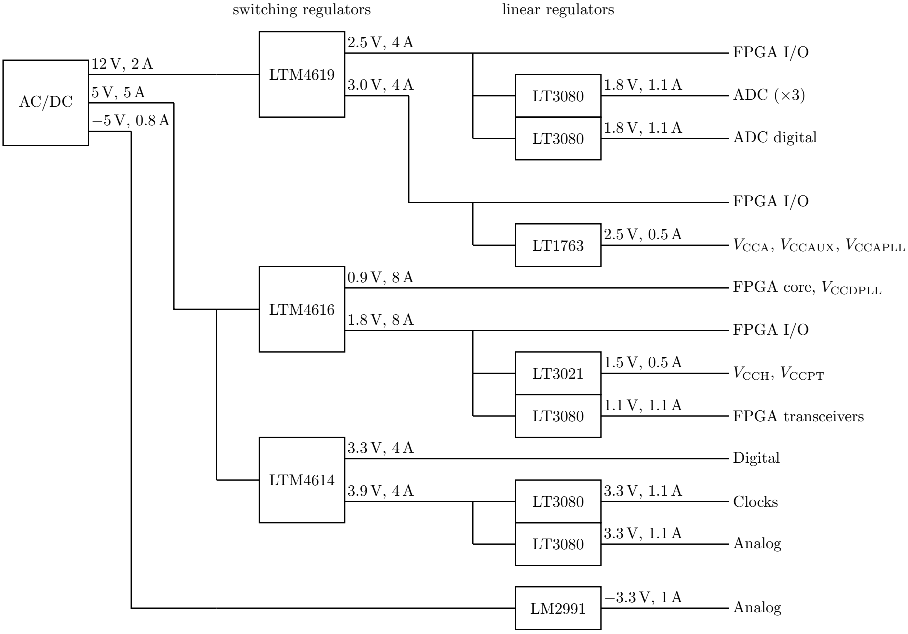


Figure 6.10: Power tree of the acquisition board.

better noise properties than a switching regulator,[7] but require their input to be at least 0.3 V higher than their output and do not modify the current; instead, it flows through the voltage difference and is dissipated as heat, resulting in a lower power efficiency and requiring careful thermal design. Some groups of power domains with the same nominal voltage are extracted from the same regulator; in that case, they are separated from each other with appropriate ferrite beads, modeled and simulated following the guidelines in [393].

The third and final step consists in the individual design of the decoupling network for each power domain. The design goal for each of them is to keep the supply voltage ripple below ΔV under current variations up to ΔI. This is accomplished by defining a maximum target impedance

$$Z_{\text{target}} = \frac{\Delta V}{\Delta I} \tag{6.23}$$

and then requiring the equivalent impedance Z_{PDN} of the power distribution network, as seen from each device's power inputs, to satisfy

$$|Z_{\text{PDN}}(f)| < Z_{\text{target}} \tag{6.24}$$

[7] While LDOs do not add much noise, they suffer from poor noise rejection at high frequencies, but this can be circumvented by adding a series ferrite bead [392].

at all frequencies f [394]. The PDN impedance model is outlined in Fig. 6.11 and consists of the following elements:

- The source is usually a voltage regulator and is modeled as having an equivalent RL type output impedance with values R_{reg} and L_{reg}, which are sometimes given in the regulator's datasheet but can be obtained from a SPICE simulation if not specified. The most important of them is the output inductance. In the case of ferrite-separated domains, this parameter can be estimated as the equivalent impedance of the ferrite bead.

- A number of decoupling capacitors are included that act as a charge buffer for the target device, providing transient current without having to retrieve it from the regulator. Two varieties are distinguished: small ceramic capacitors, and larger bulk capacitors i.e. tantalum or electrolytic. Capacitors are modeled as a series RLC circuit, with the nominal capacitance and two additional equivalent series parasitics ESR and ESL which determine the minimum equivalent impedance of the component and the series resonant frequency f_C at which it is attained; for $f > f_C$, the capacitor acts like an inductor. For precise design, ESR and ESL include the equivalent series parasitics introduced by PCB vias and short routes.

- If the supply voltage is located in a power plane on the PCB, then this plane forms a parallel-plate capacitor with the surrounding ground planes, with negligible ESR and ESL and good decoupling properties at high frequencies. Unfortunately, modeling this element requires an estimation of the PCB stack-up and floorplan so it cannot be included until later in the design process.

- The interconnect routing between the actual target device package and the power plane or the nearest decoupling capacitor is modeled as parasitic series resistance and inductance, the latter being predominant in PCB vias.

- Finally, the package and wire bonding of the target device also provide their own parasitics, which need to be included in the impedance model as perceived by the actual circuit die. Alternatively, their modeling may be substituted by a stronger requirement on ΔV.

In practice, the requirement (6.24) only needs to be fulfilled for a frequency range $f_{\text{min}} < f < f_{\text{max}}$ whose limits are given by the following considerations:

- The source impedance is equal to R_{reg} at low frequencies, which is very small and satisfies the impedance target trivially. The smallest frequency f_{min} where specific design is needed in order to meet the condition is determined

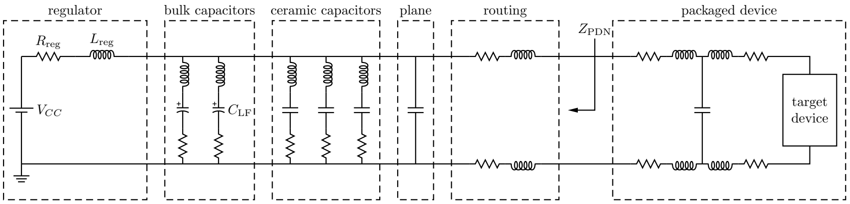


Figure 6.11: Equivalent PDN circuit model for impedance analysis and design.

by $|Z_{\text{reg}}(f_{\text{min}})| = Z_{\text{target}}$, which for $R_{\text{reg}} \ll Z_{\text{target}}$ translates into

$$f_{\text{min}} = \frac{Z_{\text{target}}}{2\pi L_{\text{reg}}}. \tag{6.25}$$

- For frequencies above 200 MHz to 300 MHz, the effect of Z_{PDN} is negated as package parasitics dominate. Therefore, it is only required to fulfill (6.24) up to a certain frequency in that range. In the particular case of the Stratix IV FPGA, an on-chip decoupling network is included that relaxes the PDN design requirements to $f_{\text{max}} = 70\,\text{MHz}$ for its exclusive power domains.

There are several possible design procedures for the decoupling network [395]. The one used in this design is known as Frequency Domain Target Impedance Method (FDTIM) or multi-pole method, and is based on the selection of capacitors of as many different values as possible in order to minimize their total amount and to achieve a maximally uniform Z_{PDN} as a function of f. The procedure starts by computing f_{min} and then determining the minimum decoupling capacitance C_{LF} needed to satisfy the impedance goal at that frequency, i.e. such that $|Z_{C_{\text{LF}}}(f_{\text{min}})| \leq Z_{\text{target}}$. Combining this with (6.25) yields

$$C_{\text{LF}} = \frac{L_{\text{reg}}}{Z_{\text{target}}^2}. \tag{6.26}$$

Bulk capacitors are then selected such that their total capacitance is at least C_{LF}; in practice, twice this value has been used as a safer limit.

The regulator source and the bulk capacitors form a first approximation to the final PDN with an impedance profile $Z_{\text{PDN}}^{(1)}(f)$ that stays below Z_{target} up to a certain frequency $f^{(1)} > f_{\text{min}}$. An iterative process then begins such that, in step k, a ceramic capacitor with a series resonant frequency $f_C \approx f^{(k)}$ are added in parallel to the network in order to lower the equivalent impedance to a new profile $Z_{\text{PDN}}^{(k+1)}(f)$ that satisfies the design goal up to $f^{(k+1)} > f^{(k)}$. Since f_C is increasing, this results in capacitors of increasingly smaller value being added to the network. The procedure ends when $f^{(k)} > f_{\text{max}}$ and the design goal has been reached.

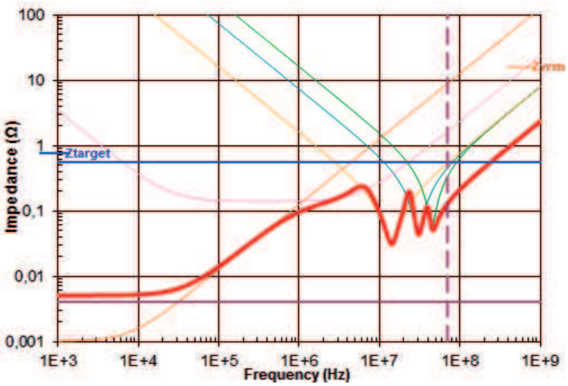


Figure 6.12: Example PDN impedance profile for the FPGA 3.0 V I/O banks domain. The thick line represents Z_{PDN} whereas the thinner lines correspond to the regulator, the bulk capacitor, and three ceramic capacitors. Other elements such as the power plane impedance are not included here.

For this design, the iterative process has been performed manually for each decoupling network with the help of the Altera PDN Design Tool [396], as it contains inbuilt models for the on-chip decoupling in each of the FPGA power domains; automation of this process is also possible using other software [395]. Figure 6.12 shows the resulting PDN for one of the power domains, where the complete impedance profile $Z_{\mathrm{PDN}}(f)$ can be seen as well as the contribution of the regulator source, the bulk capacitance, and three small capacitors with different resonance frequencies. It should be noted that the parasitics introduced by vias and routing need to be taken into account for small capacitors; in fact, for domains with particularly strong restrictions such as the FPGA core supply, it was required to employ special low-inductance capacitors and multiple connection vias per component.

6.1.4 Circuit board

The final circuit board is shown in Fig. 6.13, and assembly diagrams of the top and bottom board layers are included in Fig. 6.14. The final circuit size is 248.5 mm × 100 mm, split into separate sections for the analog and digital circuitry, the PDN and the clocking and communication subcircuits. Its total power consumption is around 20 W.

The board consists of 12 layers, whose stackup is outlined in Fig. 6.15. All 10 internal layers were required in order to implement the necessary power planes for the large amount of different power domains in the circuit, which are particularly dense below the FPGA; 3 full ground planes and 3 power planes are defined, plus partial power planes in each of the internal signal routing layers. Internal signaling is primarily devoted to very dense routing areas such as the connection between

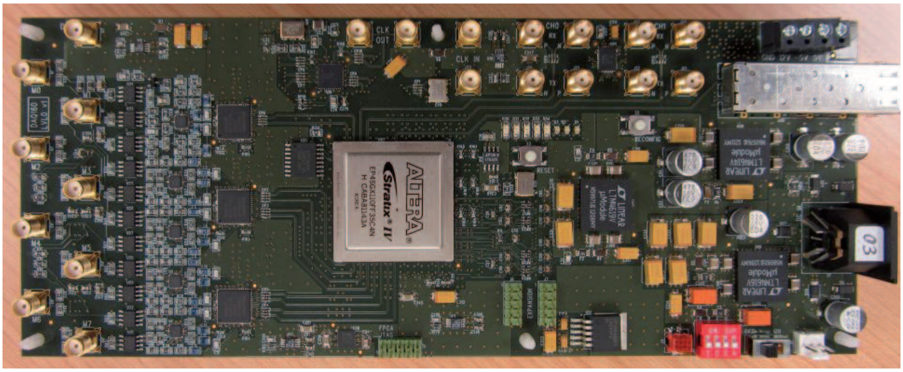


Figure 6.13: Picture of the final acquisition board, divided into an analog section (left), digital section (middle), power supply (right), and clocking and communication (top).

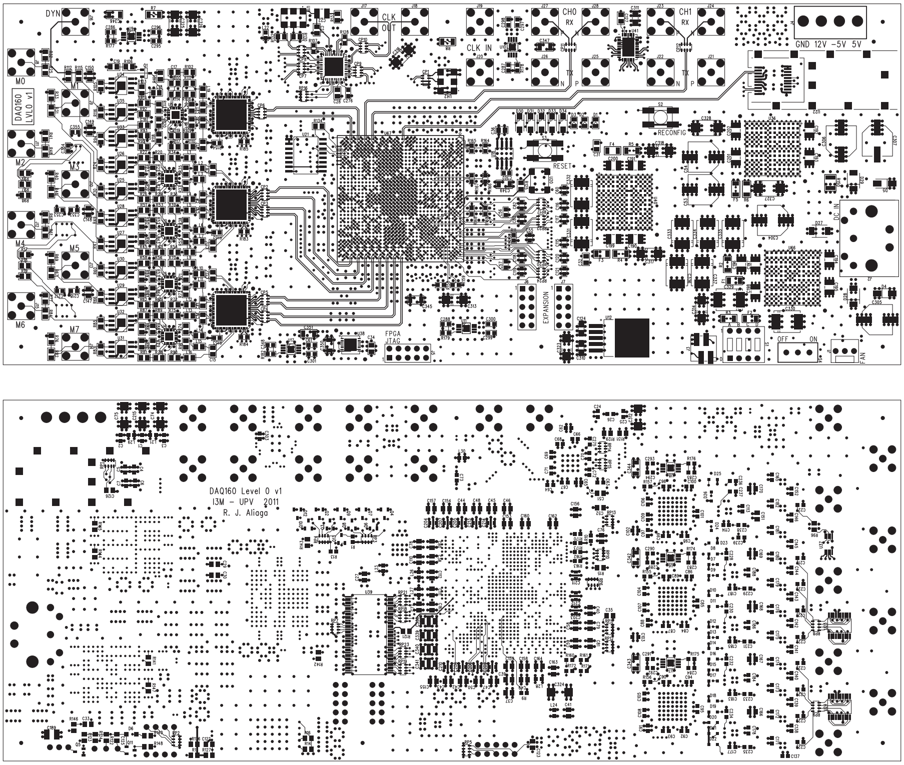


Figure 6.14: Assembly diagrams of the top and bottom board layers.

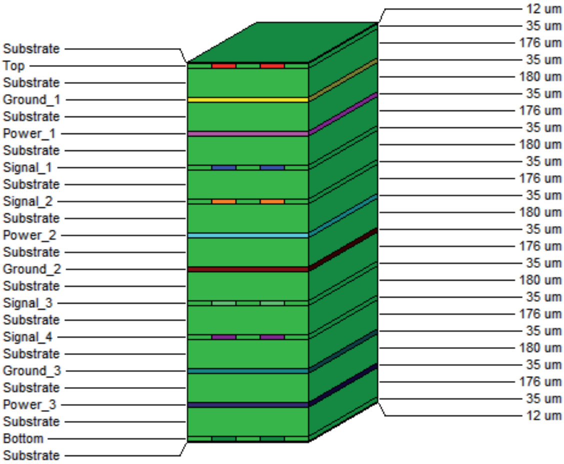


Figure 6.15: Stack-up of the 12-layer board. The internal signal layers are actually partial power layers with small regions dedicated to signal routing in congested areas.

the FPGA and the external RAM, and for sensitive differential signals such as clocks or high-speed lines; layers 8 and 9, located between two ground planes, were used for the latter in order to achieve optimal interference rejection.

6.2 Firmware design

This section describes the FPGA firmware for the acquisition boards. A single version of the firmware has been developed that contains all necessary logic modules for both the acquisition board and the mixed acquisition-concentration board, i.e. it corresponds to Fig. 6.6 (bottom). In particular, it implements a single IML that can be configured as either master or slave, and thus can be used only for the two-board setup.

A more detailed schematic of the global firmware architecture is shown in Fig. 6.16. A design methodology has been followed wherein a clear division is established between the main datapath, i.e. the logic dedicated to high speed data acquisition and processing, and the controlpath dedicated to control, configuration and monitoring. The main logic blocks in the datapath are shown on the left, and run with the local clock that is also used for ADC sampling, except for the IML receiver which uses the recovered clock locally. The controlpath is implemented as a SoC based around a Nios II microprocessor [63] running control software and an Avalon data bus [397] which all other elements are connected to, running with its own independent local 100 MHz clock. These elements include interface buses for configuration of on-board circuitry such as ADCs, RDACs and the PLL, an

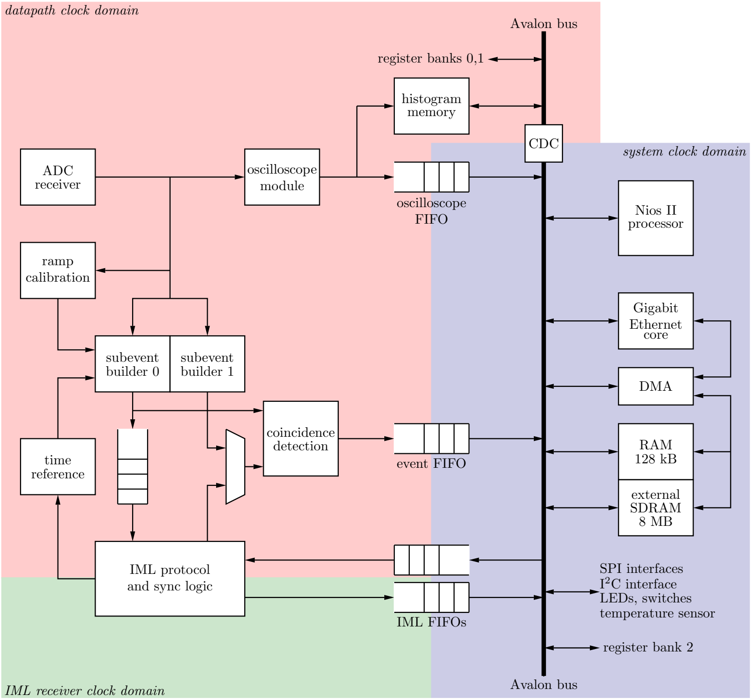


Figure 6.16: Schematic of the FPGA firmware for the acquisition modules, with the datapath on the left side and the SoC-based controlpath on the right side.

internal temperature sensor, 128 kB of on-chip RAM, controllers for the external RAM modules, and an Altera Triple Speed Ethernet core [398] that implements the MAC layer of the Gigabit Ethernet link. A DMA controller is included that allows the Ethernet core to handle data transfers automatically without microprocessor intervention.

Data transfer between the datapath and the system bus is carried out mostly through dual-port FIFO queues that also implement the clock domain crossing function. Queues are present for the storage of captured ADC waveforms, subevent and event data, and for message transfer between microprocessors on different boards over the IML; they can be accessed by the processor directly by reading or writing the appropriate bus address. A memory interface is used for the results of the histogramming module that will be described in §6.2.1, accessed through a clock domain crossing bus bridge. For control connections between the SoC and the datapath, a special register bank core has been implemented that consists of 32 write and 32 read 32-bit registers each. Control signals are individually connected to these registers, using synchronizers whenever there is a clock domain crossing. Three different register banks are used: one for control of the ADC

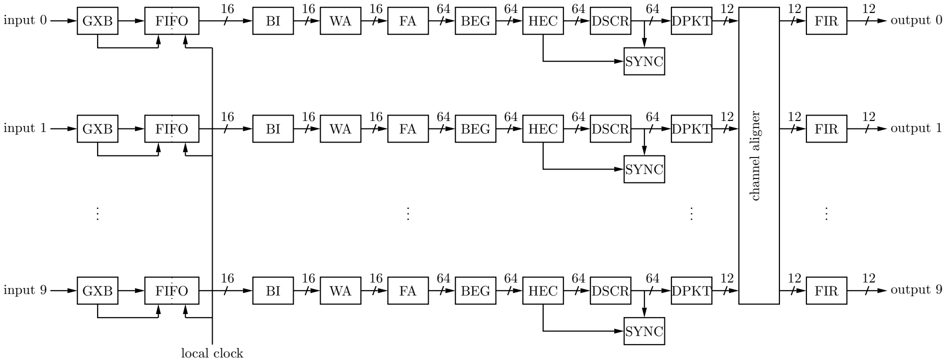


Figure 6.17: Block diagram of the full ADC receiver module for 10 channels. The following acronyms have been used for its components: GXB, Altera GXB transceiver; FIFO, elastic buffer; BI, bit inverter; WA, word aligner; FA, frame aligner; BEG, bit error generator; HEC, Hamming error corrector; DSCR, descrambler; DPKT, depacketizer; SYNC, channel sync monitor; FIR, digital filter.

receiver module, one for subevent and coincidence processing, and a third one for all remaining circuitry.

The main logic modules in Fig. 6.16 are described next. In general, modular design has been attempted, writing Verilog modules with as many parameters as possible including numbers of channels or data widths, in order to accommodate possible changes or extensions and to allow reuse in other designs.

6.2.1 ADC receiver module

The ADC receiver is one of the largest modules in the acquisition firmware. Its inputs are the self-synchronous data streams generated by the three AD9239 devices on board, corresponding to all 10 digitized channels in the PCB (eight moments, one dynode, and the ramp calibration signal), and it outputs one filtered sample per channel per clock cycle within the local clock domain. The ADCs generate digital data organized into complex 64-bit frames that contain an 8-bit header, four 12-bit samples, and a 8-bit *Error Correcting Code* (ECC), filtered by a scrambler in order to aid in embedded clock recovery. The frames from all channels need to be received, decoded and descrambled, and their contents have to be aligned and filtered.

The contents of the ADC receiver are pictured in Fig. 6.17, where each channel is shown to be composed of a number of sequential blocks. Each of these blocks has a registered output and so represents one pipeline stage in ADC channel decoding, with the goal of shortening register-to-register paths and being able to operate at 156.25 MHz. Details about these stages are given below:

- *Transceiver*: The first element in each channel is an Altera GXB transceiver that converts the incoming bit stream into parallel words of length $m = 16$, so that the parallel clock frequency matches the local frequency. Only the receiver part of the transceiver is used. An elastic buffer FIFO is used at the end in order to synchronize the output word stream with the local clock. No assumptions on deterministic latency are made.

- *Bit inverter*: This stage can be configured to invert the polarity of all bits in the received words. It is not necessary in a final design but it was included in the prototype anyway in order to correct the potential error created by routing the differential ADC output signals in the wrong order.

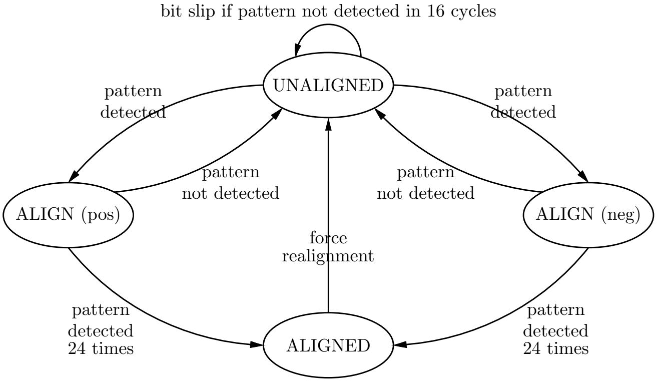
- *Word aligner*: The transceiver does not take into account the logical ordering of parallel words, therefore a simple word aligner block is included that looks for a specific 16-bit pattern in the received data. Because of the internal GXB architecture, two possible byte orderings are possible within the delivered 16-bit words, so the circuit looks for both possible forms of the pattern. Reception of several wrong patterns triggers a parallel word shift using the transceiver's bit slip interface. Alignment is reached whenever the pattern is successfully received a predetermined amount of times in a row. The word aligner algorithm is detailed in Fig. 6.18a.

- *Frame aligner*: This block contains a shift register that turns four consecutive words into a parallel 64-bit word. A valid frame is thus obtained every four clock cycles. Correct alignment is achieved by detecting a predetermined 64-bit pattern several consecutive times. The algorithm is illustrated in Fig. 6.18b.

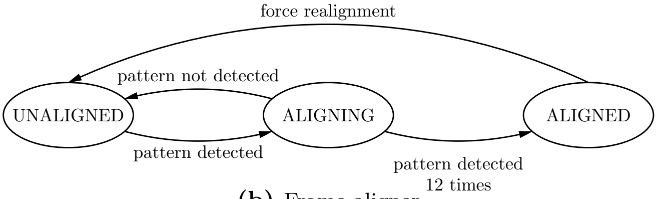
- *Bit error generator*: This stage simply applies a programmable bit error pattern to the received frames, i.e. a 64-bit XOR mask. This block is not necessary in a final design and is only intended as a debug aid for the ECC.

- *Hamming error corrector*: This block is responsible for the correction of possible bit errors in the received frames. The ADC outputs data using a Hamming (57,63) code ([399], pp. 420–421) that can detect and correct single bit errors, plus an additional parity bit that makes it possible to additionally detect double bit errors.[8] These 7 bits can be found in the ECC field of the AD9239 frame, together with an extra data bit that is stuck at 0. Error correction requires the combinational computation of the Hamming and parity bits corresponding to the received frame. A non-zero parity bit indicates the presence of a single bit error at the frame position given by the 6 Hamming bits, and is automatically corrected. A zero parity bit with non-

[8] Such an encoding is usually called SECDED, for Single Error Correction, Double Error Detection.



(a) Word aligner



(b) Frame aligner

Figure 6.18: State diagram of the word aligner and frame aligner FSMs within the ADC receiver. The amount of clock cycles that trigger state changes were chosen arbitrarily.

zero Hamming bits indicates a double bit error that cannot be recovered. The block includes a resettable error counter for debug and monitoring.

- *Descrambler*: The AD9239 includes a couple of standard scramblers that can be activated in order to balance the DC level and generate additional edges in its digital output stream, that aid in embedded clock recovery at the GXB. The SONET scrambler has been chosen, with LFSR polynomial $x^7 + x^6 + 1$, due to its smaller order and therefore more compact implementation. The descrambling circuit obtains the XOR of the corrected frame and the appropriate bit-shifted versions of it in parallel, as indicated in [112].

- *Depacketizer*: This block is just a word serializer, that accepts one frame every four clock cycles and outputs its 12-bit samples in the correct temporal order, one for every clock cycle. The resulting output is thus the stream of digitized samples.

- *Channel synchronization monitor*: Channels are constantly monitored for frame errors, given either by the Hamming error corrector stage or by an unexpected frame header after descrambling. Reception of five consecutive frames with error is interpreted as a loss of data synchronization between

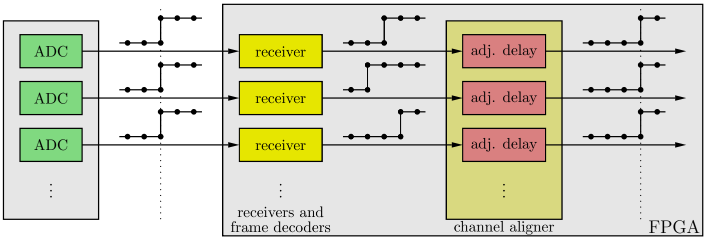


Figure 6.19: Channel alignment procedure inside the FPGA. ADCs are programmed to force a transition between two specific values, and that transition is time-aligned by channel-dependent adjustable delays.

the ADC and the FPGA and triggers an error signal that can only be reset manually.

- *Channel aligner*: A channel alignment procedure is necessary after ADC frames are received and decoded, because the latency of the transceiver and frame decoding logic for each channel may be different.[9] The channel aligner block accomplishes this by introducing adjustable delay lines (i.e. shift registers) for every channel, so that the recovered ADC samples are selectively delayed in order to correct the latency differences. Adjustment is carried out by forcing a transition between two specific values at all channels at the same time using the ADC configuration interface, looking for these transitions at the aligner inputs, and selecting the appropriate delay line taps according to the time of transition detection. The procedure is illustrated in Fig. 6.19. The maximum delay line length has been designed considering the possible latency ranges of each receiver block.

- *FIR filter*: The last block in each channel is a digital low-pass FIR filter, intended to complement the effect of the external anti-aliasing filter by introducing a sharp cutoff between the analog filter bandwidth and the Nyquist frequency. As a first step, a standard FIR filter was obtained with 31 taps for sufficient attenuation slope and symmetrical coefficients in order to guarantee linear phase and allow for a more efficient implementation. This filter was then simplified by trimming the smallest coefficients and adjusting the others so that their Hamming weight is as small as possible, i.e. 1 or 2, removing the need for multipliers as the coefficient weighting can then be accomplished using just one adder. A filter with appropriate coefficients was selected such that this procedure resulted in minimal distortion of its response. The resulting circuit is the filter shown in Fig. 6.20, of order 29, with a cutoff frequency around 40 MHz and an attenuation above 35 dB at

[9] The latency of the blocks implemented in user logic can be known exactly, but total latency would still be unknown because no assumption of deterministic latency at the transceiver is being made.

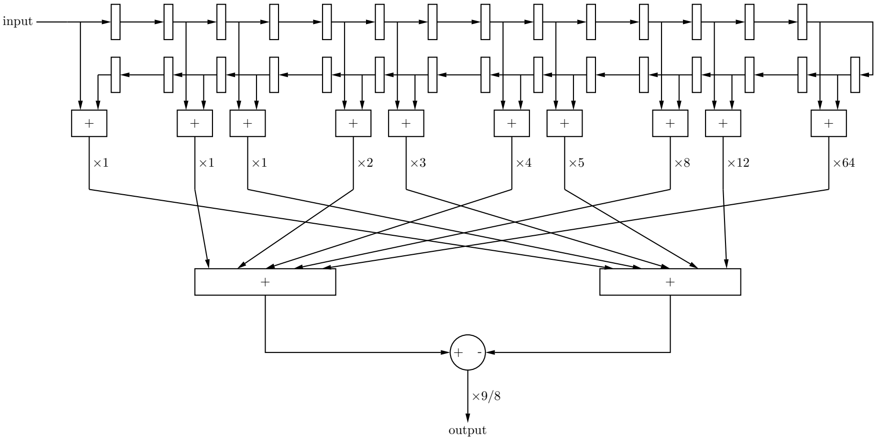


Figure 6.20: Schematic of the final FIR filter block. The multiplication by constant coefficients is hardwired and implemented using single adders.

the Nyquist frequency. An additional gain factor of 9/8 is added at the end of the circuit in order to compensate for the attenuation introduced by the simplified filter. Only the 12 most significant bits of the result are used in the subsequent processing stages for compatibility with the unfiltered signal.

Several configuration and status registers are available for interface between the Nios II controller and the ADC receiver block. These include individual enable and alignment reset signals for the different blocks in each channel, as well as the various patterns used in the three alignment procedures.

The reset and initialization procedure for the ADC receiver block is controlled by software on the microprocessor. It is executed on system power-up and can be repeated on demand. It consists in the following steps:

- The three ADCs are reset using their `PDWN` and `RESET` pins as explained in their datasheet.
- Their internal registers are configured for normal operation and unused channels are disabled. Specific frame headers for every channel are programmed. The scrambler and ECC are initially disabled.
- The routine waits until the transceivers have successfully locked in to the incoming data streams and are generating valid parallel words.
- The word alignment procedure is carried out first. The AD9239 allows the programming of a 64-bit training pattern to be continuously repeated at its output. A common pattern consisting of a repeated 16-bit word is used for

all ten channels. This step finishes when the status registers indicate that all word aligners are locked.

- For frame alignment, a full 64-bit pattern is used, until the aligners in all channels are locked.
- The scrambler and ECC are enabled.
- All ADC channels are programmed to output their negative full-scale value, i.e. 0, and the channel alignment procedure is enabled. All ADC channels are then simultaneously programmed to output their positive full-scale value, i.e. 4095. The channel aligner block then searches for transitions from 0 to 4095 in all channels, obtains the relative delays between them, and uses them to set up the programmable delay lines.
- The ADCs are finally programmed to output their converted values for normal operation.

6.2.2 Oscilloscope module

The oscilloscope module allows the capture of full waveforms coming directly from the ADC receiver module, and is mainly used as a monitoring tool. Its structure is detailed in Fig. 6.21. It contains of four identical channels that take the 12-bit sample streams coming out of each of the 10 ADC receiver channels as inputs (including the reference ramp), and select one of them for capture. Each channel contains a pre-trigger memory consisting of a shift register of variable length between 0 and 240 in steps of 16, made up of shift registers of different depths with configurable interconnections, in such a way that the control word $b_3b_2b_1b_0$ is the binary expression of the number of delay steps of 16 cycles. The output of these shift registers are connected to the oscilloscope FIFOs, with room for 16384 samples each, that hold the captured waveforms and can be read out from the control system bus.

The first oscilloscope channel is fed to a trigger unit with a small FSM that controls the write signal of the waveform FIFOs. Trigger threshold and polarity are configurable, and a trigger condition is detected whenever the current sample is higher than the last one and higher than the trigger threshold for positive polarity, or lower than both for negative polarity. Triggers may also be issued manually using the control register interface. Each trigger causes the waveform FIFOs to capture samples up to a configurable capture window or until the FIFOs are full. The delay introduced by the pre-trigger memories makes it possible to capture samples corresponding to a small time interval before the trigger condition. After each capture, the trigger needs to be reset manually.

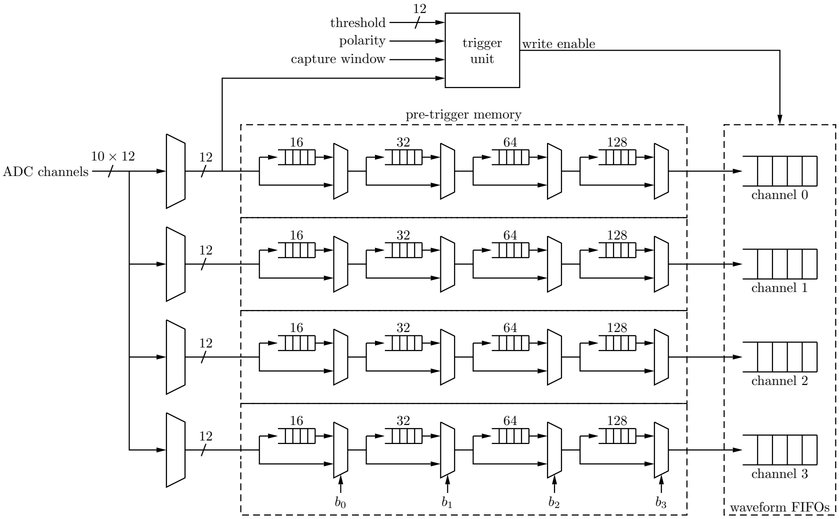


Figure 6.21: Schematic of the oscilloscope module with 4 channels, with details of the programmable pre-trigger memory contained in each one.

6.2.3 Histogram module

In addition to the oscilloscope, a small module for the automatic computation of histograms of the ADC channel sample values has been designed, given that such a function can be implemented much more efficiently using a dedicated module than resorting to waveform capture with the oscilloscope and later readout and processing of each sample; moreover, it allows for very long capture windows without going through the process of multiple waveform captures with the maximum length of 16384 imposed by the oscilloscope FIFO size.

The main idea for implementation is to use a memory block with a 12-bit address bus signal that is connected to the oscilloscope channel 0 output, so that whenever a signal sample takes the value n, the memory word at address n is increased by 1. After a specified capture window, each memory word contains the amount of times that the corresponding sample value was attained. This simple implementation is shown on the right side of Fig. 6.22; unfortunately, it results in data hazards and invalid results whenever the same value appears in consecutive or almost consecutive samples (which is, incidentally, the most common situation).

A slightly more complicated structure is needed to circumvent this problem, as shown in Fig. 6.22. The input sample first goes through a small cache queue that keeps track of the last three different sample values that were received and the amount of times they have appeared since entering the queue. After the queue, the simple memory-based implementation is located, where memory positions are

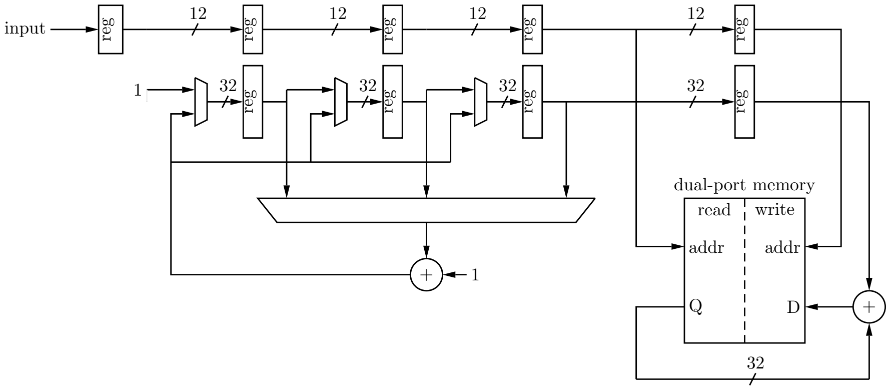


Figure 6.22: Schematic of the histogram module datapath, with the queue section on the left and the memory on the right. Control signals and multiplexers for bus access are not shown.

no longer incremented by 1 but rather by the repetition count stored in the queue. For each incoming sample, only the queue or the memory section is active, depending on whether the new sample is already in the queue or not. With this approach, consecutive memory accesses always correspond to different addresses and data hazards are avoided. A small control FSM is needed for activating the proper section, flushing the queue after capture, and managing shared access to the memory block with the Avalon bus.

6.2.4 Ramp calibration module

As was described in §6.1.1, the latency t_{AF} between the ADCs' sampling and the output of the ADC receiver module inside the FPGA is estimated by measuring the difference t_{FF} between two time values: the first one corresponds to the FPGA causing the control signal V_{charge} to go high, starting the ramp generator circuit, and the second one is the timestamp assigned to the ramp start as reconstructed from the received samples. A hardware module has been implemented that receives the stream of samples $r[n]$ from the ramp channel and carries out the whole procedure automatically, without software intervention.

The main idea behind the circuit is to continuously keep track of the line of best fit to the waveform formed by the last M received samples, for an appropriate value of M i.e. large enough that the fluctuations due to noise in the ramp signal are filtered out. This line, which varies with each sampling instant n, is described by the two linear regression coefficients $a_0[n]$, $a_1[n]$ and may be expressed as

$$l(t) = a_0[n] + a_1[n](t - t_0) \qquad (6.27)$$

where t_0 is the origin reference; in this case, it is the time reference of the first sample in the waveform i.e. $r\left[n-M+1\right]$. Computing the regression coefficients recurrently, besides being efficient, allows the module to use the momentary value of the fit slope $a_1\left[n\right]$ as a trigger condition for the detection of the ramp by checking whether it crosses some threshold value, or as a discharge condition if it keeps a very small absolute value.

The recurrent equations for the running linear regression coefficients are derived in Appendix B. Specifically, the zeroth and first order moments R_0, R_1 of the set of the last M samples are obtained recursively as (cf. (B.12) and (B.14))

$$R_0\left[n\right] = R_0\left[n-1\right] + r\left[n\right] - r\left[n-M\right] \tag{6.28}$$

$$R_1\left[n\right] = R_1\left[n-1\right] + Mr\left[n\right] - R_0\left[n\right] \tag{6.29}$$

where a shift register of length M is needed for R_0. The linear regression coefficients can be obtained as fixed linear combinations of these two moments, given by (B.8) with coefficients (B.7). In order to simplify the implementation, the following multiples of the linear regression coefficients are computed instead:

$$a_0'\left[n\right] = \frac{M\left(M+1\right)}{2}a_0\left[n\right] = \left(2M-1\right)R_0\left[n\right] - 3R_1\left[n\right] \tag{6.30}$$

$$a_1'\left[n\right] = \frac{M\left(M^2-1\right)}{6}a_1\left[n\right] = -\left(M-1\right)R_0\left[n\right] + 2R_1\left[n\right] \tag{6.31}$$

Notice that, if M is taken to be a power of two, all of these operations can be implemented using simple adders as the multiplicative coefficients have a very low number of active bits.

These coefficients are used to determine the ramp start time by computing the intersection between two lines corresponding to both circuit states: a horizontal line for the idle, discharged state, and the line of best fit to the ramp charge waveform, as outlined in Fig. 6.23.[10] The actual analog waveform may have a smoother transition between both states so that there is no clear "start time", however this definition provides a uniquely determined reference that can be used for time pick-off as long as it remains consistent between different boards.

The procedure starts by detecting the discharged state and latching the baseline value B, which is estimated as the average of the last few samples and therefore

[10] It is also possible to refine the method by considering a full line fit to the baseline waveform, i.e. not necessarily with zero slope, at the expense of increased computational complexity. It can be shown that the error introduced by the simplified procedure is approximately

$$\varepsilon \approx \frac{\alpha_{\mathrm{idle}}}{\alpha_{\mathrm{ramp}}}\left(t_{\mathrm{FF}} + \frac{M}{2}T_{\mathrm{clk}}\right) \tag{6.32}$$

where α_{idle} and α_{ramp} are the waveform slopes during the idle and ramp states, respectively. For appropriate values, this accuracy loss is negligible.

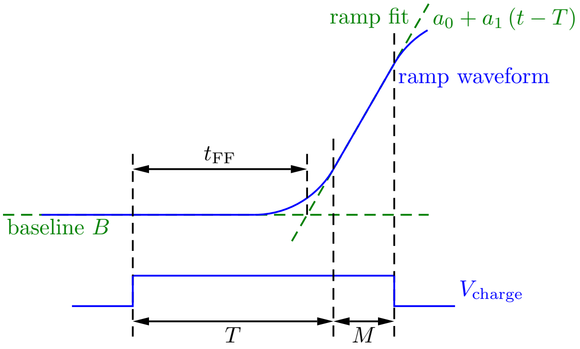


Figure 6.23: Illustration of the estimation of the ramp start time as the intersection between the lines of best fit to two different waveform sampling windows.

given by

$$B = \frac{R_0[0]}{M}, \tag{6.33}$$

taking this instant as the origin. The ramp charge signal is then triggered. After a certain number T of clock cycles, the ramp charge state is detected in the sampled waveform, and capture continues for M samples until linear coefficients a_0, a_1 are obtained corresponding to a sampling window located entirely within the ramp. The ramp start time t_{FF} then satisfies $B = l(t_{\mathrm{FF}}) = a_0 + a_1 (t_{\mathrm{FF}} - T)$ and therefore

$$t_{\mathrm{FF}} = T - \frac{a_0 - B}{a_1}. \tag{6.34}$$

It is thus necessary to perform a division to obtain t_{FF}. However, the numerator and denominator in the fraction in (6.34) can be obtained from the available values a'_0 and a'_1 by prescaling them appropriately with a factor $M\left(M^2 - 1\right)$:

$$\begin{aligned} \frac{a_0 - B}{a_1} &= \frac{M\left(M^2-1\right)a_0 - M\left(M^2-1\right)B}{M\left(M^2-1\right)a_1} \\ &= \frac{2\left(M-1\right)a'_0 - \left(M^2-1\right)R_0[0]}{6a'_1}. \end{aligned} \tag{6.35}$$

Notice that, once again, the multiplication coefficients in the numerator and denominator have a very low amount of active bits when M is a power of two and can thus be implemented as adders.

After obtaining the estimation of t_{FF}, the ramp circuit is discharged, and the measurement is repeated approximately every 100 µs.

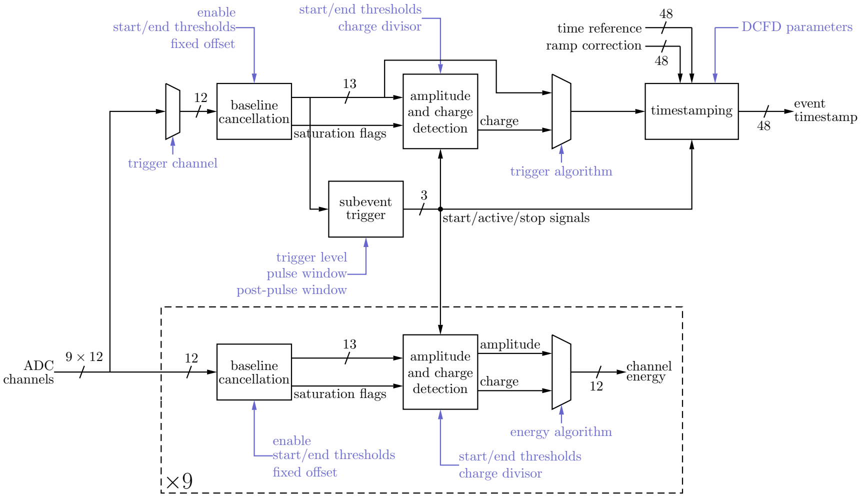


Figure 6.24: Schematic of the subevent builder module, with the trigger signal processing at the top and the energy signal processing at the bottom. Configuration registers are shown in light blue.

6.2.5 Subevent builder module

The subevent builder block is responsible for the detection of photodetector pulses within the ADC sample streams, the extraction of their features i.e. energy, position and timestamp estimation, and the formatting of the resulting subevent data for use by the coincidence detector and higher level layers. Each FPGA contains two completely independent subevent detector modules, and can thus be used to process coincidences between two detectors by connecting their trigger signals to the same board.

The structure of the module is depicted in Fig. 6.24. For each of the ADC receiver channels except the ramp, the energy is estimated independently using a baseline correction block first followed by pulse amplitude and charge detection. In addition, one of the 9 channels is selected as the trigger channel that will be used to determine the subevent trigger condition and the associated timestamp. A separate baseline cancellation block is included for it in order to allow for different trigger detection and energy estimation settings on the channel used for timing. The local subevent trigger is issued whenever the baseline-corrected trigger signal exceeds a given threshold level, and control signals are generated for pulse start and end that drive the charge and timestamp estimation logic. A fixed pulse window of programmable value is used, and a post-pulse window can be defined that allows vetoing further triggers until its end.

4	48	12	12	12	12	12	12	12	12	12
ID	timestamp	channel 0 energy	channel 1 energy	channel 2 energy	channel 3 energy	channel 4 energy	channel 5 energy	channel 6 energy	channel 7 energy	channel 8 energy

Figure 6.25: Subevent frame format.

For each detected subevent, a 160 bit frame is generated whose format is described in Fig. 6.25. It contains a 48-bit timestamp using a fixed point representation with 12 fractional bits as in §3.5.1, and a 12-bit energy value for each of the nine detector signal channels in the board. A 4 bit module identification field is included, 3 bits of which are programmed using the configuration registers; the last one discriminates between both subevent detectors in the module.

Energy estimation

For energy estimation, the first processing step is the correction of the signal baseline, i.e. the signal level during absence of pulse, which the pulse amplitude is referred to. At any given moment before the start of a pulse, the instantaneous signal level may be different from the baseline due to the remaining effect of the tail of the previous pulse on the same channel, and it may be below or above the actual baseline depending on whether the last pulse caused undershoot or not. This starting signal level is added to the detector pulse waveform and results in distortion of its amplitude, charge or any other parameters that rely on a baseline reference. The purpose of the baseline cancellation logic is to estimate and correct the baseline, fixing the signal reference at 0.

The schematic of the block is shown in Fig. 6.26 and is loosely based on the previous DAQ generation at EDNA ([193], pp. 226–228). It toggles between two states corresponding to the presence or absence of pulse at the current sample. The pulse state is activated whenever the difference between two consecutive samples exceeds a given threshold, and deactivated when the input signal level goes below another threshold value. The baseline estimator is active only during those samples that correspond to the absence of pulse, and computes the baseline as the running average of the last 64 samples during pulse absence, i.e. the mean signal value for the last 400 ns excluding pulses. The baseline is then subtracted from the input, resulting in 13-bit signed output values. Alternatively, the baseline estimator can be disabled and a fixed, programmable baseline value can be used, typically 0. In any case, flags for input signal saturation (i.e. input samples with value 0 or 4095) are generated together with the baseline-corrected output for monitoring and filtering purposes.

The second and final processing step in the designed firmware consists in the estimation of the amplitude and charge of the given pulse signal. The amplitude estimation is given by the maximum sample value attained by the signal during the pulse window, whereas the charge is given by its integral i.e. the sum of all samples

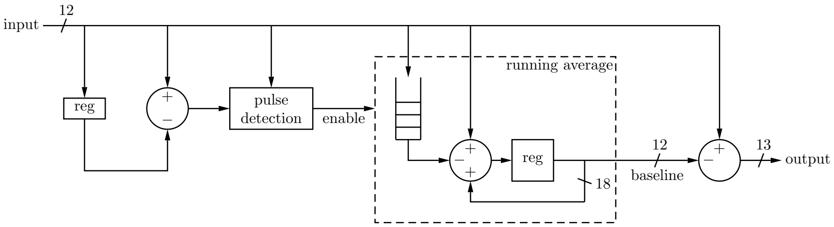


Figure 6.26: Simplified schematic of the baseline estimation and correction engine.

within a given window of significance. The integration window is determined by programmable start and end signal thresholds, such that integration begins when the input signal exceeds the first threshold and ends when it goes below the second one or the pulse window ends, whichever happens first. Instead of considering only the samples starting at trigger detection, up to three previous samples are taken into account additionally; this is particularly important for charge estimation, as the trigger condition usually takes place during the rising edge of the pulse signal. The charge signal generated by the module is a 12-bit value corresponding to the instantaneous accumulated charge, divided by a fixed but programmable shift value in order to downscale it into 12 bits.

Timestamp estimation

For timestamp assignment, the DCFD method is implemented as described in §4.3.3. This consists in generating the bipolar signal (cf. (4.40))

$$b[n] = y[n] - A \cdot y[n-k] \tag{6.36}$$

where $y[n]$ is either the trigger signal after baseline correction, or its integrated value as given by the charge output of the amplitude and charge detection block; these options correspond to digital implementations of the CFD and ARC methods, respectively. Parameters A and k are configurable: k is an integer number of clock cycle delays ranging from 1 to 16, and amplitude A can be selected between 1 and 4 in steps of 0.25. The resulting signal $b[n]$ is then used for timestamping by determining its zero crossing point.

The algorithm for estimation of the zero crossing of signal $b[n]$, which always happens on the negative direction i.e. going from positive to negative, consists in determining the two samples $b[N]$ and $b[N+1]$ where the transition occurs i.e. the smallest value of N after pulse start such that $b[N+1] < 0$, and then interpolating the signal $b[n]$ linearly in the interval between samples, taking the intersection with zero as the subevent timestamp. In clock cycle units and using the sample number as the time reference, the estimated time mark has integer part N and fractional part given by

$$\frac{b\left[N\right]}{b\left[N\right]-b\left[N+1\right]}. \tag{6.37}$$

Hence, the assigned timestamp is the sum of the local time reference value (i.e. of the timestamp counter) at the time of sample $b\left[N\right]$ plus (6.37).

Direct computation of (6.37) requires instantiating a divisor core since the denominator is variable. A different approach has been taken in this case, obtaining the fractional part iteratively using the well known bisection algorithm ([400], p. 301). In its general form, this is a numeric method for estimation of zeroes of a continuous function f within an interval $[a, b]$ such that $f\left(a\right)$ and $f\left(b\right)$ have different signs. The idea is to evaluate f at $c = \frac{1}{2}\left(a+b\right)$ i.e. the midpoint of $[a, b]$ and then look at its sign: if $f\left(a\right)$ and $f\left(c\right)$ have different signs, then $[a, c]$ must contain a zero; if not, then $f\left(c\right)$ and $f\left(b\right)$ do, so there is a zero in $[c, b]$. The procedure is repeated with the new interval (either $[a, c]$ or $[c, b]$), which has half the size of the original one, until sufficient precision is attained.

Particularization of the bisection algorithm to this setting results in an extremely hardware-efficient implementation. Because the function f is a linear interpolation, evaluation at the midpoint of an interval only requires knowing the value of f at the extremes as

$$f\left(\frac{a+b}{2}\right) = \frac{f\left(a\right)+f\left(b\right)}{2} \tag{6.38}$$

which only requires an adder for computation. Moreover, the sign of (6.38) directly gives the most significant fractional bit of the fixed point binary representation of the zero in units of $b - a$ as it indicates in which half of $[a, b]$ it lies. Hence, iterations only require keeping track of the last value of (6.38) and storing the signs of each value until the desired amount of fractional bits are computed. The implementation is outlined in Fig. 6.27, and only requires two registers and an adder, plus a shift register that stores the estimation. It computes one fractional bit per clock cycle until 12 bits are obtained, and requires a small FSM for control and initialization.

6.2.6 Coincidence module

A general module for coincidence detection would host one FIFO queue for each possible detector source where subevents are received in chronological order, followed by coincidence logic that compares the timestamps of the first i.e. earliest available subevent from each detector and checks whether the two oldest ones are within the time coincidence window by computing their timestamp difference and comparing it against configurable minimum and maximum threshold values. Both subevents would then either be registered as a coincidence, or else the oldest one would be discarded as a single event.

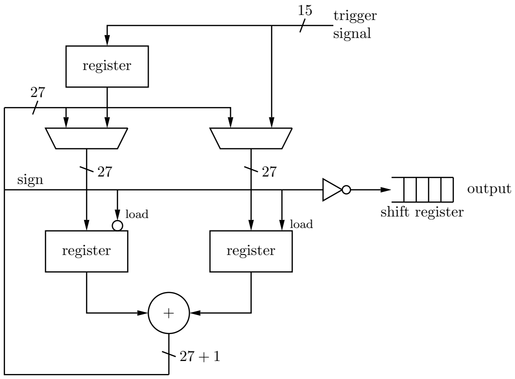


Figure 6.27: Hardware for the implementation of the lightweight bisection algorithm that estimates the zero crossing by linear interpolation.

Because only a two-detector setup is supported by the designed firmware, the coincidence module for event acceptance can be greatly simplified. Only two queues are required, one for each possible source of subevents: the first is always a local subevent builder, whereas the other one can be configured as either the other local subevent detector or the stream of subevents received by the IML from the other acquisition board. No logic for eligibility of valid pairs is needed because there is only one possible detector pair, and so the difference between the 48-bit timestamps in the visible subevents from both queues is the only required quantity for coincidence determination. In case of coincidence, both 160 bit subevent frames with the format from Fig. 6.25 are copied to the event FIFO for readout from the system controller. Support for singles-mode operation has also been included in the module, so that it can be configured to accept all singles i.e. non-coincident subevents from either source or from both of them.

6.2.7 Inter-Module Link

The IML block essentially consists in an adaptation of the firmware in the proof-of-principle implementation from §3.5 to the Altera environment and an expansion into a full, more complex communication protocol between DAQ boards. Its block diagram is outlined in Fig. 6.28 and is very similar to that from Fig. §3.23, with the addition of several queues and related control signals (not shown in the figure) for the transmission and reception of several kinds of data besides the synchronization frames themselves, which only make up a small fraction of the available link bandwidth.

As argued in §3.5.1, there must be at least one clock domain crossing signal, used as a “new frame received” flag, whose timing is critical to correct synchronization. This signal is again highlighted in blue, and its reset circuitry also carries its own

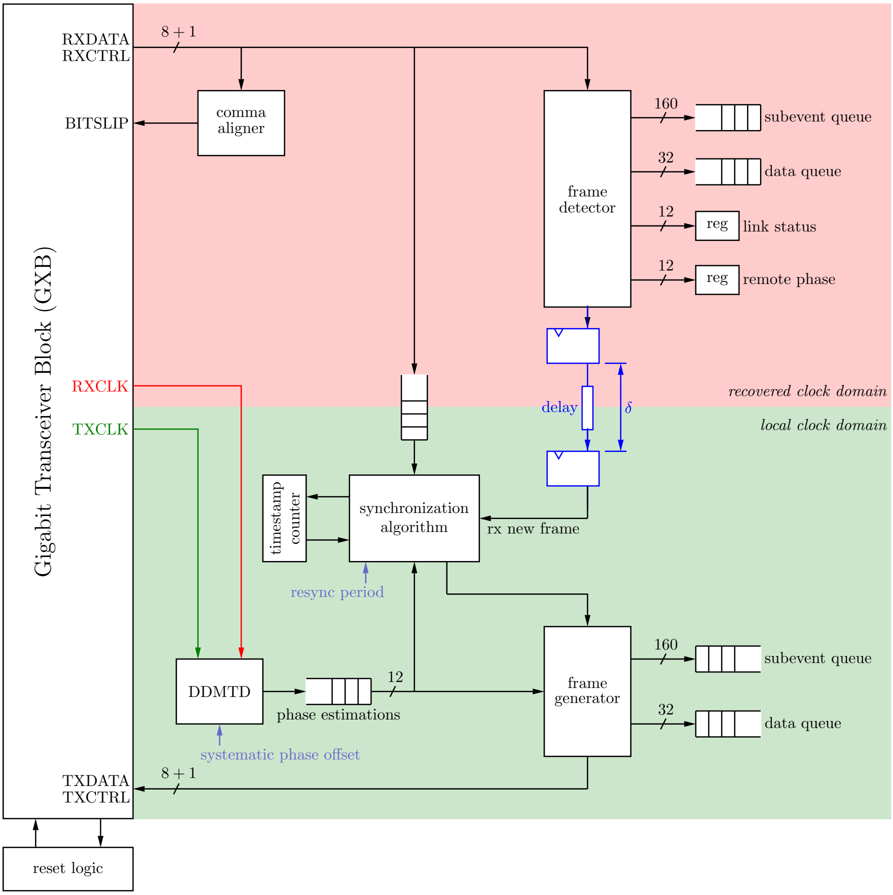


Figure 6.28: Block diagram of the Inter-Module Link, including transceiver management, communication protocol, synchronization algorithm, the local time reference, and data interfaces. Both clock domains are shown. The critical clock-domain-crossing path is highlighted in blue, and configuration registers are shown in light blue.

synchronizer in order to avoid recovery errors [401]. The content of synchronization frames is streamlined into a clock domain crossing FIFO queue with capacity for exactly one frame that is read and decoded by the synchronization algorithm block instead of the frame detector.

The frame generator is responsible for automatically embedding the departure timestamps into synchronization frames, and the domain-crossing “new frame” signal is used to latch the arrival timestamps for deterministic behavior. In order for this not to introduce errors in the synchronization procedure, it is necessary to ensure that its propagation delay δ is either always higher or always lower than the phase difference $\Delta\varphi$ between the local and recovered clocks. This was resolved in §3.5 by designing for a low value of δ and then forcing an increased safety margin for $\Delta\varphi$ in the form of a guard interval around zero. The opposite approach is followed here: the delay is increased by manually inserting logic cells in the path until it is approximately equal to T_{clk}, so that the condition $\Delta\varphi < \delta$ can be embedded into the guard interval for stability without increasing the range of forbidden phase values. It should be noted that the variations are only expected in $\Delta\varphi_{\text{M}}$ while $\Delta\varphi_{\text{S}}$ is essentially fixed for a given board and firmware version, and determined by the total delay in the path from recovered transceiver clock to reference transceiver clock in Fig. 6.4 passing through the external PLL; it is only possible to modify it by configuring the delay of the associated clock output of the LMK02000.

The measurement of $\Delta\varphi$ is based on DDMTD with the same parameters as in the Xilinx implementation, i.e. a stretching factor $N = 512$ implemented using two cascaded PLLs with frequency multiplication factors 32/27 and 16/19 respectively, clustering with learning rate $\theta = 1/256$, and 12 bits for each estimation. Similarly, 48-bit timestamps are used with 12 fractional bits.

On a first approximation, systematic errors in the estimation of $\Delta\varphi$ apparently do not impact resolution because they are canceled in the computation of the link skew as (3.55) if they are equal in both ends. However, ignoring them carries two important drawbacks. On one hand, skew-less estimation is needed in order to make sure that $\Delta\varphi < \delta$. On the other hand, the assumption that systematic errors are matched in both link ends is true in this particular two-board setup but not necessarily so between arbitrary acquisition and concentration boards in a fully developed DAQ architecture. Accurate measurement without systematic errors is a tricky issue in the Stratix FPGA due to the difficulty of forcing equal propagation delays of the transmitter and recovered clock signals to their sampling flip-flops in the DDMTD core.[11] The followed approach is to estimate the systematic error

[11] This problem is determined by the difficulty of instantiating intermediate clock buffers with controlled placement and routing with Altera devices. The Xilinx environment offers much more flexibility in this aspect.

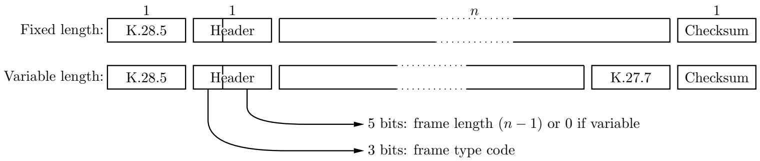


Figure 6.29: Generic formats for fixed (top) and variable (bottom) length frames in the IML communication protocol.

after the place-and-route process instead as

$$\Delta\varphi_{\text{offset}} = (\delta_{\text{clk,RXCLK}} - \delta_{\text{clk,TXCLK}}) + (\delta_{\text{data,RXCLK}} - \delta_{\text{data,TXCLK}}) \qquad (6.39)$$

where the δ values are relative delays to the clock and data inputs of the DDMTD sampling registers, provided by the Altera timing analysis tools, and then use a configuration register to make its value available to the DDMTD engine so that it can be subtracted from estimations. Notice that the value of (6.39) cannot be hardwired in the HDL code because it needs to be obtained *after* synthesis.

Communication protocol

Communication over the IML is based on parallel 8-bit words i.e. bytes at a 156.25 MHz frequency. The 8B/10B line code is used for physical transmission, making it possible to use some of the additional available control characters for frame formatting and control. The generic frame formats are described in Fig. 6.29 and allow for both fixed and variable length, depending on the type of data being sent over the link. Each frame starts with a K.28.5 comma character, which is also used for word alignment during initialization. A header byte comes next, with a 3-bit code indicating the type of frame and the rest containing frame length information. The frame payload comes afterward, consisting of either a fixed or a variable number of bytes; in the latter case, the control character K.27.7 is inserted at the end in order to signal the end of the frame. Finally, a 1-byte checksum is included that is equal to the XOR sum of all other data bytes within the frame, i.e. including the header but neither the start-of-frame nor end-of-frame control characters.

Each frame type code corresponds to a different virtual channel over the IML, and the frame payloads are distributed along separate data paths within the hardware as shown in Fig. 6.28. The available frame types are described in Table 6.5, and are assigned codes according to their priority in the sense that, whenever there are multiple requests for data transmission from different virtual channels, the IML logic always serves the one with the highest priority. Five possible frame types are available in the designed firmware, and two more are defined but currently unimplemented:

Priority	Code	Frame type
Highest	111	Hardware alert *(unimplemented)*
	110	Synchronization frame
	101	Phase measurement frame
	100	Reserved
	011	Data transfer between control processors
	010	Oscilloscope *(unimplemented)*
	001	Subevent frame
Lowest	000	Link establishment frame

Table 6.5: Description of IML frame types and their associated type field code and priority. Types 010 and 111 are not implemented in the current version.

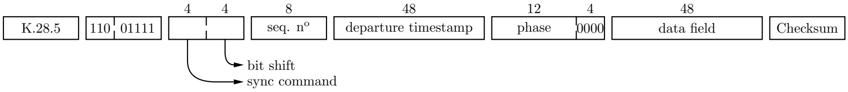


Figure 6.30: Format of pairwise synchronization frames.

- *Hardware alert:* The highest priority is reserved for alert messages generated by critical events or hardware failures such as communication loss with ADCs or other modules, that are intended to generate interrupts remotely on processors at different boards. It has not been implemented in the first version.

- *Synchronization:* Frames corresponding to the pairwise synchronization procedure are issued with a programmable period T_{resync} by the synchronization algorithm controller, with the format shown in Fig. 6.30.

- *Phase measurement:* If configured to do so, the slave module sends a frame to the master whenever the DDMTD logic generates a new phase estimate. This allows statistical analysis of the set of phase estimations and is intended for monitoring and debugging purposes. The frame format is very simple, depicted in Fig. 6.31.

- *Data transfer:* This virtual channel directly connects the Avalon buses of the master and slave modules by way of transmit and receive FIFOs in each of them. Frame length is variable, and 32-bit words keep being transmitted as long as the transmit queue has data in it, up to a maximum of 256 words i.e. 1024 bytes.

- *Oscilloscope:* This channel is intended to support the fast readout of a slave module's oscilloscope mode by connecting its oscilloscope output to the oscilloscope FIFO in a different board. For simplicity, it has not been implemented in the current version. Readout of the slave oscilloscope is instead implemented by software on the embedded microprocessors, reading the os-

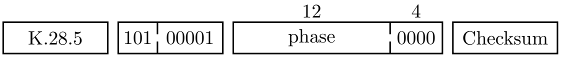


Figure 6.31: Format of phase measurement frames.

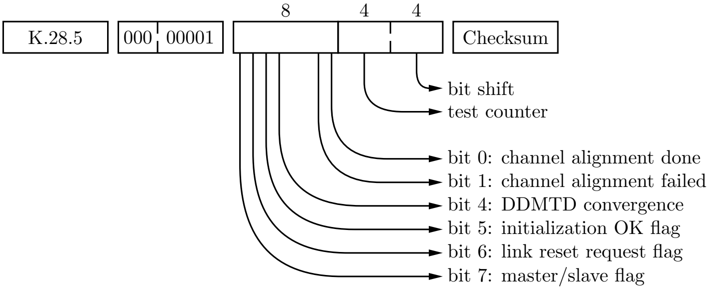

K.28.5
000
00001
8
4
4
Checksum
bit shift
test counter
bit 0: channel alignment done
bit 1: channel alignment failed
bit 4: DDMTD convergence
bit 5: initialization OK flag
bit 6: link reset request flag
bit 7: master/slave flag

Figure 6.32: Format of pre-sync or idle frames.

cilloscope FIFOs manually and sending the waveforms over the data transfer channel, which is a much slower approach.

- *Subevents:* Data from the first subevent detector are sent over the IML with low priority if configured to do so, and stored in a queue in the receiver where they can be used for coincidence detection between two boards. Up to 5 subevent information packets can be sent in each IML frame, each of them 20 bytes long with the format described in Fig. 6.25.

- *Idle frames:* The lowest priority is assigned to short frames containing link status information that serve two purposes. Before link establishment, these are the only frames being transmitted, and the information flags within them are used as handshaking signals by the link initialization procedure. After the link is established, transmission of these frames during idle periods ensures a minimum rate of transitions on the serial links, including comma characters for alignment. The frame format is shown in Fig. 6.32 and includes pre-synchronization data similar to Fig. 3.24 with additional functions such as the ability to request a link reset at the remote node, and a test counter that is incremented for each idle frame and can be used for debugging.

Link establishment

The link establishment procedure is similar to that described in §3.5.1, but is implemented entirely in software running on the Nios II processor using the various flags in pre-synchronization frames to control the process. Both nodes start by configuring the LMK02000 as an oscillator with the appropriate parameters and resetting the transceiver reference clock, receiver and transmitter. Transmitters are then activated, word alignment is repeatedly attempted until achieved, and then each node waits until the other one finishes its own word alignment as indicated in the received frames; on timeout, each node is reset and the procedure is restarted. After that, the master node checks that DDMTD converges to a valid phase and waits for a response from the slave: either an acknowledgement of valid initialization or a reset request. In the meantime, the slave configures the LMK02000 as a PLL, and waits for clock and DDMTD convergence. If something goes wrong, or $\Delta\varphi$ is outside of the guard interval, a link reset request is sent to the master and the procedure is restarted at both ends. If it is not, the valid initialization flag is set. When the master also sets its own valid initialization flag, link establishment is complete.

Experimental tests show that this algorithm reliably results in successful link establishment regardless of the module power-up order and time delay, i.e. no software hang-ups occur (although possibly several retries are needed) as long as appropriate timeout values are used in order to avoid repeated link resets leaving no time for response to the other end.

Synchronization protocol

The synchronization protocol is handled by a dedicated FSM whose state diagram is depicted in Fig. 6.33 in simplified form. It is very similar to the one used for the Xilinx implementation (cf. Fig. 3.25) with a few differences. A single FSM is used instead of two separate ones for master and slave modes, and less states are required because frame transmission and reception are separated from the synchronization algorithm itself. Synchronization frames, described in Fig. 6.30, are shorter but contain essentially the same information. The command field is reduced to a few bits, as there are only three different types of synchronization frames in use. Note that it is necessary to include the phase information because $\Delta\varphi_{\mathrm{S}}$ is needed to compute Δt in this case, unlike the Xilinx implementation where this term is always equal to 0.

An 8-bit sequence number field is included for increased resilience against frame loss due to communication errors across the link. This field is set by the master on the first frame of each pairwise synchronization procedure and copied to subsequent frames. If a second or third frame is received with the wrong sequence number, it is ignored and the procedure is restarted at the receiver end, eventually causing a timeout at the transmitter. The ongoing synchronization step is thus discarded and causes no invalid timestamp counter updates.

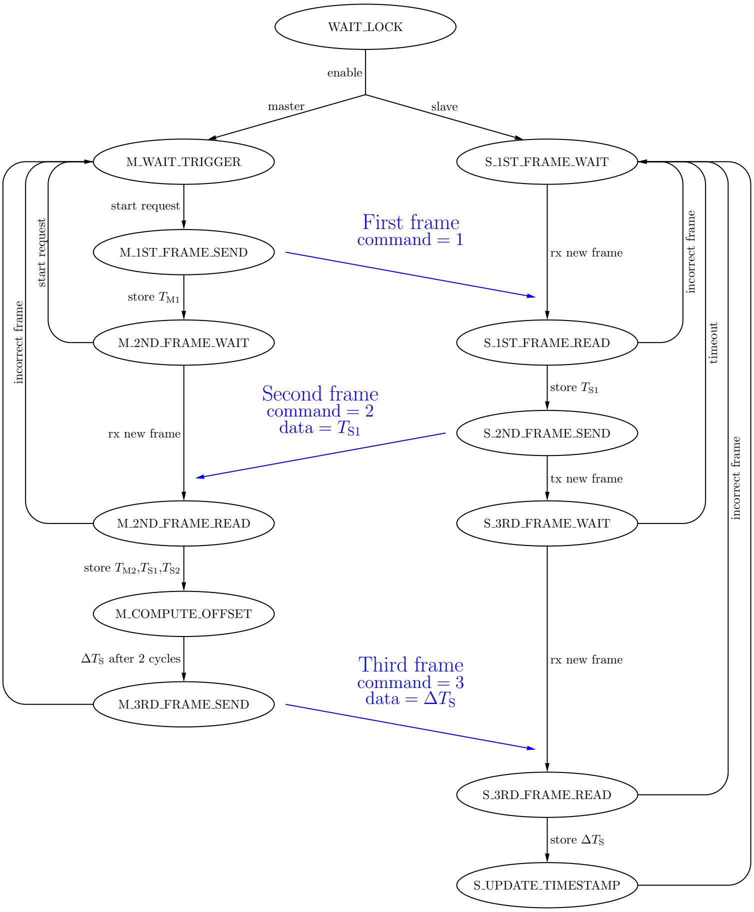


Figure 6.33: State diagram of the synchronization algorithm controller, with master and slave modes on the left and right, respectively, and frame details between nodes highlighted in blue.

6.2.8 Control software

A simple version of the embedded control software has been implemented that provides sufficient performance for the tests presented in this thesis but probably falls short in a setting with higher event rates. After an initialization stage, it consists in an infinite loop where control commands are continually accepted and executed in a blocking way i.e. one command cannot be carried out until the last one has finished. The master module accepts a single incoming Gigabit Ethernet connection where control commands are transmitted, whereas the slave retrieves them and responds to them using the data transfer channel over the IML; the master is responsible for relaying control commands between the Gigabit Ethernet connection and the slave module.

The communication protocol is very simple, with packets based on 32-bit words with the following format:

- The lowest 16 bits of the first 32-bit word identify the specific command. Even numbers are used for commands (i.e. from control PC to DAQ) and odd numbers for their responses (i.e. from DAQ to control PC).
- The highest 16 bits of the first 32-bit word contain the intended destination of the command. In the current implementation, the valid field values are just 0 for the master and 1 for the slave.
- The second 32-bit word contains the length of the packet in 32-bit words, not counting the first two, i.e. the remaining amount of words to be read or sent.
- The rest of the packet has variable length and its content depends on the particular command.

The main commands that have been implemented include:

- *Write and read registers.* For this control protocol, a control and status register map is defined that is completely separate from the hardware register banks available to the processor through the Avalon bus. The processor is responsible for translation between the register numbers on the software layers and their hardware mapping, either in register banks within the SoC or in external devices programmed through an on-board SPI or I^{2}C interface such as the ADCs and RDACs. In this way, a transparent programming interface is achieved for the DAQ system that allows control software on a PC to be independent of hardware changes - these need only be applied in the embedded control software.

- *Oscilloscope capture.* Command parameters include the amount of channels to capture and whether to use manual i.e. forced trigger or not; other parameters such as channel configuration are implemented as control registers instead. The response includes the captured waveforms or an error message if a timeout is exceeded.
- *Histogram capture.* No parameters are needed in this case, as configuration is achieved using control registers.
- *Coincidence capture.* The only parameter for this command is the amount of coincidences to capture, limited to 512. A smaller amount of events is transmitted back in case of timeout.
- Various reinitialization procedures involving ADCs, IMLs, and other acquisition blocks. The embedded control software does not check for operating failures; instead, this monitoring task has been relegated to PC control by reading the appropriate status registers and issuing the corresponding reset command.
- Generic I^{2}C commands. The processor translates between strings of bytes to be sent and received on the external I^{2}C bus and the simple I^{2}C interface that has been included in the SoC. These are used primarily for the configuration of the AMIC front-end boards.

All capture commands are implemented using a polling-based mechanism in which status register banks are continuously read until the corresponding hardware signal is activated, instead of more efficient interruption-based procedures. For this reason, coincidence capture is limited to a small amount of events per command, and it needs to be issued repeatedly for large capture sizes.

Control of the DAQ system on the PC side is achieved using a series of Matlab scripts and functions that have been developed in order to translate higher level commands into a series of these simpler commands available through the Gigabit Ethernet connection. As an example, scripts for continuous oscilloscope visualization on the PC screen consist in write register commands that set up the control parameters of the oscilloscope module and individual channels, followed by an infinite loop where capture commands are repeatedly issued and their response is plotted on screen. Other scripts implement the capture of an arbitrary amount of coincident events, or translate AMIC configuration commands over I^{2}C into the appropriate format. More complex scripts are described in §6.3.3 that implement algorithms for channel adjustment or measurement of detector resolution.

6.3 Experimental results

In this final section, the implemented DAQ system is validated and characterized using a series of test measurements. First, tests are described that involve only the acquisition modules themselves in order to evaluate internal features such as ramp calibration and synchronization resolution. Later, the photodetectors and front-end boards are attached, measurements with a radiation source in the experimental setup are described and the estimated system resolution is reported.

6.3.1 Synchronization subsystem

ADC to FPGA delay compensation

The ADC delay compensation method was evaluated first because this can be done on standalone boards. Running linear regression was implemented with a window length of $M = 64$ samples, and appropriate values of the R_c and C_c components in the analog ramp generator were chosen so that this window covered most of the ramp. Measurements of t_{FF} corresponding to a single session (i.e. a single board, without power cycles or resets) were captured and histogrammed. The resulting histogram was approximately Gaussian with a resolution of $\sigma_1 = 235\,\mathrm{ps}$, far too high for the intended resolution goal.

Under the assumption that the actual value of t_{FF} has a much smaller variation and that such a low resolution is caused by measurement noise, the moving average of the last K measurements was implemented as a delay estimator instead of single measurements, with K varying across powers of two for efficient hardware implementation. The resolution i.e. the standard deviation σ_K of such an estimator is easily shown to be equal to that of the original estimator divided by $K^{1/2}$. The captured measurements were histogrammed, and the resulting resolution values are collected in Table 6.6 for various values of K together with those predicted theoretically from the previous measurement. For low values of K, the histograms are Gaussian and the values match, however for $K \geq 512$ the distribution of estimates becomes skewed and its resolution is worse than predicted. One possible interpretation for this is that the estimator becomes too slow to respond to variations in the signal baseline as the covered time window grows longer. Another one would be quantization noise starting to have a significant impact as σ decreases to the same order of magnitude as the granularity of the measurement (i.e. 1.6 ps). The measured values are consistent with the existence of a constant, independent source of indeterminacy as the quadratic difference

$$\sqrt{\sigma_K^2 - \frac{\sigma_1^2}{K}} \tag{6.40}$$

K	σ_K (measured)	$K^{-1/2}\sigma_1$ (expected)
32	41.9 ps	41.5 ps
64	30.3 ps	29.4 ps
128	22.3 ps	20.8 ps
256	16.9 ps	14.7 ps
512	13.4 ps	10.4 ps
1024	11.1 ps	7.3 ps
2048	9.6 ps	5.2 ps

Table 6.6: Delay estimation resolution as a function of the number K of measurements used for the moving average.

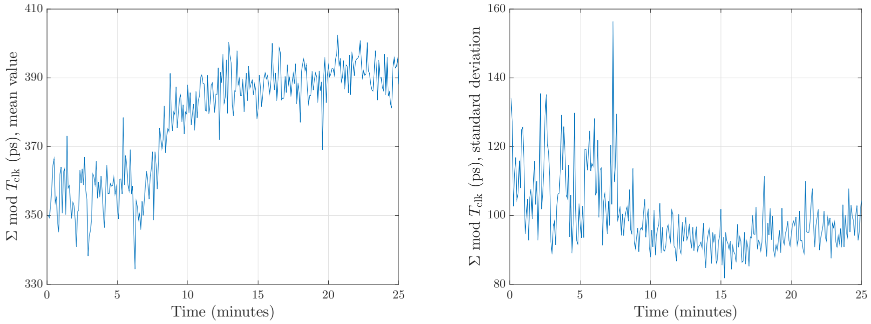
between measurements and predictions is approximately constant and equal to 8.2 ps.

In accordance to these results, the moving average of $K = 256$ measurements was chosen as the final delay estimator. This estimator was monitored for different power cycles and acquisition boards, capturing its instantaneous value during sessions of 5 min. For individual acquisitions, the delay remained almost constant with a worst case i.e. peak-to-peak variation below 20 ps. The average delay exhibited a large variation of up to 40 ns between different captures, although the variation seemed to take place mainly as integer multiples of T_{clk}, presumably due to frame decoding and channel alignment logic, while smaller variations depended on the phase difference between clock distribution nets, which was usually fixed.

Synchronization algorithm

The next evaluation step involved the implementation of module synchronization across the IML and consisted in a repetition of the tests described in §3.5.2 for the Xilinx version, adapted to the new setting. The tests were repeated for all six combinations of the three available boards and possible roles of each one (i.e. master or slave).

Specifically, a first test was done regarding the implementation of phase difference estimation, which used the same parameters as the previous version i.e. DDMTD with $N = 512$ and $\theta = 1/256$ for comparison. The measured phase differences were recorded for several power cycles. In all cases, the evolution of phase estimation was observed to present large variations during the first 1 s to 2 s after power-up and then slowly stabilize until convergence was reached within 10 s to 15 s. The slow response property of the clustering algorithm cannot be blamed for this behavior, as disabling it only reduced the convergence time to 8 s to 12 s; the evolution of the junction temperature within the FPGA and external PLL during power-up and its effect on propagation delays are suspected to be the cause instead.



(a) Mean value over 5 s.

(b) Variation over 5 s.

Figure 6.34: Evolution of Σ from (6.43) over periods of 5 s. As the air conditioning is turned off and temperature falls, there is a latency shift of about 40 ps and resolution improves from 110 ps to 90 ps.

	Master board	Slave board
Random jitter	1.05 ps	1.11 ps
Deterministic jitter	190 fs	690 fs
Total jitter	14.9 ps	16.2 ps

Table 6.7: Jitter measurements on the local clocks at both nodes of an Inter-Module Link.

After convergence, the mean phase value was seen to remain approximately constant, with a very slow drift attributed to temperature variations; in particular, long captures revealed an appreciable difference in $\Delta\varphi_{\mathrm{M}} + \Delta\varphi_{\mathrm{S}}$ between daytime and nighttime of about 40 ps with relatively sharp steps that are assumed to correspond to the air conditioning in the EDNA facilities switching on and off, as shown in Fig. 6.34. Histograms of 5 min captures feature a single peak with a resolution of $\sigma_{\Delta\varphi} = 49\,\mathrm{ps}$. The quality of the local clock was also measured in master and slave boards from the test clock output shown in Fig. 6.5 using a Tektronix DSA70404C signal analyzer, and random jitter was found to remain below 1.2 ps in all cases, within the order of magnitude of the design estimation from §6.1.1. Specific values are summarized in Table 6.7.

The next test involved the synchronization algorithm itself. A resynchronization rate of 10 Hz was used for the tests, with guard intervals of $[-640\,\mathrm{ps}, 640\,\mathrm{ps}]$. As in the test with the Xilinx boards, pins in the expansion header were configured to output pulses with a width of $8\,T_{\mathrm{clk}} \approx 50\,\mathrm{ns}$ whenever the local integer timestamps fell within a certain fixed range with a frequency around 150 Hz, which were then monitored using a Tektronix TDS3014C oscilloscope in order to measure their time difference. After every reset or power cycle, the algorithm converged in a single iteration, resulting in pulses at the oscilloscope becoming aligned with a delay below T_{clk} and their relative differences varying in a very small range. This variation was evaluated by capturing and histogramming the fractional part of the timestamps over periods of 5 min, obtaining a resolution of $\sigma_{\mathrm{TS}} = 51\,\mathrm{ps}$.

Unfortunately, the time difference between oscilloscope pulses did not correspond exactly to the fractional part of the slave timestamp, but presented static offsets of up to 700 ps, which were nevertheless approximately constant for a fixed setup. Two possible causes were identified for this mismatch:

- It may be the case that the models for flight time between the master and slave nodes developed in §3.3 and §3.4.1 are incorrect or incomplete, and there are additional unidentified fixed latency components. Because all elements in the propagation path are accounted for, the only possible source of indeterminacy lies in the FPGA transceivers, which may be introducing fixed latencies that vary from device to device, contrary to specifications.
- Another possibility is that the oscilloscope measurement introduces a setup-dependent systematic error. This may be caused by differences in the propagation delay of probes, or by the waveform differences between the digital output pulses from each board, captured in an uncontrolled environment and with no regard for impedance matching. In particular, the difficulty of defining a proper time mark in the pulses may result in poor estimations of the delay between them.

Of course, both may be true simultaneously. In any case, ruling out the first possibility is of prime importance, with the goal of validating the theoretical modeling presented in this dissertation. A series of additional tests was conducted in order to do so. The reasoning behind these new tests is as follows:

Assume, for contradiction, that the oscilloscope measurement introduces no systematic error, or that it is negligible compared to the actual deviation from the theoretical model. Up to this point, it has been accepted that transceiver latencies are such that

$$\begin{aligned} t_{\mathrm{TX}} &= C_{\mathrm{TX}} \\ t_{\mathrm{RX}} &= C_{\mathrm{RX}} + n\frac{T_{\mathrm{clk}}}{10} \end{aligned} \tag{6.41}$$

where C_{TX} and C_{RX} are constant, not just for a given transceiver across power cycles, but also across different devices or transceivers within the same FPGA. Hence, assume that this is not necessarily the case. Adding (3.33) and (3.34), the round-trip time is obtained as

$$\begin{aligned} RTT = {} & L_{\mathrm{TX,M}} + L_{\mathrm{TX,S}} + L_{\mathrm{RX,M}} + L_{\mathrm{RX,S}} + t_{\mathrm{p,MS}} + t_{\mathrm{p,SM}} + \\ & + C_{\mathrm{TX,M}} + C_{\mathrm{TX,S}} + C_{\mathrm{RX,M}} + C_{\mathrm{RX,S}} + \Sigma \end{aligned} \tag{6.42}$$

where the term

$$\Sigma = \frac{n_{\mathrm{M}} + n_{\mathrm{S}}}{10} T_{\mathrm{clk}} + \frac{\Delta\varphi_{\mathrm{M}} + \Delta\varphi_{\mathrm{S}}}{2\pi} T_{\mathrm{clk}} \tag{6.43}$$

contains the supposedly variable latency components, so it can be monitored and measured experimentally. Module design and cable choice guarantees that $t_{\mathrm{p,MS}} = t_{\mathrm{p,SM}} = t_{\mathrm{p}}$ is constant except for temperature drift. Because *RTT* is always an integer multiple of T_{clk}, taking the modulus results in

$$\Sigma \equiv -C_{\mathrm{TX,M}} - C_{\mathrm{TX,S}} - C_{\mathrm{RX,M}} - C_{\mathrm{RX,S}} - 2t_{\mathrm{p}} \mod T_{\mathrm{clk}}. \tag{6.44}$$

Similarly, link skew may be expressed as

$$\begin{aligned} \Delta t &= (C_{\mathrm{TX,M}} - C_{\mathrm{TX,S}}) + (C_{\mathrm{RX,S}} - C_{\mathrm{RX,M}}) + (t_{\mathrm{p,MS}} - t_{\mathrm{p,SM}}) + \Delta \\ &\approx (C_{\mathrm{TX,M}} - C_{\mathrm{TX,S}}) + (C_{\mathrm{RX,S}} - C_{\mathrm{RX,M}}) + \Delta \end{aligned} \tag{6.45}$$

where

$$\Delta = \frac{n_{\mathrm{S}} - n_{\mathrm{M}}}{10} T_{\mathrm{clk}} + \frac{\Delta\varphi_{\mathrm{S}} - \Delta\varphi_{\mathrm{M}}}{2\pi} T_{\mathrm{clk}} \tag{6.46}$$

is the ideal value of Δt. The error introduced in the measurement of Δt is thus

$$\varepsilon = \Delta t - \Delta = C_{\mathrm{TX,M}} - C_{\mathrm{TX,S}} + C_{\mathrm{RX,S}} - C_{\mathrm{RX,M}} \tag{6.47}$$

and so, by (3.30), an error of $\varepsilon/2$ is introduced by the synchronization algorithm. In particular, if δ_{osc} is the time difference between pulses measured with the oscilloscope, then the assumption of accuracy implies that it is equal to this timestamp error. Hence, $\varepsilon = 2\delta_{\mathrm{osc}}$ can also be obtained experimentally.

Tests were conducted whereby two particular boards out of three available (denote them as A, B and C) were selected as the master and slave, respectively, and power cycled repeatedly. Each time, the values of Σ and δ_{osc} were monitored and captured. For a given power cycle, both values remained approximately constant, with Σ having a resolution around 100 ps. Moreover, their mean values also remained constant across power cycles, with Σ having a very small variation of 14 ps. The results are summarized in Table 6.8. It follows that the values

$$\begin{aligned} 2\delta_{\mathrm{osc}} &= C_{\mathrm{TX,M}} - C_{\mathrm{TX,S}} + C_{\mathrm{RX,S}} - C_{\mathrm{RX,M}} \\ -(\Sigma + 2t_{\mathrm{p}}) \mod T_{\mathrm{clk}} &= C_{\mathrm{TX,M}} + C_{\mathrm{TX,S}} + C_{\mathrm{RX,S}} + C_{\mathrm{RX,M}} \mod T_{\mathrm{clk}} \end{aligned} \tag{6.48}$$

both remain constant across power cycles *for a given set of two boards*. Adding and subtracting both expressions, one obtains that $C_{\mathrm{TX,M}} + C_{\mathrm{RX,S}}$ and $C_{\mathrm{TX,S}} + C_{\mathrm{RX,M}}$ are constant values, i.e. the sum of the variable terms in the transmitter from a board and the receiver from the other board is always constant. It is highly unlikely that these values vary between power cycles in such a coordinated way, especially considering that they belong to different boards; it is far more likely that they do not vary at all. Hence, the conclusion of this test is that C_{TX} and C_{RX} are very probably constant across resets *for a given transceiver*.

Test	Boards used	Iterations	Σ		δ_{osc}	
	(master, slave)		Average	St. dev.	Average	St. dev.
1	(A, B)	23	467 ps	14.2 ps	−53 ps	79 ps
2	(A, B)	18	472 ps	13.9 ps	−61 ps	58 ps
3	(C, B)	9	381 ps	12.6 ps	225 ps	130 ps
4	(C, A)	9	426 ps	14.1 ps	361 ps	56 ps

Table 6.8: Results of the tests for determining whether the latency of FPGA transceivers is indeed deterministic or rather contributes to synchronization errors.

Further tests were carried out in which one of the boards was swapped with the unused board: first the master node changed while the slave node remained the same, and then the other way around. Variations in the order of 50 ps to 100 ps were then observed in the mean value of Σ as indicated in Table 6.8. Now consider the variation of all terms in (6.48) from setup (A, B) to setup (C, B): t_{p} remains constant because the same cables are being used to connect both nodes, and $C_{\text{TX,S}}$, $C_{\text{RX,S}}$ also remain constant as has been established by the previous test, because the slave board $S = B$ does not change. It follows that

$$\begin{aligned}\Sigma_{(A,B)} - \Sigma_{(C,B)} &= (C_{\text{TX,C}} - C_{\text{TX,A}}) + (C_{\text{RX,C}} - C_{\text{RX,A}}) \\ &= (C_{\text{TX,C}} + C_{\text{RX,C}}) - (C_{\text{TX,A}} + C_{\text{RX,A}})\,. \end{aligned} \tag{6.49}$$

Hence, the sum $C_{\text{TX}} + C_{\text{RX}}$ is similar for both master boards. Repeating this test by changing the slave boards instead of the masters gives the same result for boards A and B. It follows that $C_{\text{TX}} + C_{\text{RX}}$ is more or less constant for all three boards with a maximum difference of less than 100 ps between them. Again, it is highly unlikely that there are large variations in C_{TX} and C_{RX} in such a way that their sum remains constant all the way, so it is safe to conclude that they are almost constant from FPGA to FPGA.

As a summary, these tests show with a large degree of certainty that the deviations of FPGA transceiver latency from the ideal theoretical values are comparatively small, and that the large static offsets observed during oscilloscope-based validation of the synchronization scheme are an artifact of the measurement method itself. A more exact method is thus needed to properly evaluate the synchronization and its resolution; such a procedure is described in the next section.

6.3.2 Time coincidence

Instead of relying on an external oscilloscope for the evaluation of inter-module synchronization with all its associated measurement artifacts, one can use the modules as oscilloscopes that yield pulse timestamps for the estimation of time differences. Thus, for the next array of tests, a pulse source was used in such a way that pairs of acquisition channels received simultaneous pulses. Their timestamp

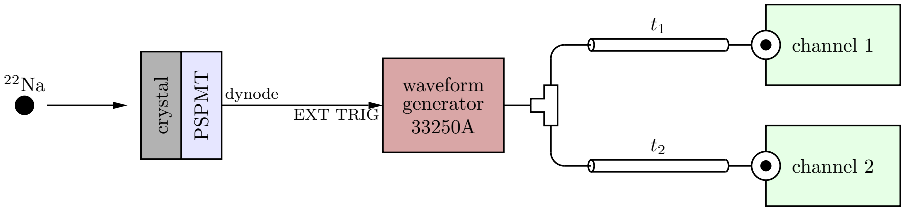


Figure 6.35: Setup used for the time coincidence tests. Pulses with fixed waveform and random separation are fed to different acquisition channels using coaxial cables of matched propagation delays $t_1 \approx t_2$.

difference should ideally be zero, and so the deviations from that value directly give a measure of synchronization error and resolution.

The setup for these tests is described in Fig. 6.35. An Agilent 33250A arbitrary waveform generator was used as a source of pulses with a fixed waveform. The pulses were generated with random separations instead of a fixed period in order to avoid coupling effects between the acquisition and timestamping process and the pulse period. To this end, an AMIC front-end board with a PMT and a ^{22}Na radioactive source was employed to generate randomly distributed photodetector pulses which were then used to drive the external trigger input of the waveform generator; in this way, the random separation between pulses is preserved but the shape of the pulses is made uniform.

Trapezoidal, almost-triangular pulses were generated with a width of 70 ns, an edge time of 43.8 ns, and an amplitude of 1 V. A T connector was used to split the signal into two different paths, where it was distributed to different acquisition channels through coaxial cables of matched length. The detector modules were configured in time coincidence mode. Baseline cancellation was disabled in order to remove its effect on measurements; fixed baseline values were estimated instead as the mean value of the results of 1 ms long captures with the histogram module. Pulses were independently detected and timestamped in each channel using DCFD in amplitude mode with parameters $A = 1$ and $k = 4$ in (6.36). This time delay corresponds to the original and delayed waveforms intersecting at their falling and rising edges, respectively, so that the slope of the difference signal is maximized and time resolution is optimized due to (4.37); this observation was later ratified experimentally. The time difference distribution of at least 10^5 pulses was obtained for each test, and measurements were repeated interchanging the cable connections to both channel in order to account for the possible fixed time bias $t_2 - t_1$ from cable length mismatch.

The mean value μ of the time difference distribution was studied first, both for single-board setups where the simultaneous pulses were distributed to different channels on the same board and for double-board setups. In the latter case, ADC-to-FPGA delay compensation was enabled, and its possible systematic error had an effect on the measure of μ; it was not needed in the single-board case because it results in the same correction offset being added to both timestamps and subsequently cancelled in the time difference computation. Channel 0 in the master board was always used as a reference, and its μ with respect to all other channels in both boards was sequentially obtained. The bias $t_2 - t_1$ from cable connections was found out to be between 19 ps and 21 ps in all cases. The distribution of time differences always presented a Gaussian shape, and different values of μ were obtained for different channels, ranging approximately from -150 ps to 150 ps, and being constant for a given channel even if tests were repeated (although a slight drift with temperature was observed). Moreover, the distribution μ was found out to be independent of the choice between a single-board and a double-board setup. The conclusion is that there are systematic errors in the order of up to 150 ps caused by fixed delay differences in the analog stages in each channel that can be nevertheless corrected with a one-time calibration of individual boards. In particular, neither ADC-to-FPGA compensation nor timestamp synchronization appear to be contributing significantly to systematic error.

Synchronization resolution was obtained next. For this test, μ was disregarded and only the width σ of the distribution was considered. ADC-to-FPGA delay compensation was disabled, in order to remove the effect of its resolution on the resulting measurements; a systematic error is hereby introduced that does not affect variance and can thus be neglected. The time difference distribution was measured first for channels 0 and 1 on the same acquisition board and a resolution $\sigma_1 = 85$ ps was obtained. This value includes the time uncertainty effects from the source and the timing electronics, i.e. predominantly the resolution of the timing algorithm but also electronic noise and jitter at the waveform generator and the acquisition module. Next, the measurement was repeated for channel 0 on the master and channel 1 on the slave board. In this case, a resolution $\sigma_2 = 98$ ps was obtained. Notice that this value includes all of the previous effects plus the variation σ_{sync} due to the timestamp synchronization method, and no further additional effects. The effect of timestamp synchronization can be assumed to be statistically independent of all other factors due to their source, so one can reasonably estimate

$$\sigma_2^2 = \sigma_1^2 + \sigma_{\text{sync}}^2 \tag{6.50}$$

and obtain the synchronization resolution from this formula. With the measured values, an estimation $\sigma_{\text{sync}} = 49$ ps is obtained, corresponding to a FWHM resolution of 115 ps assuming a normal distribution. It is worth noticing that this value is very similar to σ_{TS}, which can be obtained experimentally with a much simpler setup and procedure i.e. just by monitoring timestamp updates in the synchronization algorithm.

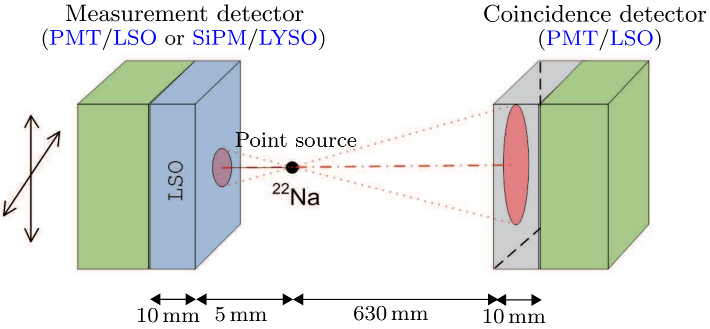


Figure 6.36: Diagram of the detector and source setup used for the tests.

6.3.3 Photodetector evaluation

Finally, the whole DAQ system was evaluated with actual photodetectors and a radioactive source. The experimental setup is outlined in Fig. 6.36 and consists of two photodetectors faced up against each other at a distance of 635 mm, and a ^{22}Na point source located on the line between the centers of the detectors, at 5 mm and 630 mm distance, respectively. Each detector contains a continuous slab of 10 mm deep scintillating crystal covered by black epoxy, with a 49 mm × 49 mm area coupled to a photodetector using optical grease. Two different types of photodetector unit were employed in the tests:

- A Hamamatsu H8500 PSPMT [376]. This detector has 64 outputs and an effective sensitive area matching that of the crystal. Parallelepiped LSO crystals were used with the PSPMTs.
- An array of 16 × 16 Hamamatsu S10362-11-50P SiPM devices [402]. These have an active area of 1 mm × 1 mm but were soldered on a rectangular grid with 3.00 mm × 3.05 mm separation, so that the effective scintillation area is only 10 % of the total. A pyramidal frustum LYSO crystal was used with the SiPM array.

PMT and SiPM detectors were tested on the close side, mounting the detector unit on a translation table controlled by stepper motors in order to align the ^{22}Na source at different, known crystal positions. A PMT detector was placed on the far side in all tests and employed primarily for electronic collimation, by filtering coincidences where the subevent on the far detector fell outside of the center region. In this way, the cone of valid LORs is very narrow at the close detector and restricts events to a particular crystal position.

The AMIC front-end boards were configured to output five significant signals: the four Anger outputs $A = J^{++}$, $B = J^{+-}$, $C = J^{--}$ and $D = J^{-+}$ described in

§4.3.2, that can be used to obtain the centroid using CoG logic, and a fifth signal E consisting of the sum of all photodetector outputs; signal E thus contains the total subevent energy, which can also be inferred from the sum of the four Anger outputs. The dynode signal is also available in PMT detectors as a trigger signal, however the SiPM detector lacks such a dedicated timing signal. Signal E was used instead for trigger, with lower expectations on time resolution since AMIC outputs have a much lower bandwidth and slower rising edges than detector signals. For PMT detectors, both the dynode signal and the energy signal E were tested as trigger signals for comparison.

Automatic channel calibration

The baseline bias point i.e. voltage of each acquisition channel input is initially unknown: it can be different for each power cycle because it evolves over time and, moreover, different detectors can be potentially connected to the channels. An automatic procedure is therefore advisable for estimation of the baseline and adjustment of the programmable DC offset so that the idle voltage is as close as possible to the edge of the ADC input range, in order to take advantage of its full dynamic range. The offsets are determined by the value Ω programmed into the RDAC, which varies discretely between 0 and 255, and the circuit analysis implies that the baseline value is approximately linearly decreasing with Ω along its valid i.e. non-saturated range.

The following adjustment procedure was repeated individually for each active channel:

1. The first step consists in the determination of the valid offset range, which will be of the form $\Omega \in [\Omega_{\text{min}}, \Omega_{\text{max}}]$, such that the channel is saturated above (at 4095) for $\Omega < \Omega_{\text{min}}$ and saturated below (at 0) for $\Omega > \Omega_{\text{max}}$. Saturation is detected by capturing a small waveform window of 64 samples using the oscilloscope module and then checking whether all samples are 0 or 4095. Estimation of Ω_{min} and Ω_{max} is done separately by programming and measuring different offset values following the bisection method, starting at the midpoint $\Omega = 128$ and increasing or decreasing it depending on the captured results. This ensures a fixed amount of capture steps, 15 in total.

2. The RDAC offset step and the electronic noise in the channel are determined next by performing histogram measurements with relatively long capture windows of 1 ms. In the absence of pulse activity and with a fixed baseline, the channel is expected to contain only Gaussian noise, so Gaussian fits can be applied to the histograms that yield mean and standard deviation parameters (μ, σ) equal to the baseline and rms noise in terms of ADC LSBs; even if pulses are present, their short duration implies only small perturbations in the histogram fit assuming that the subevent rate is low enough.

A reference offset value $\Omega_0 = \lfloor (\Omega_{\mathrm{min}} + \Omega_{\mathrm{max}}) / 2 \rfloor$ close to the middle of the valid range is used. More specifically, histograms are captured for the two consecutive offset values $\Omega = \Omega_0$ and $\Omega = \Omega_0 + 1$, with best fit parameters (μ_0, σ_0) and (μ_1, σ_1), respectively. It is then reasonable to assume $\sigma_0 \approx \sigma_1$, so their average $\sigma = (\sigma_0 + \sigma_1) / 2$ is used as an estimation of noise. Because of the linear relationship between Ω and the baseline μ, the baseline difference $\Delta\mu = \mu_1 - \mu_0$ is used as an estimation of the RDAC offset step, such that the baseline $\mu(\Omega)$ corresponding to a programmed offset value Ω will be approximately equal to

$$\mu(\Omega) = \mu_0 + (\Omega - \Omega_0) \cdot \Delta\mu. \tag{6.51}$$

3. As a third step, the obtained values are double-checked for errors that may have been introduced by the simple measurement process; for example, invalid values of Ω_{min} or Ω_{max} can be derived if the oscilloscope captures used in their determination take place during a pulse waveform. A simple test is applied, consisting simply in computing

$$|\Delta\mu| \cdot (\Omega_{\mathrm{max}} - \Omega_{\mathrm{min}} + 1), \tag{6.52}$$

which corresponds roughly to the valid ADC input range in terms of LSBs, and checking whether it is close to the ideal value of 4096.

4. The final step consists in determining the optimal offset value and trigger detection thresholds. The design criteria for these values are given by the probabilities that they result in unwanted behavior i.e. channel saturation or false triggers due to noise. By assuming Gaussian noise, these probabilities can be expressed as variation intervals of fixed length in terms of σ, in the sense that the probability p that a given sample during a pulse-less time window is above the baseline and differs from it by more than $N \cdot \sigma$ is

$$p = \frac{1}{2}\left(1 - \mathrm{erf}\left(\frac{N}{\sqrt{2}}\right)\right). \tag{6.53}$$

This yields, for example, the probability of having a false trigger occur at any given time if the trigger threshold is set at $N \cdot \sigma$ above the baseline. Table 6.9 summarizes the resulting probabilities and approximate false trigger rates in terms of N for the acquisition module.[12] Trigger thresholds for the trigger channel are consequently programmed as $\lfloor N \cdot \sigma \rfloor$ for certain fixed values of N that are obtained experimentally and depend on the particular detector (PMT or SiPM).

In each energy channel, the start and end thresholds for baseline cancellation and charge detection are also configured as appropriate multiples of σ,

[12] Note that N is the same parameter that appeared previously in (3.2) describing total jitter.

N	p	Rate of false triggers (per sample)	 (absolute)
3	1.4×10^{-3}	Once every 740	Once every 4.7 µs
4	3.2×10^{-5}	Once every 32000	Once every 200 µs
5	2.3×10^{-7}	Once every 3.5×10^{6}	Once every 22 ms
6	9.9×10^{-10}	Once every 1×10^{9}	Once every 6.5 s
7	1.3×10^{-12}	Once every 8×10^{11}	Once every 83 min

Table 6.9: False trigger rates as a function of the parameter N in the trigger threshold.

since their performance is assumed to depend only on noise in the channel. Specifically, $N = 6$ and $N = 4$ are used for the baseline cancellation start and end thresholds, respectively, and $N = 8$ for both charge detection thresholds. For the offset value, the channel baseline samples are required to stay above a certain level $u > 0$ with probability $1 - p$; the value of u represents a margin of error that accounts for possible undershoot in pulse waveforms. The condition is thus $\mu(\Omega) - N\sigma \geq u$ which yields the optimal offset value

$$\Omega = \Omega_0 + \left\lfloor \frac{N\sigma + u - \mu_0}{\Delta\mu} \right\rfloor . \tag{6.54}$$

A value of $N = 6$ has been used for all tests.

Energy and time resolution

Energy and time resolution were measured only at the center point of the close crystal, i.e. aligning the center points of both scintillating crystals with the ^{22}Na source. The results of the automatic channel calibration procedure described above was employed to determine the optimal trigger threshold values, using $N = 30$ for PMTs and $N = 6$ for SiPMs; such a high value could be safely employed for PMT detectors because of the very low noise levels. All energy signals (A, B, C, D and E) were treated as charge signals in the case of PMT detectors and amplitude signals in SiPM detectors, and the dynode signal was always processed as an amplitude signal.

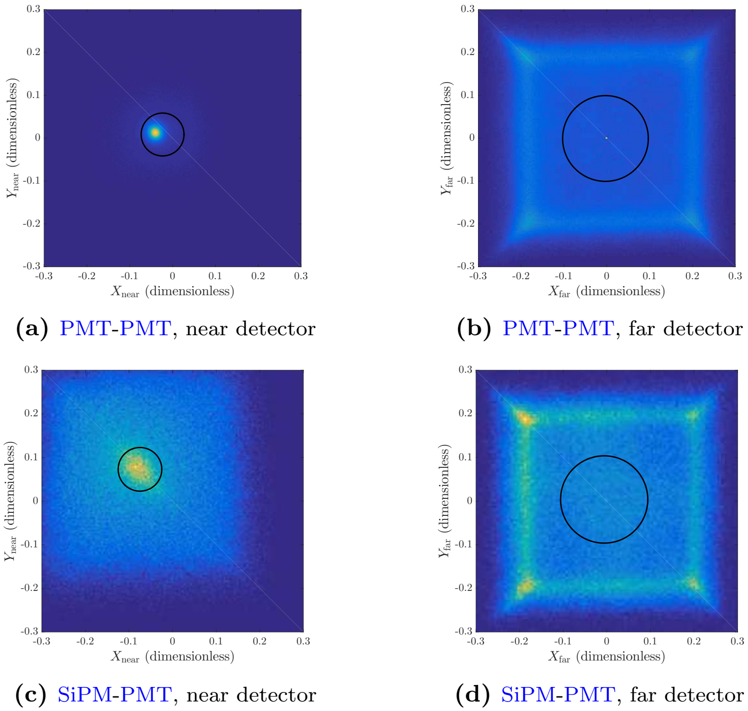
A total of three tests were performed: two for PMT-PMT coincidence, using the dynode signals and the energy signals for timing, respectively, and one for SiPM-PMT coincidence. Timestamping was performed using DCFD in amplitude mode with parameters $A = 2$ and $k = 1$ in (6.36) and pulse windows of 48 cycles i.e. approximately 300 ns. Very wide coincidence windows of ±50 ns were used for event detection in order to capture and study the random event background in addition to true coincidences. Events where any ADC channel reached its full scale value were considered saturated and filtered away. A total of 7.5×10^{6} and 5×10^{6} events were captured for PMT-PMT, respectively, and 5×10^{5} for SiPM-PMT; such a lower amount is due to the very high noise levels in the SiPM detectors, that forces high trigger thresholds and results in a much lower accepted coincidence rate and longer measurement times.

For each set of acquired events, a fully automatic and deterministic process was followed in order to determine appropriate energy and coincidence windows and to estimate the associated resolution values. The following filtering and processing steps were applied using Matlab:

1. *Electronic collimation:* For each event, its centroids in the near and far detectors $(X_{\text{near}}, Y_{\text{near}})$, $(X_{\text{far}}, Y_{\text{far}})$ were obtained using Anger logic as (cf. (4.22))
$$X = \frac{A+B-C-D}{A+B+C+D}\ , Y = \frac{A-B-C+D}{A+B+C+D}. \tag{6.55}$$
In each detector, the coordinates corresponding to the center position were estimated as the average values $(E[X], E[Y])$ over all valid acquired events; these coordinates should ideally be zero, but this procedure accounts for small systematic errors in the measurement. Finally, all events were filtered out except those satisfying both conditions
$$\begin{gathered}(X_{\text{far}} - E[X_{\text{far}}])^2 + (Y_{\text{far}} - E[Y_{\text{far}}])^2 < d_{\text{far}}^2 \\ (X_{\text{near}} - E[X_{\text{near}}])^2 + (Y_{\text{near}} - E[Y_{\text{near}}])^2 < d_{\text{near}}^2\end{gathered} \tag{6.56}$$
i.e. those whose centroids are within a given distance of the estimated detector center. While Anger logic introduces major distortions in the estimation of positions near the crystal borders, it is known to be accurate near the center and so the procedure is reasonable for small values of d_{far} and d_{near}. It was observed experimentally that histograms of X and Y presented peaks and saturated around 0.2, and the values $d_{\text{near}} = 0.05$ and $d_{\text{far}} = 0.1$ were chosen.

   Flood histograms for both types of detectors are shown in Fig. 6.37, together with the region of accepted events. The main collimation effect is determined by the far centroid. For PMT-PMT, the near centroid filtering is auxiliary and only filters a few random and scattered events. However, for SiPM-PMT coincidence, the dispersion of the centroids measured at the SiPM detector is much higher due to the vastly increased noise, so this step results in more exhaustive filtering.

2. *Coarse windowing:* For the set of collimated events, coarse energy and coincidence windows were determined around the peaks in the corresponding histograms by fitting Gaussian shapes around them. The resulting coarse energy and coincidence windows were not applied to filter collimated data but only computed to assist in later steps. It must be noted that this procedure only provides approximate windows due to the effect of events outside



(a) PMT-PMT, near detector

(b) PMT-PMT, far detector

(c) SiPM-PMT, near detector

(d) SiPM-PMT, far detector

Figure 6.37: Flood histogram of captured events at the near and far detectors for PMT-PMT and SiPM-PMT coincidence, with the acceptance region at the center highlighted in black.

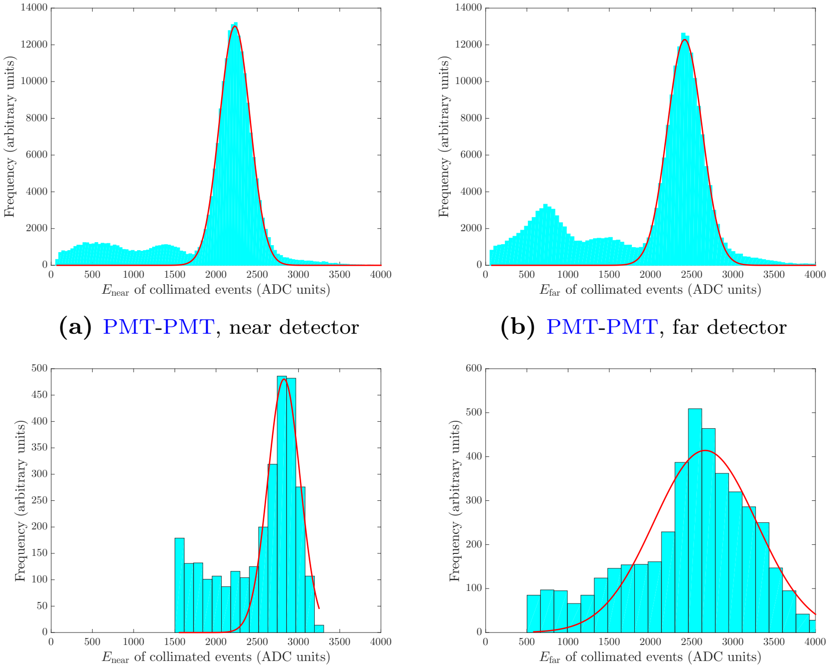
the peak on the fitting results. For instance, Fig. 6.38 shows the collimated energy histograms for both types of detector. For PMT-PMT coincidence, two peaks can be clearly observed at low energies related to Compton scattered events, particularly for the far detector, that cause a slight distortion on the best Gaussian fit. For SiPM-PMT coincidence, these peaks are partially masked by the higher trigger threshold, which introduces a preliminary, coarser energy windowing effect by itself.

For the energy in both the near and the far detector, a standard Gaussian function was fit to the collimated histogram, obtaining its mean value μ and variance σ^2. A window at the FWTM of the Gaussian peak was chosen, i.e. the interval

$$[\mu - N_{\text{FWTM}}/2 \cdot \sigma, \mu + N_{\text{FWTM}}/2 \cdot \sigma] \tag{6.57}$$

where

$$N_{\text{FWTM}} = 2\sqrt{2 \log 10} \approx 4.29 \tag{6.58}$$



(a) PMT-PMT, near detector

(b) PMT-PMT, far detector

(c) SiPM-PMT, near detector

(d) SiPM-PMT, far detector

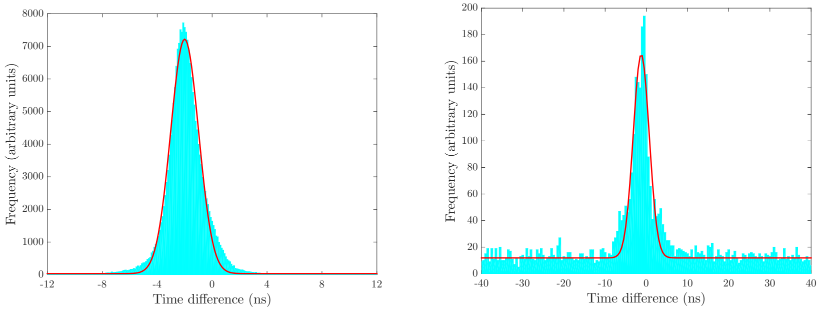
Figure 6.38: Energy histograms of electronically collimated events at the near and far detectors for PMT-PMT and SiPM-PMT coincidence, with the best Gaussian fit superimposed in red.

is such that the value of the Gaussian at the extremes of the interval is 10 % of its amplitude.

For coincidence windowing, the fitting step was applied for a modified Gaussian function with a positive pedestal

$$g_p\left(x\right) = b + a \exp\left(-\frac{\left(x-\mu\right)^2}{2\sigma^2}\right) \quad , \ a, b > 0 \qquad (6.59)$$

in order to account for random coincidences on the time difference histogram, since they are distributed uniformly. As can be observed in Fig. 6.39 where the histograms and their best fits are displayed, random coincidences can be largely ignored for the PMT-PMT case but they have a considerable effect on measurements involving SiPMs; the figure also shows that the Gaussian with pedestal is a indeed a good approximation. As with energy, the FWTM interval (6.57) was chosen as a coarse coincidence window; note that this



(a) PMT-PMT coincidence

(b) PMT-SiPM coincidence

Figure 6.39: Time difference histograms of electronically collimated events for coincidence with PMT and SiPM detectors, with the best Gaussian fit superimposed in red.

interval is strictly smaller than the one resulting of a Gaussian fit without a pedestal.

3. *Coincidence windowing and resolution:* For the determination of the fine coincidence window, the Gaussian fit with pedestal was applied to the set of collimated events after filtering them with the coarse energy windows on the near and far detector, instead of the whole set of collimated events. This is meant to provide a more accurate fit to actual unscattered events. As usual, a coincidence window was chosen as the FWTM of the Gaussian component. The events were then filtered further using this coincidence window, and a *second* Gaussian fit with pedestal was applied to the resulting data; this step slightly enhances the accuracy of the fit by avoiding the contribution of events that have already been rejected as invalid. The final sets of filtered events and corresponding fit are shown in Fig. 6.40, where it can be observed that the rate of random events, determined by the Gaussian-to-pedestal ratio b/a in (6.59), has decreased considerably for PMT-SiPM coincidence. The fine coincidence window is defined in terms of this final fit as its FWHM instead of the FWTM i.e. as

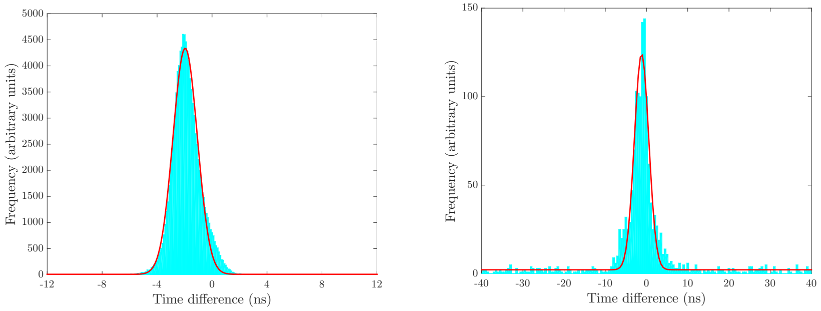
$$[\mu - N_{\text{FWHM}}/2 \cdot \sigma, \mu + N_{\text{FWHM}}/2 \cdot \sigma] \tag{6.60}$$

where

$$N_{\text{FWHM}} = 2\sqrt{2\log 2} \approx 2.35 \tag{6.61}$$

and the coincidence resolution is estimated as its length $N_{\text{FWHM}} \cdot \sigma$ as is customary. The final resolution values are shown in Table 6.10.

4. *Energy windowing and resolution:* For energy resolution, an analogous procedure to that of coincidence resolution was followed. Collimated events were filtered with the coarse coincidence and far energy windows, and a Gaussian



(a) PMT-PMT coincidence

(b) PMT-SiPM coincidence

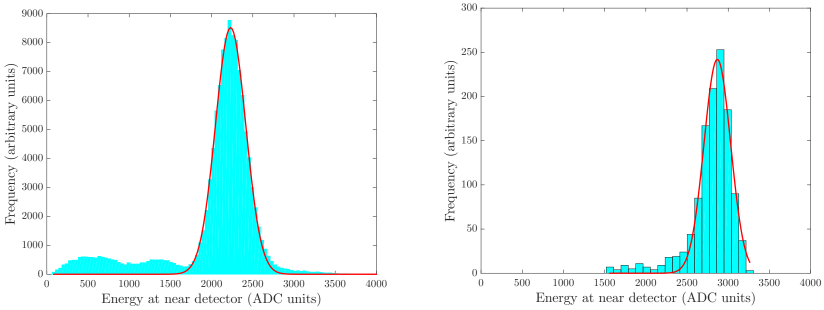
Figure 6.40: Fine time difference histograms used for the determination of coincidence resolution with PMT and SiPM detectors, with the best Gaussian fit superimposed in red.

fit was applied to the resulting histogram of near energy values. Its FWTM was used as a first estimation of the near energy window, and then a second, more accurate Gaussian fit was computed for the filtered events. As with coincidence, the near energy window was defined as the FWHM of the resulting fit and the energy resolution as its length. The energy resolution values were computed only for the near detector and are summarized in Table 6.10, while the final fits are shown in Fig. 6.41.

It must be noted that the energy and coincidence windows obtained with the previous procedure would be implemented online within the event building firmware in the FPGAs in a final DAQ implementation. The situation presented here does not correspond to real working conditions but rather to the experimental characterization of the detectors and the acquisition system.

Final resolution estimates are collected in Table 6.10, together with the total amount of events before and after filtering with the energy and coincidence windows determined by offline processing. The results for both PMT-PMT tests are very similar, with coincidence resolution begin about 8.5 % worse when the energy signal is used for timing instead of the dynode; this suggests that signal E is a valid choice for triggering, albeit with slightly worse performance. For the SiPM detectors, energy and coincidence windowing results in a considerably larger filtering ratio, and coincidence resolution is much worse although the test suggests that better energy resolution values can be achieved.

It must be noted that the time values in Table 6.10 correspond to coincidence resolution, but the time resolution of individual detectors can be inferred from them using (4.28). Indeed, denote the time resolution of PMT and SiPM detectors



(a) PMT-PMT coincidence

(b) PMT-SiPM coincidence

Figure 6.41: Fine near energy histograms used for the determination of energy resolution with PMT and SiPM detectors, with the best Gaussian fit superimposed in red.

	PMT vs PMT		SiPM vs PMT
	timing with AMIC	timing with dynode	
Captured events	7.5×10^6	5×10^6	5×10^5
Filtered events	9×10^4	5×10^4	1×10^3
Filtering ratio	1:83	1:100	1:500
Coincidence resolution (FWHM)	2.04 ns	1.88 ns	3.86 ns
Energy resolution (FWHM)	19.2 %	19.1 %	13.0 %

Table 6.10: Measured coincidence and energy resolution values at the center of the scintillator for all three test cases.

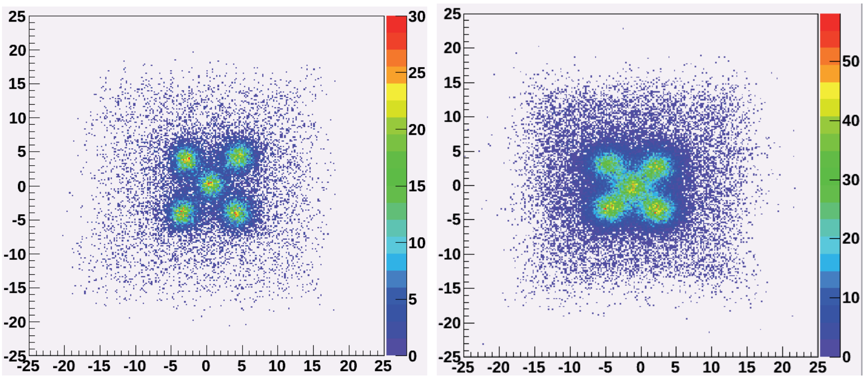
by σ_{PMT} and σ_{SiPM} respectively, and assume that both PMTs have approximately the same resolution. Coincidence resolution for the PMT-PMT test then satisfies

$$\sigma^2_{\mathrm{PMT-PMT}} = \sigma^2_{\mathrm{PMT}} + \sigma^2_{\mathrm{PMT}} = 2\sigma^2_{\mathrm{PMT}} \tag{6.62}$$

which yields a time resolution $\sigma_{\mathrm{PMT}} = 1.44\,\mathrm{ns}$ for individual PMTs. The SiPM-PMT measurements then exhibit a coincidence resolution such that

$$\sigma^2_{\mathrm{SiPM-PMT}} = \sigma^2_{\mathrm{SiPM}} + \sigma^2_{\mathrm{PMT}} \tag{6.63}$$

from which $\sigma_{\mathrm{SiPM}} = 3.58\,\mathrm{ns}$. Finally, this value can be used to estimate the coincidence resolution that would be obtained between two identical SiPM detectors if they were available as $\sigma_{\mathrm{SiPM-SiPM}} = \sqrt{2}\,\sigma_{\mathrm{SiPM}} = 5.06\,\mathrm{ns}$.



(a) PMT-PMT coincidence **(b)** PMT-SiPM coincidence

Figure 6.42: Histograms of measured near positions for PMT and SiPM, using a centered X-shaped grid of source position with 4 mm separation along each axis.

Position resolution

A final test was carried out for the determination of position resolution at the center of the crystal. The stepper motors were used to locate the ^{22}Na source at five different positions on the close detector crystal, forming an X shape centered at the center of the crystal with a 4 mm × 4 mm separation between them, and 10^5 coincidences were captured at each of them. For timestamping and event detection, the same parameters were used as in the previous tests; again, a large coincidence window was used, relegating the windowing step to offline processing. Electronic collimation was applied only at the far detector using $d_{\text{far}} = 0.1$, and the energy and coincidence windows obtained during the previous tests were applied to filter the results.

Figure 6.42 shows the 2D histogram of detected event position for both close detectors, obtained as (6.55). For PMT, the five positions are clearly separated, and a spatial resolution of 2.7 mm and 2.6 mm in different axes is obtained at the center. For SiPM, the image is noisier and the points are blurred but still distinguishable; the measured resolutions at the center are 4.4 mm and 3.9 mm.

Chapter 7

Summary and discussion

In this work, a fully modular and scalable data acquisition system architecture for PET has been proposed that is based on a novel distributed synchronization scheme over data links. A proof-of-concept implementation of the sub-nanosecond synchronization method has been presented first and used to validate the proposal and establish expected performance estimates. A partial implementation of the proposed architecture has then been designed, programmed and evaluated under various configurations. The developed electronics have been shown to be fully functional and established as a replacement of the previous DAQ system at the EDNA facilities with better performance and scalability perspectives.

The conclusions and contributions of this work can be divided into three broad topics: generic DAQ architecture, synchronization, and the particular PET DAQ implementation.

Architectural study

A strong effort has been made throughout this dissertation to establish and justify the design requirements for the desired DAQ system from a theoretical point of view. To this end, the general structure of DAQ systems has been described together with the main issues and trade-offs arising in their specification, first for a generic experimental physics setting in chapter §2 and then in the particular case of PET in chapter §5. Specifically, early digitization and triggerless readout have been identified as the main architectural features that need to be complied with in order to guarantee compatibility with the highest possible detector performance, while modular design and the use of high-capacity point-to-point digital connections over cabling for communication between well-defined functional modules have been established as cost-efficient design choices that maximize the scalability of DAQ systems and the reusability of electronic designs.

Using an elementary argument, it was already established in the introduction chapter that true scalability is only possible if any given functional subset of the DAQ system, including the whole system, provides an interface for precise synchronization. This point has been expanded in §5.2, where a detailed argument is presented that this condition, together with the specifications stated above, imply the necessity of a system-wide synchronization scheme over cabling. Traditional clock tree designs carry several drawbacks such as complex calibration, sensitivity to environmental changes, reduced versatility or increased cabling, as highlighted in §5.2 and especially chapter §3 in much greater detail. Synchronization over data links has thus been presented as a superior choice.

In light of these considerations, a generic architecture for PET DAQ has been proposed based on the replication of two types of module (acquisition and concentration) connected exclusively over digital links with precise synchronization capability. Reducing the number of module types to just one in order to increase modularity might be possible but seems inefficient given that their requirements and function are fundamentally different. The proposed architecture is thus optimal in the sense that it needs the least possible amount of different electronic designs while maintaining arbitrary scalability, albeit with soft limits on bandwidth and synchronization as described in §5.2. It also reduces the cabling requirements to a minimum: only one full-duplex digital link is needed per module, besides power distribution. Some drawbacks are also presented; for instance, its use of the available link bandwidth is extremely asymmetric, in that information flows mostly in one way and the master-to-slave bandwidth is mostly wasted, yet the link must remain active at all times in order to maintain syntonization.

Unfortunately, the fully fleshed-out architecture has not been able to be evaluated experimentally due to financial and organizational constraints. A partial version consisting of only two modules has been implemented instead.

Synchronization

Possible synchronization schemes over data links fit for the development of the proposed DAQ architecture have been discussed in chapter §3. The study has focused on the particular, bare minimum case of synchronization between two timing nodes, which nevertheless allows extension to any connected network by replication of the point-to-point solution over any spanning tree. A thorough analysis of common synchronization methods has revealed their main limitations and how to overcome them; specifically, an extension has been proposed based on the detailed modeling and measurement of link asymmetry that increases achievable resolution by one to two orders of magnitude. It has also been shown that the extended method provides the best possible theoretical performance (3.14) in the two-node case, i.e. half the uncertainty of latency components. Extensions to larger networks deviate from that bound if based on mere replication, however, and thus provide an interesting possible area of research.

A review of related precise synchronization schemes over data links published as a result of recent research has been conducted in §3.4.2 and their shortcomings have been highlighted. Many of them cannot compensate for temperature drift due to one-directionality, while others require the interruption of the data link for the establishment or calibration of synchronization. The main disadvantage, however, lies in the fact that virtually all of them are based on the physical alignment of the phases of all timing clocks in the system, the only exception being [77] which is itself partially based on the work presented in this thesis and relies on offline fine coincidence processing. Fractional timestamps as a time reference of timestamping nodes have been introduced instead as a means of avoiding that restriction and modeling different, even potentially variable phases. It has been argued in §3.5 that this allows for simpler synchronization algorithms in which coarse and fine synchronization procedures are combined into a single step.

The performance of the proposed method has been shown to depend fundamentally on the accuracy in the estimation of the phase difference terms that appear as components of the link skew whenever there is a clock domain crossing between different syntonized clocks. The DDMTD method for phase measurement, first introduced for synchronization applications by the White Rabbit project [138], has been analyzed in §3.3.2 as a promising choice and shown to feature one extremely important drawback: it has a non-zero probability of producing wrong phase estimates that result in synchronization errors of $T_{\text{clk}}/4$. An extension has been proposed based on a simple clustering post-processing step that can be easily implemented online and completely removes these errors; it also results in vastly enhanced measurement resolution, albeit at the cost of reduced sensitivity to rapid phase changes. Experimental results have been used to validate the proposal in a realistic setting, and measured synchronization resolutions have been shown to be within ToF PET specifications i.e. negligible compared to the 500 ps figure proposed in [288] and therefore compatible with any PET system according to the observations made in §5.2.

The hardware requirements for the proposed method have been identified as the use of FPGAs with embedded transceivers that admit deterministic latency configurations. A study of the devices offered by the two main vendors, Xilinx and Altera, has resulted in the theoretical prediction that Xilinx implementations result in better synchronization accuracy than Altera implementations due to the internal structure of their transceivers allowing the alignment of transmitter clocks and the removal of one clock domain crossing in each link. The synchronization method has been implemented using FPGAs from both vendors with essentially equivalent algorithm parameters and rms resolutions of 28 ps for Xilinx and 49 ps for Altera have been estimated in §3.5.2 and §6.3.2, respectively. These measurements partially validate the theoretical prediction, although the expected resolution ratio was $\sqrt{2}$, lower than obtained experimentally.

DAQ implementation

Circuit boards have been designed with the purpose of substituting and upgrading the existing two-detector experimental setup at EDNA. A single board design has been developed that is nevertheless able to act as either an acquisitor, a concentrator, or a mixed-function module, in order to reduce design costs while still being able to partially evaluate the proposed architecture. Its features and integration with the previous electronics have been thoroughly described in chapter §6. It is clear, of course, that the developed boards can be used for data acquisition in many other situations besides a PET system as long as the electrical specifications match i.e. number of channels, voltage range, bandwidth, etc.

Great emphasis has been made in the automation and the definition of very accurately determined protocols for most necessary adjustments and measurements. Examples of this include the automatic calibration of channel offsets, well-defined processes for resolution estimation, and of course link delay self-calibration. This methodology is meant to completely replace the manual adjustment procedures that were being carried out at the EDNA setup in previous incarnations for every setup modification or each new acquisition run, requiring researchers to dedicate an extensive amount of time to these lengthy, error-prone tasks.

The implemented modules have been shown to work with both PMT and SiPM based photodetectors interchangeably; in particular, the ability to work with arrays of 256 SiPM has been proved, which was a novel feature at the time of its publication in 2012 to the best of the author's knowledge. The performance of both photodetectors can only be partially compared since the testing conditions were not the same: the scintillator area coverage for the SiPM detector was only 10 % of that of the PMT. In any case, decent results have been obtained for all measured resolutions using only the simplest configuration for the front-end and digital algorithms i.e. Anger logic, lack of gain calibration, event detection by fixed threshold crossing, and basic, fixed, non-interpolated DCFD for timing. For instance, a coincidence resolution below 1.9 ns has been measured with PMTs, while 3.4 ns were obtained with the previous DAQ system under similar conditions as reported in [278]. No reference could be found for comparison of measured time resolution with SiPM under the same conditions, but previous work at EDNA has shown that the employed detector configuration imposes a limit close to 2.0 ns [23]. In any case, the measured resolutions can be expected to improve by optimizing the digital algorithm parameters and introducing additional online processing steps such as waveform interpolation, as suggested in [278].

It is important to stress out the fact that two different items are presented in this thesis: the generic DAQ architecture for PET systems on one hand, and the particular circuit implementation on the other. The first item is presented in chapter §5, with all previous chapters devoted to discussing different aspects of it, whereas the second item is only described in chapter §6. In particular, the time resolution of the implemented DAQ should not be confused with the specifications on synchronization. The generic architecture is meant to be ToF-compliant, whereas the particular implementation is not; instead, it is intended to be capable of proving that the architecture itself is good enough for ToF by providing an estimate of synchronization resolution and showing that it is in the order of 100 ps. Therefore, it would be possible to develop *different* acquisition modules that allow the proposed architecture to reach ToF-like coincidence resolution. At the current state of the art, it is not possible to achieve that goal using sub-200 MHz FRS acquisition and continuous scintillators; one should therefore resort to analog CFDs, TDCs with pixelated detectors, or custom integrated TDCs.

Summary of the contributions

As a summary of the items discussed above, the main contributions of this thesis are listed here.

- Related to DAQ architecture:
  - A study of DAQ architectures and the identification of the main trends as well as the most important specifications in order to satisfy the design requirements: triggerless readout with local digitization, modular design, and digital connections with embedded synchronization capability.
  - The acknowledgment that trends and advances in DAQ systems for PET parallel those of general DAQ systems for high energy physics experiments in most ways.
  - The proposal of a specific DAQ architecture for PET systems that satisfies the design requirements: arbitrary scalability, mobility, reusability, and compatibility with the highest possible performance i.e. ToF PET. The architecture defines two types of module (acquisition and concentration) and specifies the functional requirements of each of them, as well as that of the connections between them.
- Related to synchronization:
  - The proposal of a particular synchronization scheme over data links, or rather the extension of existing synchronization algorithms on point-to-

    point links, that is able to achieve resolutions around 100 ps, and the proof of its optimality in the Lundelius-Lynch sense.

  - The concept of fractional timestamps as a part of each node's timestamp counter in order to account for phase differences between their local clocks and, thus, lift the restriction of clock alignment that is present in almost all previous precise synchronization methods.

  - An extension of the DDMTD method for phase measurement using a simple clustering algorithm that considerably enhances it by removing the possibility of incorrect measurements and significantly improving its resolution.

  - A comparative between the available hardware that may be used for the proposed method, and the theoretical justification and experimental confirmation that Xilinx FPGAs offer superior performance than equivalent Altera FPGAs in this particular application.

- Related to the developed circuit boards:

  - The development of a particular implementation of the proposed PET DAQ system, including board design, firmware design and control software, as well as its testing, debugging, launch and characterization. Experimental measures confirm that the new DAQ system offers better performance than the previous one.

  - The definition of specific criteria and protocols for calibration, adjustment and measurement in the EDNA setup using the new DAQ system that replace the manual procedures that were in place before.

Possible extensions of this work

As is usual with any thesis of this kind, there are a number of possible direct extensions of the work presented here, and several open questions and research topics have arisen that can be examined with greater depth. A non-exhaustive list of these lines of work is presented here; some of them have already been hinted at within the previous paragraphs.

- The most obvious extension concerns the specification, design and implementation of a dedicated concentrator module with a reasonable number of downlinks to lower level modules. Its functional requirements have already been defined, and a large part of the hardware schematics and firmware contents could be reused from the implementation of the acquisition module presented in chapter §6. This development would enable the deployment of a DAQ system of arbitrary size by replication of the two board designs, that could be used for experimental setups other than PET.[1] Moreover, it would allow testing and validating the following items:

  - Scalability and stability of the architecture. In particular, that of frequency replication across the tree hierarchy (in terms of jitter and stability), the link establishment protocol, and tolerance and fault recovery in the case of link loss.
  - Developing and validating a scalable coincidence detection engine.
  - Characterization of synchronization resolution of the proposed scheme across a multi-level network as a function of system size.
  - A study of the extension of the synchronization algorithm to multi-hop situations and its effect on resolution.

- Extensions of the analysis of synchronization algorithms to more than two nodes can be carried out. Only the "series extension" has been considered in this thesis, where a tree hierarchy is defined and synchronization is reduced to the two node case along each branch. However, the Lundelius-Lynch bound suggests that more precise methods might be available. For instance, the effect of redundancy may be studied, either by connecting acquisition modules to multiple higher-level concentrators or by adding horizontal connections between same-level modules that result in additional time reference estimates that need to be combined somehow. Another example is the extension of synchronization to ring or toroidal topologies; these configurations allow for more efficient coincidence detection and symmetrize the bandwidth usage in both halves of bidirectional links, but it is not immediately clear how to extend the syntonization and synchronization scheme to them.

- The development of acquisition modules that are capable of achieving ToF PET-like coincidence resolution. This would make it possible to validate the stated properties of the proposed architecture using these modules. In particular, a line of work consisting in the implementation of front-end microelectronics for very precise timestamping of detectors consisting of continuous crystals and SiPM detectors was pursued [23] but eventually dropped due to lack of funding.

- The development of the complete version of the firmware for the acquisition modules. Several simplifications have been made in the version used for evaluation of the modules as presented in §6.2; for instance, only a single IML per FPGA has been implemented, and some frame types have only been defined. Additionally, control may be greatly enhanced by introducing

[1] In fact, the design presented here is in itself enough to build a DAQ of arbitrary size as shown in Fig. 6.3, but such a configuration presents severe resolution and latency issues.

hardware acquisition procedures based on processor interrupts and extensive use of DMA instead of processor polling, especially if high event rates are to be supported.

- An extension of the analysis of DDMTD carried out in §3.3.2 and especially appendix A is possible. In particular, an accurate model of the distribution of possible phase measurements could be obtained and expressions for the probability p of error and resolution σ of valid measurements as a function of input jitter values and the stretching factor N could be derived from it. The initial analysis included in this dissertation is rather elementary, and arguments and computations become considerably more complex from that point onward; partial advances have been made but are not ready to be presented yet.

- A study of the digital algorithms and configuration parameters used for the estimation of energy and time marks of subevents. In the tests presented in §6.3.3, fixed configurations have been used both for energy estimation and for timing i.e. DCFD in amplitude mode with $A = 2$ and $k = 1$. Previous work [278] has shown that the resulting time resolution is dependent on the choice of these parameters and how it can be improved by introducing additional online processing steps such as interpolation. Other design choices that have an effect on final resolution but have not been rigorously evaluated include the design of analog shaping filters and the sampling frequency. All of these items can be studied and tested using the electronics implemented in this thesis.

Extension topics that involve the development and fabrication of additional circuit boards are unlikely to be pursued unless a specific application project is found, due to the significant requirements on financial and time investment. Other research tasks are more likely to be undertaken; in fact, the last two items are being followed by the author on a personal level, albeit at a slow pace. In particular, work on the last topic has been ongoing since late 2013 and is expected to be published at some point in 2016.

Appendix A

Distribution of DDMTD phase measurements

In this appendix, approximate analytical expressions are derived regarding the distribution of the raw phase difference estimates provided by the DDMTD algorithm before postprocessing. The objective is not to obtain exact probability values or distributions but rather the qualitative properties of the method and the impact of its parameters; hence, several simplifying assumptions will be made. It will be assumed that the pattern 0011 is used to detect positive clock edges, but the methodology and final formulae are easy to extrapolate to more general cases.

Consider one input clock signal with period T_{clk} that is sampled by a DDMTD clock with period T_{D} given by (3.41). The following assumptions on the clocks will be made for simplicity:

Assumption (Input clock)**.** *The input clock satisfies the following properties:*

1. *Its edges are instantaneous, i.e. with null rise and fall times.*
2. *Its duty cycle is exactly* 50 %*.*

Assumption (Clock jitter)**.** *The absolute jitter in the input and DDMTD clocks is entirely random and given by random variables X_{C} and X_{D}, respectively, satisfying the following properties:*

1. *X_{C} and X_{D} follow symmetrical distributions with null mean.*
2. *X_{C} and X_{D} are independent of each other. Moreover, realizations of X_{C} and X_{D} from different clock edges are independent.*

3. *Jitter is low enough that the probability of $X_{\mathrm{C}}+X_{\mathrm{D}}$ approaching or exceeding $1/4$ of the clock cycle is negligible.*

A few comments on these conditions are in order. The first clock condition is equivalent to disregarding the effect of metastability; the focus will therefore be on the impact of clock jitter on the measurements. The second assumption on jitter means that the correlation between clock jitter in both clocks is not taken into account; such correlation may appear as the result of the DDMTD clock being synthesized from the input clock. Finally, the last condition in both assumptions imply that the analysis can be performed independently for rising and falling edges; these conditions can be replaced by reducing the jitter margin depending on the asymmetry of the clock signal.

All time values will be normalized by T_{clk}, so that the input clock is assumed to have period 1, and the DDMTD clock has period $1+1/N$ and samples the input clock period with a step size of $1/N$. Jitter in the input and DDMTD clocks is assumed to have normalized variance σ_{C}^2 and σ_{D}^2, respectively. The input clock signal is assumed to have its nominal edges at normalized time $t=n$ (positive) and $t=n+\frac{1}{2}$ (negative) for $n\in\mathbb{Z}$.

Probability of detecting a value at a given time

Let t be a normalized time instant where a DDMTD sample is supposed to be taken (i.e. a multiple of $1/N$) and consider the probabilities $P_0\left(t\right)$ and $P_1\left(t\right)$ that the corresponding sample is equal to 0 or 1, respectively. Clearly

$$P_0\left(t\right)+P_1\left(t\right)=1. \tag{A.1}$$

The sample is not actually taken from the input clock signal at time t but rather at $t+x_{\mathrm{D}}\left(t\right)$, where $x_{\mathrm{D}}\left(t\right)$ is the realization of X_{D} corresponding to the DDMTD clock edge at nominal time t. Suppose that t is close to a positive edge of the input clock; more specifically, that $|t-n|<\frac{1}{4}$ for some integer n, so that t is closer to the expected position of the n-th positive edge than it is to any other edge, positive or negative. This edge takes place at time $n+x_{\mathrm{C}}\left(n\right)$, where $x_{\mathrm{C}}\left(n\right)$ is the realization of X_{C} corresponding to the n-th input clock edge.

In this case, the stored sample will be 0 if and only if the actual sampling instant comes before the input clock edge, i.e.

$$P_0\left(t\right)=P\left\{t+x_{\mathrm{D}}\left(t\right)<n+x_{\mathrm{C}}\left(n\right)\right\}=P\left\{t+x<n\right\} \tag{A.2}$$

where $x=x_{\mathrm{D}}\left(t\right)-x_{\mathrm{C}}\left(n\right)$ corresponds to the random variable $X=X_{\mathrm{D}}-X_{\mathrm{C}}$. To see this, notice that, by assumption, X_{C} and X_{D} are symmetric and independent, so X has the same distribution as $X_{\mathrm{D}}+X_{\mathrm{C}}$ and variance

$$\sigma^2=\sigma_{\mathrm{D}}^2+\sigma_{\mathrm{C}}^2. \tag{A.3}$$

In particular, by the third assumption on clock jitter, the absolute value of x cannot exceed $\frac{1}{4}$, so $|t+x-n| < \frac{1}{2}$ and the actual sampling instant can reach neither of the negative edges surrounding the n-th positive edge. Therefore, the value of the stored sample will depend exclusively on whether sampling occurs before or after the n-th positive edge, and (A.2) is proved.

These considerations imply that, for $t \in \left[-\frac{1}{4}, \frac{1}{4}\right]$, the probability of sampling a 0 is

$$P_0\left(t\right) = P\left\{t + x < 0\right\} = P\left\{x < -t\right\} = F\left(-t\right) \tag{A.4}$$

where F is the cumulative distribution function of X. For $t \in \left[\frac{1}{4}, \frac{3}{4}\right]$, located within an interval of length $\frac{1}{2}$ centered at the first negative clock edge, the situation is completely symmetrical to the case near the positive edge, due to the assumptions of symmetry in jitter and duty cycle. In particular, $P_1\left(t\right) = P_0\left(t - \frac{1}{2}\right)$ and vice versa. For other values or t, the probabilities are obtained as $P_0\left(t\right) = P_0\left(t \bmod 1\right)$ as they are obviously periodic.

Probability of detecting a sequence at a given time

Consider now the probability $P_{0011}\left(t\right)$ of detecting a rising edge sequence 0011 centered at time t. This corresponds to four consecutive samples being taken at times $t - 3/2N$, $t - 1/2N$, $t + 1/2N$ and $t + 3/2N$, respectively. Then, for $t \in \left[-\frac{1}{4}, \frac{1}{4}\right]$, the assumption of independence implies that

$$P_{0011}\left(t\right) = P_0\left(t - \frac{3}{2N}\right) P_0\left(t - \frac{1}{2N}\right) P_1\left(t + \frac{1}{2N}\right) P_1\left(t + \frac{3}{2N}\right) \tag{A.5}$$

or

$$\begin{aligned} P_{0011}\left(t\right) = {} & F\left(-\left(t - \frac{3}{2N}\right)\right) \cdot F\left(-\left(t - \frac{1}{2N}\right)\right) \cdot \\ & \cdot \left[1 - F\left(-\left(t + \frac{1}{2N}\right)\right)\right] \cdot \left[1 - F\left(-\left(t + \frac{3}{2N}\right)\right)\right] \end{aligned} \tag{A.6}$$

by (A.4) and (A.1). Actually, this is valid for $|t| \leq \frac{1}{4} - \frac{3}{2N}$ instead of $\frac{1}{4}$, but N is assumed to be large enough, or jitter to be low enough, that the conclusion does not change. For $t \in \left[\frac{1}{4}, \frac{3}{4}\right]$, symmetry implies that the probability is $P_{0011}\left(t\right) = P_{1100}\left(t - \frac{1}{2}\right)$ where, analogously

$$\begin{aligned} P_{1100}\left(t\right) = {} & \left[1 - F\left(-\left(t - \frac{3}{2N}\right)\right)\right] \cdot \left[1 - F\left(-\left(t - \frac{1}{2N}\right)\right)\right] \cdot \\ & \cdot F\left(-\left(t + \frac{1}{2N}\right)\right) \cdot F\left(-\left(t + \frac{3}{2N}\right)\right) \end{aligned} \tag{A.7}$$

if $t \in \left[-\frac{1}{4}, \frac{1}{4}\right]$. It is obvious how these formulae are to be modified for other sequences such as 01 or 000111.

In the particular case where absolute jitter values X_{C} and X_{D} follow normal distributions, then X follows a normal distribution $\mathcal{N}(0, \sigma)$ with σ given by (A.3), i.e.

$$F(t) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{t}{\sqrt{2}\sigma}\right)\right) \tag{A.8}$$

where

$$\operatorname{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-u^2}\, du \tag{A.9}$$

is the Gauss error function. It follows that P_{0011} is given by

$$\begin{aligned} P_{0011}(t) = \frac{1}{16} & \left[1 + \operatorname{erf}\left(-\frac{t - \frac{3}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \left[1 + \operatorname{erf}\left(-\frac{t - \frac{1}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \\ & \cdot \left[1 - \operatorname{erf}\left(-\frac{t + \frac{1}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \left[1 - \operatorname{erf}\left(-\frac{t + \frac{3}{2N}}{\sqrt{2}\sigma}\right)\right] \end{aligned} \tag{A.10}$$

for $t \in \left[-\frac{1}{4}, \frac{1}{4}\right]$, and

$$\begin{aligned} P_{0011}(t) = \frac{1}{16} & \left[1 - \operatorname{erf}\left(-\frac{t - \frac{1}{2} - \frac{3}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \left[1 - \operatorname{erf}\left(-\frac{t - \frac{1}{2} - \frac{1}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \\ & \cdot \left[1 + \operatorname{erf}\left(-\frac{t - \frac{1}{2} + \frac{1}{2N}}{\sqrt{2}\sigma}\right)\right] \cdot \left[1 + \operatorname{erf}\left(-\frac{t - \frac{1}{2} + \frac{3}{2N}}{\sqrt{2}\sigma}\right)\right] \end{aligned} \tag{A.11}$$

for $t \in \left[\frac{1}{4}, \frac{3}{4}\right]$; for other values of t, P_{0011} is extended with a period of 1. Again, the generalization of these formulae to other sequences is straightforward.

Appendix B

Running linear regression

In this appendix, equations are derived for the recurrent computation of the line of best fit corresponding to the last few samples of a discrete signal. This method will be called *running linear regression* by analogy with the well-known concept of running average, i.e. the average value of the last few samples.

Consider first the classic problem of fitting a linear function $l(x) = a_0 + a_1 x$ to a given set of M points $(x_0, y_0), \ldots, (x_{M-1}, y_{M-1})$. The typical criterion for selection is optimality in the least squares sense, i.e. to choose coefficients a_0, a_1 such that the mean square error

$$\frac{1}{M}\sum_{n=0}^{M-1}\left(y_n - l(x_n)\right)^2 = \frac{1}{M}\sum_{n=0}^{M-1}\left(y_n - a_0 - a_1 x_n\right)^2 \tag{B.1}$$

is minimized. This can be solved in a variety of ways, e.g. by equating the partial derivatives of (B.1) with respect to a_0 and a_1 to zero and solving the resulting linear system, which is

$$\begin{pmatrix} \sum 1 & \sum x_n \\ \sum x_n & \sum x_n^2 \end{pmatrix}\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} \sum y_n \\ \sum x_n y_n \end{pmatrix} \tag{B.2}$$

where all sums are defined over $n = 0, \ldots, M-1$.

Now assume that the points correspond to a set of M consecutive samples $y[n]$, $n = 0, \ldots, M-1$ of a waveform that were captured with a uniform sampling rate. The x axis for representation can then be defined in units of sampling clock cycles, with the origin located exactly at the first sample, so that the situation corresponds to the set of points $(x_n, y_n) = (n, y[n])$. In this case, the sums inside the matrix in (B.2) can be substituted with the well-known formulae for the sums of consecutive integers and squares

$$\sum_{n=0}^{M-1} n = \frac{M\left(M-1\right)}{2} \ , \ \sum_{n=0}^{M-1} n^2 = \frac{M\left(M-1\right)\left(2M-1\right)}{6}. \tag{B.3}$$

Moreover, let us denote the *zeroth* and *first-order moments* of the sampled waveform as

$$Y_0 = \sum_{n=0}^{M-1} y\left[n\right] \ , Y_1 = \sum_{n=0}^{M-1} ny\left[n\right]. \tag{B.4}$$

The system (B.2) can then be rewritten in the form

$$\begin{pmatrix} M & \frac{M(M+1)}{2} \\ \frac{M(M+1)}{2} & \frac{M(M+1)(2M+1)}{6} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} Y_0 \\ Y_1 \end{pmatrix} \tag{B.5}$$

whose solution is

$$\begin{aligned} a_0 &= \gamma_{00} Y_0 + \gamma_{01} Y_1 \\ a_1 &= \gamma_{10} Y_0 + \gamma_{11} Y_1 \end{aligned} \tag{B.6}$$

where the coefficients γ_{ij} are given by

$$\gamma_{00} = \frac{2\left(2M-1\right)}{M\left(M+1\right)} \ , \gamma_{01} = \gamma_{10} = -\frac{6}{M\left(M+1\right)} \ , \gamma_{11} = \frac{12}{M\left(M^2-1\right)}. \tag{B.7}$$

Notice that these coefficients depend only on M, and can therefore be precomputed or hardwired in an implementation where the sampling window M remains fixed.

Let us now consider an indefinitely long sequence of samples $\{y\left[n\right]\}$ and the problem of determining the linear regression coefficients $a_0\left[n\right]$, $a_1\left[n\right]$ corresponding to the set of M samples ending at $y\left[n\right]$, i.e. $y\left[n-M+1\right]$, $y\left[n-M+2\right]$, $\ldots$, $y\left[n-1\right]$, $y\left[n\right]$. For a fixed value of n, the situation is the same as before, and so if the origin for $a_0\left[n\right]$ is taken to be the time of the first sample $y\left[n-M+1\right]$, then the coefficients are given by the same expression

$$\begin{aligned} a_0\left[n\right] &= \gamma_{00} Y_0\left[n\right] + \gamma_{01} Y_1\left[n\right] \\ a_1\left[n\right] &= \gamma_{10} Y_0\left[n\right] + \gamma_{11} Y_1\left[n\right] \end{aligned} \tag{B.8}$$

where γ_{ij} are given by (B.7) and the running moments are now

$$Y_0\left[n\right] = \sum_{k=0}^{M-1} y\left[n-M+1+k\right] = \sum_{k=n-M+1}^{n} y\left[k\right] \tag{B.9}$$

$$Y_1\left[n\right] = \sum_{k=0}^{M-1} k \cdot y\left[n-M+1+k\right] = \sum_{k=n-M+1}^{n} \left(k-\left(n-M+1\right)\right) y\left[k\right]. \tag{B.10}$$

The equations (B.8) allow fast computation of the running linear regression coefficients as long as the running moments are available. The moments themselves can be obtained recursively in an efficient manner. For instance, one has

$$Y_0\,[n] - Y_0\,[n-1] = \sum_{k=n-M+1}^{n} y\,[k] - \sum_{k=n-M}^{n-1} y\,[k] = y\,[n] - y\,[n-M] \tag{B.11}$$

that yields the recurrence relation

$$Y_0\,[n] = Y_0\,[n-1] + y\,[n] - y\,[n-M]\,. \tag{B.12}$$

For the first moment,

$$\begin{aligned}
& Y_1\,[n] - Y_1\,[n-1] = \\
& = \sum_{k=n-M+1}^{n} (k-(n-M+1))\,y\,[k] - \sum_{k=n-M}^{n-1} (k-(n-M))\,y\,[k] \\
& = \sum_{k=n-M+1}^{n-1} [(k-(n-M+1)) - (k-(n-M))]\,y\,[k] \\
& + (M-1)\,y\,[n] - 0 \cdot y\,[n-M] \\
& = -\sum_{k=n-M+1}^{n-1} y\,[k] + (M-1)\,y\,[n] \\
& = -\sum_{k=n-M+1}^{n} y\,[k] + My\,[n] = -Y_0\,[n] + My\,[n] \qquad \text{(B.13)}
\end{aligned}$$

which results in the recurrence relation

$$Y_1\,[n] = Y_1\,[n-1] + My\,[n] - Y_0\,[n] \tag{B.14}$$

provided that $Y_0\,[n]$ is being computed, too.

Bibliography

[1] R. Esteve, J. Toledo, J. M. Monzó, A. Sebastiá, J. D. Martínez, C. W. Lerche, and J. M. Benlloch, "A high performance data acquisition system for a 16-head PET scanner," presented at the 11th Int. Workshop on Radiation Imaging Detectors (iWoRID), Prague, Czech Republic, Jun. 2009. 2, 204, 212

[2] Modular scintillator readout solution for nuclear medicine. SensL. [Online]. Available: http://www.sensl.com/downloads/ds/PB-Matrix9SM.pdf 2

[3] Module TEK (MTEK). Philips. [Online]. Available: http://www.digitalphotoncounting.com/products-4/module-tek/ 3

[4] Y. Haemisch, "The digital photon counter (DPC,dSiPM) - a disruptive technology for application in medical imaging, high energy physics and beyond," in *3^{rd} International Conference on Technology and Instrumentation in Particle Physics (TIPP)*, Amsterdam, The Netherlands, Jun. 2014. [Online]. Available: https://indico.cern.ch/event/192695/session/19/contribution/454 3

[5] R. J. Aliaga, J. M. Monzó, M. Spaggiari, N. Ferrando, R. Gadea, and R. J. Colom, "PET system synchronization and timing resolution using high-speed data links," in *IEEE-NPSS Real Time Conference*, Lisbon, Portugal, May 2010, pp. 1–7. [Online]. Available: http://dx.doi.org/10.1109/RTC.2010.5750335 5, 216

[6] R. J. Aliaga, V. Herrero-Bosch, J. M. Monzo, A. Ros, R. Gadea-Girones, and R. J. Colom, "Evaluation of a modular PET system architecture with synchronization over data links," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 1035–1043. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551264 5

[7] R. J. Aliaga, J. M. Monzó, M. Spaggiari, N. Ferrando, R. Gadea, and R. J. Colom, "PET system synchronization and timing resolution using high-speed data links," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1596–1605, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2140130 5, 37, 103, 112, 169

[8] R. J. Aliaga, V. Herrero-Bosch, J. M. Monzo, A. Ros, R. Gadea-Girones, and R. J. Colom, "Evaluation of a modular PET system architecture with synchronization over data links," *IEEE Transactions on Nuclear Science*, vol. 61, no. 1, pp. 88–98, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNS.2014.2298399 5, 204, 211

[9] R. Fernow, *Introduction to experimental particle physics.* Cambridge University Press, 1986. 8, 12, 13, 22

[10] J. Gutleber, R. Moser, and L. Orsini, “Data acquisition in high energy physics,” in *Astronomical Data Analysis Software and Systems XVII*, London, UK, Sep. 2008, pp. 47–56. [Online]. Available: http://adsabs.harvard.edu/abs/2008ASPC..394...47G 8

[11] M. S. Livingston, “Early history of particle accelerators,” in *Advances in electronics and electron physics*, L. Marton and C. Marton, Eds. Academic Press, 1980, vol. 50, pp. 1–88. 10

[12] R. Esteve, “Study and design of the ALICE TPC front-end and readout electronics for the CERN LHC,” Ph.D. dissertation, Universidad Politécnica de Valencia, 2003. 10, 11, 13, 14, 16

[13] R. Frühwirth, M. Regler, R. K. Bock, H. Grote, and D. Notz, *Data analysis techniques for high-energy physics*, 2nd ed. Cambridge University Press, 2000. 10, 21, 23

[14] F. Krauss, “Introduction to particle physics,” University lecture, University of Durham, 2010. [Online]. Available: http://www.ippp.dur.ac.uk/~krauss/Lectures/IntoToParticlePhysics/2010/Lecture4.pdf 10

[15] R. K. Bock and A. Vasilescu, *The particle detector briefbook.* Springer Verlag, 1998. 10

[16] W. Herr and B. Muratori, “Concept of luminosity,” in *CAS-CERN Accelerator School: Intermediate Course on Accelerator Physics*, Zeuthen, Germany, Sep. 2003, pp. 361–378. [Online]. Available: http://dx.doi.org/10.5170/CERN-2006-002.361 10

[17] C. Grupen and B. Shwartz, *Particle detectors*, 2nd ed. Cambridge University Press, 2008. 11

[18] K. T. Pozniak, “FPGA-based, specialized trigger and data acquisition systems for high-energy physics experiments,” *Measurement Science and Technology*, vol. 21, p. 062002 (17pp), 2010. [Online]. Available: http://dx.doi.org/10.1088/0957-0233/21/6/062002 11, 13, 16, 18, 19, 27, 28, 29, 345

[19] F. Hartmann and J. Kaminski, “Advances in tracking detectors,” *Annual Review of Nuclear and Particle Science*, vol. 61, pp. 197–221, 2011. [Online]. Available: http://dx.doi.org/10.1146/annurev-nucl-102010-130052 12

[20] G. F. Knoll, *Radiation detection and measurement*, 3rd ed. John Wiley & Sons, 2000. 13, 14, 127, 129, 133, 134, 230, 232

[21] Z. He, “Review of the Shockley-Ramo theorem and its application in semiconductor gamma-ray detectors,” *Nuclear Instruments and Methods, Section A*, vol. 463, pp. 250–267, 2001. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(01)00223-6 13

[22] V. Herrero, “Análisis y desarrollo de un *front-end* integrado para aplicaciones de tomografía por emisión de positrones,” Ph.D. dissertation, Universidad Politécnica de Valencia, 2007. 13, 139, 143, 340

[23] J. M. Monzo, A. Ros, V. Herrero-Bosch, I. V. Perino, R. J. Aliaga, R. Gadea-Girones, and R. J. Colom-Palero, “Evaluation of a timing integrated circuit architecture for continuous crystal and SiPM based PET systems,” *Journal of Instrumentation*, vol. 8, p. C03017, Mar. 2013. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/8/03/C03017 13, 288, 291

[24] J. F. Toledo, “Study and design of the readout unit module for the LHCb experiment,” Ph.D. dissertation, Universidad Politécnica de Valencia, 2001. 13, 22, 24

[25] J. A. Dueñas, D. Mengoni, V. V. Parkar, R. Berjillos, M. Assie, D. Beaumel, A. M. Sánchez-Benítez, and I. Martel, “Identification of light particles by means of pulse shape analysis with silicon detector at low energy,” *Nuclear Instruments and Methods, Section A*, vol. 676, pp. 70–73, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2012.02.032 14

[26] D. Mengoni, J. A. Dueñas, M. Assié, C. Boiano, P. R. John, R. J. Aliaga, D. Beaumel, S. Capra, A. Gadea, V. González, A. Gottardo, L. Grassi, V. Herrero-Bosch, T. Houdy, I. Martel, V. V. Parkar, R. Pérez-Vidal, A. Pullia, E. Sanchis, A. Triossi, and J. J. Valiente Dobón, “Digital pulse-shape analysis with a TRACE early silicon prototype,” *Nuclear Instruments and Methods, Section A*, vol. 764, pp. 241–246, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2014.07.054 14

[27] M. F. Bieniosek, P. D. Olcott, and C. S. Levin, “Time resolution performance of an electro-optical-coupled PET detector for time-of-flight PET/MRI,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Valencia, Spain, Oct. 2011, pp. 2531–2533. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6152683 14

[28] M. Streun, G. Brandenburg, H. Larue, E. Zimmermann, K. Ziemons, and H. Halling, “Pulse recording by free-running sampling,” *IEEE Transactions on Nuclear Science*, vol. 48, no. 3, pp. 524–526, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.940111 15

[29] P. S. Cooper, “High speed data acquisition,” in *VII ICFA School on Instrumentation in Elementary Particle Physics*, Leon, Mexico, Jul. 1997, pp. 3–13. [Online]. Available: http://dx.doi.org/10.1063/1.55064 16, 17, 22, 23

[30] K. T. Pozniak, “Diagnostic layer integration in FPGA-based pipeline measurement systems for HEP experiments,” *Measurement Science and Technology*, vol. 18, pp. 2432–2445, 2007. [Online]. Available: http://dx.doi.org/10.1088/0957-0233/18/8/019 17, 28

[31] V. González, “Procesado y fusión jerárquica de datos en redes de sensores distribuidos. Aplicación al experimento ATLAS/LHC del CERN,” Ph.D. dissertation, Universitat de València, 1998. 17, 18, 21

[32] S. M. Ross, *Introduction to probability models*, 10th ed. Academic Press, 2010. 18, 189

[33] J. Toledo, F. J. Mora, and H. Müller, “Past, present and future of data acquisition systems in high energy physics experiments,” *Microprocessors and Microsystems*, vol. 27, pp. 353–358, 2003. [Online]. Available: http://dx.doi.org/10.1016/S0141-9331(03)00065-6 18, 25, 26, 29

[34] E. C. Milner, A. W. Booth, M. Botlo, and J. Dorenbosch, “Data acquisition studies for the Superconducting Super Collider,” *IEEE Transactions on Nuclear Science*, vol. 39, no. 2, pp. 138–142, Apr. 1992. [Online]. Available: http://dx.doi.org/10.1109/23.277473 18

[35] N. Neufeld, “LHC trigger & DAQ - an introductory overview,” in *IEEE-NPSS Real Time Conference*, Berkeley, CA, USA, Jun. 2012, pp. 1–4. [Online]. Available: http://dx.doi.org/10.1109/RTC.2012.6418180 18, 19, 30, 34

[36] V. Lindenstruth and I. Kisel, “Overview of trigger systems,” *Nuclear Instruments and Methods, Section A*, vol. 535, pp. 48–56, 2004. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2004.07.267 19, 34

[37] W. H. Smith, “Triggering at LHC experiments,” *Nuclear Instruments and Methods, Section A*, vol. 478, pp. 62–67, 2002. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(01)01720-X 19, 22

[38] W. R. Leo, *Techniques for nuclear and particle physics experiments.* Springer Verlag, 1994. 22

[39] E. Rutherford, “The scattering of α and β particles by matter and the structure of the atom,” *Philosophical Magazine, Series 6*, vol. 21, pp. 669–688, 1911. [Online]. Available: http://dx.doi.org/10.1080/14786440508637080 24

[40] C. T. R. Wilson, “On an expansion apparatus for making visible the tracks of ionising particles in gases and some results obtained by its use,” *Proceedings of the Royal Society of London, Series A*, vol. 87, pp. 277–292, 1912. [Online]. Available: http://www.jstor.org/stable/93225 24

[41] D. A. Glaser, “Some effects of ionizing radiation on the formation of bubbles in liquids,” *Physical Review*, vol. 87, no. 4, p. 665, 1952. [Online]. Available: http://dx.doi.org/10.1103/PhysRev.87.665 24

[42] P. Ponting and H. Verweij, “Instrumentation buses for high energy physics, past, present and future,” *IEEE Transactions on Nuclear Science*, vol. 38, no. 2, pp. 322–324, Apr. 1991. [Online]. Available: http://dx.doi.org/10.1109/23.289318 24, 25, 26

[43] L. Costrell, *Standard Nuclear Instrument Modules*, National Bureau of Standards, Washington, DC Std., 1964. 24

[44] R. S. Larsen and R. W. Downing, “Electronics packaging issues for future accelerators and experiments,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Rome, Italy, Oct. 2004, pp. 1127–1131. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2004.1462401 25, 34

[45] *CAMAC - A Modular Instrumentation System for Data Handling*, EURATOM Std. EUR 4100e, 1972. 25

[46] *FASTBUS Modular High-Speed Data Acquisition and Control System*, IEEE Std. 960-1986, 1986. 25

[47] *IEEE Standard for A Versatile Backplane Bus: VMEbus*, IEEE Std. 1014-1987, 1987. 26

[48] C. F. Parkman, “VICbus - the VME intercrate bus,” in *Buscon UK Proceedings*, London, UK, Oct. 1987. 26

[49] R. S. Larsen, R. W. Downing, Z. A. Liu, A. P. Lowell, V. Pavlicek, S. Simrock, and R. Somes, "New developments in next generation platform standards for physics instrumentation and controls," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Valencia, Spain, Oct. 2011, pp. 2023–2027. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6154411 26

[50] D. Calvet, "A review of technologies for the transport of digital data in recent physics experiments," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 789–794, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.873000 26, 27, 30

[51] B. Gonçalves, J. Sousa, A. Batista, R. Pereira, M. Correia, A. Neto, B. Carvalho, H. Fernandes, and C. A. F. Varandas, "ATCA advanced control and data acquisition systems for fusion experiments," *IEEE Transactions on Nuclear Science*, vol. 57, no. 4, pp. 2147–2154, Aug. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2049501 26, 34

[52] *PCI Local Bus Specification: Revision 3.0*, PCI-SIG Std., 2002. 26

[53] *Standard for Low Voltage Differential Signaling (LVDS) for Scalable Coherent Interface*, IEEE Std. P1596.3-1996, 2002. 27

[54] *CY7B923/CY7B933 HOTLink® transmitter/receiver*, Cypress Semiconductor, 2014. [Online]. Available: http://www.cypress.com/?docID=31689 27

[55] *HDMP 1032-1034 Transmitter-Receiver Chipset Datasheet*, Agilent Technologies. [Online]. Available: http://www.physics.ohio-state.edu/~cms/cfeb/datasheets/hdmp1032.pdf 27, 90, 95

[56] *Media Access Control (MAC) Parameters, Physical Layer, Medium Attachment Units, and Repeater for 100 Mb/s Operation, Type 100BASE-T*, IEEE Std. 802.3u-1995, 1995. 27

[57] *Media Access Control Parameters, Physical Layers, Repeater and Management Parameters for 1,000 Mb/s Operation*, IEEE Std. 802.3z-1998, 1998. 27

[58] A. Aloisio, F. Cevenini, and V. Izzo, "An approach to DWDM for real-time applications," *IEEE Transactions on Nuclear Science*, vol. 51, no. 3, pp. 526–531, Jun. 2004. [Online]. Available: http://dx.doi.org/10.1109/TNS.2004.828713 27

[59] D. Breton, E. Delagnes, and M. Houry, "Very high dynamic range and high sampling rate VME digitizing boards for physics experiments," *IEEE Transactions on Nuclear Science*, vol. 52, no. 6, pp. 2853–2860, Dec. 2005. [Online]. Available: http://dx.doi.org/10.1109/TNS.2005.860165 27

[60] G. Hall, "Recent progress in front end ASICs for high-energy physics," *Nuclear Instruments and Methods, Section A*, vol. 541, pp. 248–258, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2005.01.064 27

[61] R. Szczygiel, "A state of the art rad-hard digital ASIC design for high energy physics experiments," *Measurement Science and Technology*, vol. 18, pp. 2413–2417, 2007. [Online]. Available: http://dx.doi.org/10.1088/0957-0233/18/8/016 27

[62] J. J. Rodriguez-Andina, M. J. Moure, and M. D. Valdes, "Features, design tools, and application domains of FPGAs," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1810–1823, Aug. 2007. [Online]. Available: http://dx.doi.org/10.1109/TIE.2007.898279 28, 337

[63] *Nios II Processor Reference Handbook*, NII5V1-13.1, Altera, 2014. [Online]. Available: http://www.altera.com/literature/hb/nios2/n2cpu_nii5v1.pdf 29, 239

[64] *MicroBlaze Processor Reference Guide*, UG081, Xilinx, 2012. [Online]. Available: http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_1/mb_ref_guide.pdf 29

[65] *PowerPC Processor Reference Guide*, UG011, Xilinx, 2010. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug011.pdf 29

[66] *Virtex-II Platform FPGAs: Complete Data Sheet*, DS031, Xilinx, 2014. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds031.pdf 29

[67] S. Anvar, O. Gachelin, P. Kestener, H. Le Provost, and I. Mandjavidze, "FPGA-based system-on-chip designs for real-time applications in particle physics," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 682–687, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.875076 29, 30

[68] E. J. Siskind, "Data acquisition system issues for large experiments," *Nuclear Instruments and Methods, Section A*, vol. 579, pp. 839–843, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2007.05.303 29

[69] *Virtex-II Pro and Virtex-II Pro X Platform FPGAs: Complete Data Sheet*, DS083, Xilinx, 2011. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds083.pdf 30

[70] R. J. Aliaga, V. Herrero-Bosch, S. Capra, A. Pullia, J. A. Dueñas, L. Grassi, A. Triossi, C. Domingo-Pardo, R. Gadea, V. González, T. Hüyük, E. Sanchís, A. Gadea, and D. Mengoni, "Conceptual design of the TRACE detector readout using a compact, dead time-less analog memory ASIC," *Nuclear Instruments and Methods, Section A*, vol. 800, pp. 34–39, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2015.07.067 31

[71] D. Barrientos, M. Bellato, D. Bazzacco, D. Bortolato, P. Cocconi, A. Gadea, V. González, M. Gulmini, R. Isocrate, D. Mengoni, A. Pullia, F. Recchia, D. Rosso, E. Sanchis, N. Toniolo, C. A. Ur, and J. J. Valiente-Dobón, "Performance of the fully digital FPGA-based front-end electronics for the GALILEO array," *IEEE Transactions on Nuclear Science*, Oct. 2015, pending publication. [Online]. Available: http://dx.doi.org/10.1109/TNS.2015.2480243 31

[72] A. Navarro-Tobar *et al.*, "Upgrade of the second level of the readout electronics for the CMS drift tubes subdetector," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Valencia, Spain, Oct. 2011, pp. 819–822. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6154546 32

[73] I. Koronov, H. Angerer, A. Mann, and S. Paul, "SODA: Time distribution system for the PANDA experiment," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Orlando, FL, USA, Oct. 2009, pp. 1863–1865. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2009.5402172 33, 95

[74] M. Kavatsyuk, M. Hevinga, I. Koronov, P. J. J. Lemmens, P. Marciniewski, P. Schakel, F. Schreuder, R. Speelman, G. Tambave, T. Johansson, and H. Löhner, "Trigger-less readout electronics for the PANDA electromagnetic calorimeter," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*,

Valencia, Spain, Oct. 2011, pp. 43–47. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6154360 33, 90, 95

[75] F. Alessio, "Trigger-less readout architecture for the upgrade of the LHCb experiment at CERN," *Journal of Instrumentation*, vol. 8, p. C12019, Dec. 2013. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/8/12/C12019 33, 34

[76] I. H. Lazarus, D. E. Appelbe, P. A. Butler, P. J. Coleman-Smith, J. R. Cresswell, R. D. Herzberg, I. Hibbert, D. T. Joss, S. C. Letts, R. D. Page, V. F. E. Pucknell, P. H. Regan, J. Sampson, J. Simpson, J. Thornhill, and R. Wadsworth, "The GREAT triggerless total data readout method," *IEEE Transactions on Nuclear Science*, vol. 48, no. 3, pp. 567–569, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.940120 33

[77] S. Anvar, F. Château, H. Le Provost, F. Louis, K. Manolopoulos, Y. Moudden, B. Vallage, and E. Zonca, "Design and implementation of a nanosecond time-stamping readout system-on-chip for photo-detectors," *Nuclear Instruments and Methods, Section A*, vol. 735, pp. 587–595, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2013.10.019 34, 99, 100, 101, 287

[78] R. S. Larsen, "Advances in developing next-generation electronics standards for physics," in *IEEE-NPSS Real Time Conference*, Beijing, China, May 2009, pp. 7–15. [Online]. Available: http://dx.doi.org/10.1109/RTC.2009.5321803 34, 35, 49

[79] F. Carrió, V. Castillo, A. Ferrer, L. Fiorini, Y. Hernández, E. Higón, B. Mellado, L. March, P. Moreno, R. Reed, C. Solans, A. Valero, and J. A. Valls, "The sROD module for the ATLAS Tile Calorimeter Phase-II upgrade demonstrator," *Journal of Instrumentation*, vol. 9, p. C02019, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/9/02/C02019 34

[80] M. Hansen *et al.*, "CMS ECAL electronics developments for HL-LHC," *Journal of Instrumentation*, vol. 10, p. C03028, Mar. 2015. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/10/03/C03028 34

[81] S. Kulis and M. Idzik, "Triggerless readout with time and amplitude reconstruction of events based on deconvolution algorithm," in *Workshop on Timing Detectors*, Kraków, Poland, Nov. 2010, pp. 49–57. [Online]. Available: http://dx.doi.org/10.5506/APhysPolBSupp.4.49 34

[82] S. Bachmann, N. Berger, A. Blondel, S. Bravar, A. Buniatyan, G. Dissertori, P. Eckert, P. Fischer, C. Grab, R. Gredig, M. Hildebrandt, P.-R. Kettle, M. Kiehn, A. Papa, I. Perić, M. Pohl, S. Ritt, P. Robmann, A. Schöning, H.-C. Schultz-Coulon, W. Shen, S. Shresta, A. Stoykov, U. Straumann, R. Wallny, D. Wiedner, and B. Windelband, "The proposed trigger-less TBit/s readout for the Mu3e experiment," *Journal of Instrumentation*, vol. 9, p. C01011, Jan. 2014. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/9/01/C01011 34

[83] G. Mazza, D. Calvo, P. de Remigis, T. Kugathasan, M. Mignone, A. Rivetti, L. Toscano, R. Wheadon, and L. Zotti, "Triggerless readout architecture for the silicon pixel detector of the PANDA experiment," *Journal of Instrumentation*, vol. 8, p. C02017, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/8/02/C02017 34

[84] J. de Cuveland and V. Lindenstruth, "A first-level event selector for the CBM experiment at FAIR," *Journal of physics: Conference series*, vol. 331, no. 2, p.

022006, 2011. [Online]. Available: http://dx.doi.org/10.1088/1742-6596/331/2/022006 34

[85] R. S. Larsen, "xTCA for physics standards roadmaps & SLAC initiatives," in *IEEE-NPSS Real Time Conference*, Nara, Japan, May 2014, pp. 1–7. [Online]. Available: http://dx.doi.org/10.1109/RTC.2014.7097432 34

[86] A. D. O. Karlsson and B. Martin, "ATCA: its performance and application for real time systems," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 688–693, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.873404 34

[87] R. S. Larsen, "PICMG xTCA standards extensions for physics: new developments and future plans," in *IEEE-NPSS Real Time Conference*, Lisbon, Portugal, May 2010, pp. 1–7. [Online]. Available: http://dx.doi.org/10.1109/RTC.2010.5750327 34

[88] S. Baron, T. Mastoridis, J. Troska, and P. Baudrenghien, "Jitter impact on clock distribution in LHC experiments," *Journal of Instrumentation*, vol. 7, p. C12023, Dec. 2012. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/7/12/C12023 37

[89] F. M. Gardner and W. C. Lindsey, "Guest editorial: Special issue on synchronization," *IEEE Transactions on Communications*, vol. 28, no. 8, pp. 1105–1106, Aug. 1980. [Online]. Available: http://dx.doi.org/10.1109/TCOM.1980.1094774 38

[90] S. H. Hall, G. W. Hall, and J. A. McCall, *High-speed digital system design*. John Wiley & Sons, 2000. 39, 167

[91] *Stratix II GX Device Handbook, Volume 2*, SIIGX5V2-4.3, Altera, 2007. [Online]. Available: http://www.altera.com/literature/hb/stx2gx/stxiigx_sii5v2.pdf 43, 46, 93

[92] W. Stallings, *Data and computer communications*, 10th ed. Pearson Education International, 2014. 44

[93] A. S. Tanenbaum, *Computer networks*, 4th ed. Prentice Hall, 2002. 45

[94] A. X. Widmer and P. A. Franaszek, "A DC-balanced, partitioned-block, 8B/10B transmission code," *IBM Journal of Research and Development*, vol. 27, no. 5, pp. 440–451, Sep. 1983. [Online]. Available: http://dx.doi.org/10.1147/rd.275.0440 45

[95] A. Aloisio, F. Cevenini, R. Giordano, and V. Izzo, "Characterizing jitter performance of multi gigabit FPGA-embedded serial transceivers," *IEEE Transactions on Nuclear Science*, vol. 57, no. 2, pp. 451–455, Apr. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2032291 45, 106, 219

[96] T. J. Ferguson and J. H. Rabinowitz, "Self-synchronizing Huffman codes," *IEEE Transactions on Information Theory*, vol. 30, no. 4, pp. 687–693, Jul. 1984. [Online]. Available: http://dx.doi.org/10.1109/TIT.1984.1056931 45

[97] *CYP15G0101DXB/CYV15G0101DXB Single-channel HOTLink II^{TM} transceiver*, Cypress Semiconductor, 2012. [Online]. Available: http://www.cypress.com/?docID=35742 46, 90

[98] *SLK2511A OC-48/24/12/3 SONET/SDH multirate transceiver*, Texas Instruments, 2009. [Online]. Available: http://www.ti.com/lit/ds/slls610c/slls610c.pdf 46, 90

[99] *TLK1221 Ethernet transceiver*, Texas Instruments, 2009. [Online]. Available: http://www.ti.com/lit/ds/slls713c/slls713c.pdf 46, 90

[100] *TLK3101 2.5 Gbps to 3.125 Gbps transceiver*, Texas Instruments, 2008. [Online]. Available: http://www.ti.com/lit/ds/symlink/tlk3101.pdf 46, 90

[101] *DS92LV16 16-bit bus LVDS serializer/deserializer - 25-80 MHz*, Texas Instruments, 2013. [Online]. Available: http://www.ti.com/lit/ds/symlink/ds92lv16.pdf 46, 90

[102] *Virtex-5 FPGA RocketIO GTP Transceiver User Guide*, UG196, Xilinx, 2009. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug196.pdf 46, 90, 91, 338

[103] M. Bellato, "AGATA global trigger and synchronization hardware," INFN Padova, Tech. Rep., Nov. 2005. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00729086 49, 97, 100, 345

[104] H. Li, G. Gong, W. Pan, Q. Du, and J. Li, "Temperature effect on White Rabbit timing link," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1021–1026, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNS.2015.2425659 50, 97, 98

[105] E. Buterbaugh, *Perfect Timing II: A design guide for clock generation and distribution.* Cypress Semiconductor Corporation, 2002. [Online]. Available: http://www.cypress.com/?docID=14449 50

[106] A. Demir, A. Mehrotra, and J. Roychowdhury, "Phase noise in oscillators: A unifying theory and numerical methods for characterization," *IEEE Transactions on Circuits and Systems I*, vol. 47, no. 5, pp. 655–674, May 2000. [Online]. Available: http://dx.doi.org/10.1109/81.847872 50

[107] D. C. Lee, "Analysis of jitter in phase-locked loops," *IEEE Transactions on Circuits and Systems II*, vol. 49, no. 11, pp. 704–711, Nov. 2002. [Online]. Available: http://dx.doi.org/10.1109/TCSII.2002.807265 50

[108] N. Ashby, "Relativity in the Global Positioning System," *Living Reviews in Relativity*, vol. 6, p. 1, 2003. [Online]. Available: http://www.livingreviews.org/lrr-2003-1 53

[109] H. Kopetz and W. Ochsenreiter, "Clock synchronization in distributed real-time systems," *IEEE Transactions on Computers*, vol. C-36, no. 8, pp. 933–940, Aug. 1987. [Online]. Available: http://dx.doi.org/10.1109/TC.1987.5009516 53, 55

[110] K. Arvind, "Probabilistic clock synchronization in distributed systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 5, no. 5, pp. 474–487, May 1994. [Online]. Available: http://dx.doi.org/10.1109/71.282558 55

[111] P. Baron, D. Calvet, E. Delagnes, X. de la Broise, A. Delbart, F. Druillole, E. Mazzucato, E. Monmarthe, F. Pierre, and M. Zito, "AFTER, an ASIC for the readout of the large T2K time projection chambers," *IEEE Transactions on Nuclear Science*, vol. 55, no. 3, pp. 1744–1752, Jun. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.924067 56

[112] *Quad, 12-bit, 170 MSPS/210 MSPS/250 MSPS serial output 1.8V ADC*, AD9239, Analog Devices, 2010. [Online]. Available: http://www.analog.com/static/imported-files/data_sheets/AD9239.pdf 58, 215, 227, 243

[113] P. D. Reynolds, P. D. Olcott, G. Pratx, F. W. Y. Lau, and C. S. Levin, "Convex optimization of coincidence time resolution for a high-resolution PET system," *IEEE Transactions on Medical Imaging*, vol. 30, no. 2, pp. 391–400, Feb. 2011. [Online]. Available: http://dx.doi.org/10.1109/TMI.2010.2080282 61, 160

[114] M. Nakao, "Timing distribution for the Belle II data acquisition system," *Journal of Instrumentation*, vol. 7, p. C01028, Jan. 2012. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/7/01/C01028 61

[115] J. Zhang, J. Wu, Z. Han, L. Liu, K. Tian, and J. Dong, "Low power, accurate time synchronization MAC protocol for real-time wireless data acquisition," *IEEE Transactions on Nuclear Science*, vol. 60, no. 5, pp. 3683–3688, Oct. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2250306 62

[116] D. L. Mills, "Internet time synchronization: the Network Time Protocol," *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, Oct. 1991. [Online]. Available: http://dx.doi.org/10.1109/26.103043 62

[117] W. Lewandowski, J. Azoubib, and W. J. Klepczynksi, "GPS: Primary tool for time transfer," *Proceedings of the IEEE*, vol. 87, no. 1, pp. 163–172, Jan. 1999. [Online]. Available: http://dx.doi.org/10.1109/5.736348 62

[118] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 3, no. 3, pp. 281–323, May 2005. [Online]. Available: http://dx.doi.org/10.1016/j.adhoc.2005.01.002 62

[119] J. Lundelius and N. Lynch, "An upper and lower bound for clock synchronization," *Information and Control*, vol. 62, pp. 190–204, 1984. [Online]. Available: http://dx.doi.org/10.1016/S0019-9958(84)80033-9 63, 68

[120] *IEEE Standard for a precision clock synchronization protocol for networked measurement and control systems*, IEEE Std. 1588-2008, 2008. [Online]. Available: http://dx.doi.org/10.1109/IEEESTD.2008.4579760 63

[121] H. Huckeba and R. Dlugy-Hegwer, "Precise time synchronization using IEEE 1588 for LXI applications," in *IEEE Autotestcon*, Anaheim, CA, USA, Sep. 2006, pp. 129–135. [Online]. Available: http://dx.doi.org/10.1109/AUTEST.2006.283609 63

[122] P. Ferrari, A. Flammini, D. Marioli, S. Rinaldi, and A. Taroni, "Synchronization of the probes of a distributed instrument for real-time Ethernet networks," in *IEEE International Symposium on Precision Clock Synchronization (ISPCS)*, Vienna, Austria, Oct. 2007, pp. 33–40. [Online]. Available: http://dx.doi.org/10.1109/ISPCS.2007.4383770 63

[123] P. Loschmidt, R. Exel, A. Nagy, , and G. Gaderer, "Limits of synchronization accuracy using hardware support in IEEE 1588," in *IEEE International Symposium on Precision Clock Synchronization (ISPCS)*, Ann Arbor, MI, USA, Sep. 2008, pp. 12–16. [Online]. Available: http://dx.doi.org/10.1109/ISPCS.2008.4659205 63, 68

[124] J. Savoj and B. Razavi, *High-speed CMOS circuits for optical receivers.* Kluwer Academic Publishers, 2001. 81, 106

[125] S. Liu, J. Tan, and B. Hou, “Multicycle synchronous digital phase measurement used to further improve phase-shift laser range finding,” *Measurement Science and Technology*, vol. 18, no. 6, pp. 1756–1762, 2007. [Online]. Available: http://dx.doi.org/10.1088/0957-0233/18/6/S14 81

[126] P. Moreira, P. Alvarez, J. Serrano, I. Darwezeh, and T. Wlostowski, “Digital dual mixer time difference for sub-nanosecond time synchronization in Ethernet,” in *IEEE International Frequency Control Symposium (FCS)*, Newport Beach, CA, USA, Jun. 2010, pp. 449–453. [Online]. Available: http://dx.doi.org/10.1109/FREQ.2010.5556289 81, 84

[127] D. W. Allan and H. Daams, “Picosecond time difference measurement system,” in *Proceedings of the 29th Annual Symposium on Frequency Control*, Atlantic City, NJ, May 1975, pp. 404–411. [Online]. Available: http://dx.doi.org/10.1109/FREQ.1975.200112 81

[128] P. Moreira and I. Darwazeh, “Digital femtosecond time difference circuit for CERN’s timing system,” in *London Communications Symposium*, London, UK, Sep. 2011. [Online]. Available: http://www.ee.ucl.ac.uk/lcs/previous/LCS2011/LCS1136.pdf 84, 86

[129] T. Kohonen, *Self-organization and associative memory*, 3rd ed. Springer Verlag, 1989. 86

[130] G. Cervelli, A. Marchioro, and P. Moreira, “A 0.13-µm CMOS serializer for data and trigger optical links in particle physics experiments,” *IEEE Transactions on Nuclear Science*, vol. 51, no. 3, pp. 836–841, Jun. 2004. [Online]. Available: http://dx.doi.org/10.1109/TNS.2004.829551 90

[131] A. Aloisio, F. Cevenini, R. Giordano, and V. Izzo, “High-speed, fixed-latency serial links with FPGAs for synchronous transfers,” *IEEE Transactions on Nuclear Science*, vol. 56, no. 5, pp. 2864–2873, Oct. 2009. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2027236 90, 92, 95

[132] *RocketIO^{TM} Transceiver User Guide*, UG024, Xilinx, 2007. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug024.pdf 90

[133] *Virtex-4 RocketIO Multi-Gigabit Transceiver User Guide*, UG076, Xilinx, 2008. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug076.pdf 90

[134] F. Lemke, D. Slogsnat, N. Burkhardt, and U. Bruening, “A unified DAQ interconnection network with precise time synchronization,” *IEEE Transactions on Nuclear Science*, vol. 57, no. 2, pp. 412–418, Apr. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2042176 90, 95, 98

[135] R. Giordano and A. Aloisio, “Fixed-latency, multi-gigabit serial links with Xilinx FPGAs,” *IEEE Transactions on Nuclear Science*, vol. 58, no. 1, pp. 194–201, Feb. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2101083 90, 91, 95, 97

[136] A. Rohlev, A. Borga, J. Serrano, M. Cattin, and M. Stettler, “Sub-nanosecond machine timing and frequency distribution via serial data links,” in *Topical Workshop on Electronics for Particle Physics (TWEPP)*, Naxos, Greece, Sep. 2008, pp. 411–413. [Online]. Available: http://dx.doi.org/10.5170/CERN-2008-008.411 90, 97, 100, 104

[137] P. P. M. Jansweijer and H. Z. Peek, "Measuring propagation delay over a coded serial communication channel using FPGAs," *Nuclear Instruments and Methods, Section A*, vol. 626, pp. S169–S172, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2010.04.126 90

[138] P. Moreira, J. Serrano, T. Wlostowski, P. Loschmidt, and G. Gaderer, "White Rabbit: sub-nanosecond timing distribution over Ethernet," in *IEEE International Symposium on Precision Clock Synchronization (ISPCS)*, Brescia, Italy, Oct. 2009, pp. 58–62. [Online]. Available: http://dx.doi.org/10.1109/ISPCS.2009.5340196 90, 287

[139] *Stratix GX FPGA Family Datasheet*, DS-STXGX-2.2, Altera, 2004. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/ds/ds_sgx.pdf 93

[140] *Stratix IV GX Device Handbook, Volume 2: Transceivers*, SIV5V2-4.6, Altera, 2014. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/hb/stratix-iv/stx4_5v2.pdf 93, 94, 338

[141] A. Aloisio, F. Cevenini, R. Giordano, and V. Izzo, "Emulating the GLink chip set with FPGA serial transceivers in the ATLAS level-1 muon trigger," *IEEE Transactions on Nuclear Science*, vol. 57, no. 2, pp. 467–471, Apr. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2036175 95

[142] X. Liu, Q.-X. Deng, and Z.-K. Wang, "Design and FPGA implementation of high-speed, fixed-latency serial transceivers," *IEEE Transactions on Nuclear Science*, vol. 61, no. 1, pp. 561–567, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2296301 95

[143] M. Lipiński, T. Włostowski, J. Serrano, and P. Alvarez, "White Rabbit: a PTP application for robust sub-nanosecond synchronization," in *IEEE International Symposium on Precision Clock Synchronization (ISPCS)*, Munich, Germany, Sep. 2011, pp. 25–30. [Online]. Available: http://dx.doi.org/10.1109/ISPCS.2011.6070148 96, 100

[144] J.-L. Ferrant, M. Gilson, S. Jobert, M. Mayer, M. Ouellette, L. Montini, S. Rodrigues, and S. Ruffini, "Synchronous Ethernet: a method to transport synchronization," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 126–134, Sep. 2008. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2008.4623717 96

[145] L. Shang, K. Song, P. Cao, C. Li, S. Liu, and Q. An, "A prototype clock system for LHAASO WCDA," *IEEE Transactions on Nuclear Science*, vol. 60, no. 5, pp. 3537–3543, Oct. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2280907 97, 100

[146] M. Bellato, L. Berti, D. Bortolato, P. J. Coleman Smith, P. Edelbruck, X. Grave, R. Isocrate, I. Lazarus, D. Linget, P. Medina, C. Oziol, G. Rampazzo, C. Santos, B. Travers, and A. Triossi, "Global trigger and readout system for the AGATA experiment," *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 91–98, Feb. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2007.910034 97

[147] M. Bellato, D. Bortolato, J. Chavas, R. Isocrate, G. Rampazzo, A. Triossi, D. Bazzacco, D. Mengoni, and F. Recchia, "Sub-nanosecond clock synchronization and trigger management in the nuclear physics experiment AGATA," *Journal of*

Instrumentation, vol. 8, p. P07003, Jul. 2013. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/8/07/P07003 97, 100

[148] S. Schatral, F. Lemke, and U. Bruening, "Design of a deterministic link initialization mechanism for serial LVDS interconnects," *Journal of Instrumentation*, vol. 9, p. C03022, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/9/03/C03022 98, 100

[149] I. Papakonstantinou, C. Soos, S. Papadopoulos, S. Détraz, C. Sigaud, P. Stejskal, S. Storey, J. Troska, F. Vasey, and I. Darwazeh, "A fully bidirectional optical network with latency monitoring capability for the distribution of timing-trigger and control signals in high-energy physics experiments," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1628–1640, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2154364 98, 100

[150] A. Hidvégi, P. Geßler, K. Rehlich, and C. Bohm, "Timing and triggering system prototype for the XFEL project," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1852–1856, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2151205 98, 100

[151] A. Tarazona, K. Gnanvo, S. Martoiu, H. Muller, and J. Toledo, "A point-to-point link for data, trigger, clock and control over copper or fibre," *Journal of Instrumentation*, vol. 9, p. T06004, Jun. 2014. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/9/06/T06004 99, 100

[152] H. Le Provost, Y. Moudden, S. Anvar, F. Château, V. Gautard, F. Louis, K. Ménager, B. Vallage, and E. Zonca, "A readout system-on-chip for a cubic kilometer submarine neutrino telescope," *Journal of Instrumentation*, vol. 6, p. C12044, Dec. 2011. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/6/12/C12044 99

[153] *ML505/ML506/ML507 Evaluation Platform*, UG347, Xilinx, 2011. [Online]. Available: http://www.xilinx.com/support/documentation/boards_and_kits/ug347.pdf 103

[154] P. P. M. Jansweijer and H. Z. Peek, "Measuring propagation delay over a 1.25 Gbps bidirectional data link," National Institute for Subatomic Physics (NIKHEF), Tech. Rep. ETR 2010-01, May 2010. [Online]. Available: http://www.nikhef.nl/pub/services/biblio/technicalreports/ETR2010-01.pdf 104

[155] *16-bit, low power, voltage output, I^2C interface digital-to-analog converter*, DAC8571, Texas Instruments, 2003. [Online]. Available: http://www.ti.com/lit/ds/symlink/dac8571.pdf 104

[156] J. Gaither, *Digital phase-locked loop (DPLL) reference design*, Application Note XAPP854, Xilinx, Oct. 2006. [Online]. Available: http://www.xilinx.com/support/documentation/application_notes/xapp854.pdf 104

[157] J. D. H. Alexander, "Clock recovery from random binary signals," *Electronics Letters*, vol. 11, no. 22, pp. 541–542, oct 1975. [Online]. Available: http://dx.doi.org/10.1049/el:19750415 106

[158] *Virtex-5 FPGA Data Sheet: DC and switching characteristics*, DS202, Xilinx, 2014. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds202.pdf 106

[159] M. Ramezani and C. A. T. Salama, "Analysis of a half-rate bang-bang phase locked loop," *IEEE Transactions on Circuits and Systems II*, vol. 49, pp. 505–509, Jul. 2002. [Online]. Available: http://dx.doi.org/10.1109/TCSII.2002.805020 106

[160] A. M. Scott, "PET imaging in oncology," in *Positron emission tomography*, D. L. Bailey, D. W. Townsend, P. E. Valk, and M. N. Maisey, Eds. Springer, 2005, pp. 311–325. 120, 345

[161] R. Badawi. (1999) Introduction to PET physics. Division of Nuclear Medicine, University of Washington. [Online]. Available: http://depts.washington.edu/nucmed/IRL/pet_intro/ 120, 121, 345

[162] K. Otsuka and T. Togawa, "Hippocratic thermography," *Physiological measurement*, vol. 18, no. 3, pp. 227–232, Aug. 1997. [Online]. Available: http://dx.doi.org/10.1088/0967-3334/18/3/007 121

[163] W. Wayand, "The history of minimally invasive surgery," *Global Surgery 2004*, pp. 37–38, 2004. 121

[164] P. K. Spiegel, "The first clinical X-ray made in America - 100 years," *American Journal of Roentgenology*, vol. 164, no. 1, pp. 241–243, 1995. [Online]. Available: http://dx.doi.org/10.2214/ajr.164.1.7998549 121

[165] G. Hevesy, "The absorption and translocation of lead by plants," *Biochemistry Journal*, vol. 17, pp. 439–445, 1923. 121

[166] F. Joliot and I. Curie, "Artificial production of a new kind of radio-element," *Nature*, vol. 133, no. 3354, pp. 201–202, 1934. [Online]. Available: http://dx.doi.org/10.1038/133201a 121

[167] H. N. Wagner Jr., "A brief history of positron emission tomography (PET)," *Seminars in Nuclear Medicine*, vol. 28, no. 3, pp. 213–220, 1998. [Online]. Available: http://dx.doi.org/10.1016/S0001-2998(98)80027-5 121, 122

[168] B. Cassen, L. Curtis, C. Reed, and R. Libby, "Instrumentation for ^{131}I use in medical studies," *Nucleonics*, vol. 9, no. 2, pp. 46–50, 1951. 121

[169] R. Nutt, "The history of positron emission tomography," *Molecular Imaging & Biology*, vol. 4, no. 1, pp. 11–26, 2002. [Online]. Available: http://dx.doi.org/10.1016/S1095-0397(00)00051-0 122

[170] W. H. Sweet, "The uses of nuclear disintegration in the diagnosis and treatment of brain tumor," *New England Journal of Medicine*, vol. 245, no. 23, pp. 875–878, 1951. [Online]. Available: http://dx.doi.org/10.1056/NEJM195112062452301 122

[171] F. R. Wrenn, M. L. Good, and P. Handler, "The use of positron-emitting radioisotopes for the localization of brain tumors," *Science*, vol. 113, no. 2940, pp. 525–527, 1951. [Online]. Available: http://dx.doi.org/10.1126/science.113.2940.525 122

[172] G. L. Brownell and W. H. Sweet, "Localization of brain tumors with positron emitters," *Nucleonics*, vol. 11, no. 11, pp. 40–45, 1953. 122

[173] H. O. Anger, "Scintillation camera," *Review of Scientific Instruments*, vol. 29, pp. 27–33, 1958. [Online]. Available: http://dx.doi.org/10.1063/1.1715998 122, 148

[174] A. Ros, "Optimización de cristales centelleadores para la determinación de la DOI en tomografía de rayos gamma," Ph.D. dissertation, Universitat de València, 2012. 122, 130, 149

[175] H. O. Anger and D. J. Rosenthal, "Scintillation camera and positron camera," *Medical radioisotope scanning*, vol. 19, pp. 59–75, 1959. 122

[176] J. S. Robertson, R. B. Marr, M. Rosenblum, V. Radeka, and Y. L. Yamamoto, "32-crystal positron transverse section detector," in *Tomographic imaging in nuclear medicine*, G. S. Freedman, Ed. Society of Nuclear Medicine Press, 1973. 122, 205

[177] P. V. Harper, K. A. Lathrop, D. Charleston, and R. Beck, "Optimization of scanning method using Tc^{99m}," *Nucleonics*, vol. 22, no. 1, p. 50, 1964. 122

[178] G. L. Brownell, C. A. Burnham, D. A. Chesler, J. A. Correia, J. E. Correll, B. Hoop, J. A. Parker, and R. Subramanyam, "Transverse section imaging of radionuclide distributions in heart, lung and brain," in *Reconstruction tomography in diagnostic radiology and nuclear medicine*, M. M. Ter-Pogossian, M. E. Phelps, and G. L. Brownell, Eds. University Park Press, 1977, pp. 293–307. 122

[179] D. E. Kuhl and R. Q. Edwards, "Reorganizing data from transverse section scans of the brain using digital processing," *Radiology*, vol. 91, pp. 975–983, 1968. [Online]. Available: http://dx.doi.org/10.1148/91.5.975 122

[180] G. N. Hounsfield, "Computerised transverse axial scanning (tomography): Part 1. Description of system," *British Journal of Radiology*, vol. 46, no. 552, pp. 1016–1022, 1973. [Online]. Available: http://dx.doi.org/10.1259/0007-1285-46-552-1016 122

[181] M. M. Ter-Pogossian, M. E. Phelps, E. J. Hoffman, and N. A. Mullani, "A positron-emission transaxial tomograph for nuclear imaging (PETT)," *Radiology*, vol. 114, pp. 89–98, 1975. [Online]. Available: http://dx.doi.org/10.1148/114.1.89 122, 141

[182] Z. H. Cho and M. Farukhi, "Bismuth germanate as a potential scintillation detector in positron cameras," *Journal of Nuclear Medicine*, vol. 18, pp. 840–844, 1977. [Online]. Available: http://jnm.snmjournals.org/content/18/8/840.short 122

[183] C. L. Melcher and J. S. Schweitzer, "Cerium-doped lutetium oxyorthosilicate: a fast, efficient new scintillator," *IEEE Transactions on Nuclear Science*, vol. 39, no. 4, pp. 502–505, Aug. 1992. [Online]. Available: http://dx.doi.org/10.1109/23.159655 122, 135, 205

[184] D. W. Townsend, J. P. J. Carney, J. T. Yap, and N. C. Hall, "PET/CT today and tomorrow," *Journal of Nuclear Medicine*, vol. 45, no. 1 suppl., pp. 4S–14S, 2004. [Online]. Available: http://jnm.snmjournals.org/content/45/1_suppl/4S.short 122, 123, 339

[185] H.-P. W. Schlemmer, B. J. Pichler, M. Schmand, Z. Burbar, C. Michel, R. Ladebeck, K. Jattke, D. Townsend, C. Nahmias, P. K. Jacob, W.-D. Heiss, and C. D. Claussen, "Simultaneous MR/PET imaging of the human brain: feasibility study," *Radiology*, vol. 248, no. 3, pp. 1028–1035, 2008. [Online]. Available: http://dx.doi.org/10.1148/radiol.2483071927 122

[186] W. R. Hendee and E. R. Ritenour, *Medical imaging physics*, 4th ed. John Wiley & Sons, 2002. 124, 129

[187] K. S. Krane, *Introductory nuclear physics*, 3rd ed. John Wiley & Sons, 1987. 125

[188] J. Nuyts, “Nuclear medicine technology and techniques,” University lecture, Katholieke Universiteit Leuven, 2013. [Online]. Available: ftp://134.58.179.7/pub/nuyts/cursus/cursus_nucleo.pdf 126, 134, 161, 172

[189] J. D. Martínez, “Estudio y diseño de la electrónica de adquisición de datos para Mamografía por Emisión de Positrones con detectores continuos,” Ph.D. dissertation, Universidad Politécnica de Valencia, 2008. 126, 147, 204

[190] R. R. Raylman, B. E. Hammer, and N. L. Christensen, “Combined MRI-PET scanner: a Monte Carlo evaluation of the improvements in PET resolution due to the effects of a static homogeneous magnetic field,” *IEEE Transactions on Nuclear Science*, vol. 43, no. 4, pp. 2406–2412, Aug. 1996. [Online]. Available: http://dx.doi.org/10.1109/23.531789 127

[191] V. I. Grafutin and E. P. Prokop’ev, “Positron annihilation spectroscopy in materials structure studies,” *Physics-Uspekhi*, vol. 45, no. 1, pp. 59–74, 2002. [Online]. Available: http://dx.doi.org/10.1070/PU2002v045n01ABEH000971 127

[192] C. S. Levin and E. J. Hoffman, “Calculation of positron range and its effect on the fundamental limit of positron emission tomography system spatial resolution,” *Physics in Medicine and Biology*, vol. 44, no. 3, pp. 781–799, 1999. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/44/3/019 127

[193] J. M. Monzó, “Estudio e implementación de algoritmos digitales para la mejora de la resolución temporal en sistemas de tomografía por emisión de positrones,” Ph.D. dissertation, Universidad Politécnica de Valencia, 2012. 127, 166, 168, 252

[194] C. W. Lerche, “Depth of interaction enhanced gamma-ray imaging for medical applications,” Ph.D. dissertation, Universitat de València, 2006. 128, 131, 134, 135, 139, 149, 152, 339, 340, 345

[195] A. P. Jeavons, D. W. Townsend, N. L. ford, K. Kull, A. Manuel, O. Fischer, and M. Peter, “A high-resolution proportional chamber positron camera and its applications,” *IEEE Transactions on Nuclear Science*, vol. 25, no. 1, pp. 164–173, Feb. 1978. [Online]. Available: http://dx.doi.org/10.1109/TNS.1978.4329298 128

[196] P. Bruyndonckx, Y. Wang, S. Tavernier, and P. Carnochan, “Design and performance of a data acquisition system for VUB-PET,” *IEEE Transactions on Nuclear Science*, vol. 48, no. 1, pp. 150–156, Feb. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.907579 128, 197

[197] J. L. Lacy, C. S. Martin, and L. P. Armendarez, “High sensitivity, low cost PET using lead-walled straw detectors,” *Nuclear Instruments and Methods, Section A*, vol. 471, pp. 88–93, 2001. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(01)00959-7 128

[198] A. Miceli, J. Glister, A. Andreyev, D. Bryman, L. Kurchaninov, P. Lu, A. Muennich, F. Retiere, and V. Sossi, “Simulations of a micro-PET system based on liquid xenon,” *Physics in Medicine and Biology*, vol. 57, no. 6, pp. 1685–1700, 2012. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/57/6/1685 128

[199] S. Seifert, J. H. L. Steenbergen, H. T. van Dam, and D. R. Schaart, “Accurate measurement of the rise and decay times of fast scintillators with solid state photon counters,” *Journal of Instrumentation*, vol. 7, no. 9, p. P09004, Sep. 2012. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/7/09/P09004 133

[200] J. G. Rogers and C. J. Batty, "Afterglow in LSO and its possible effect on energy resolution," *IEEE Transactions on Nuclear Science*, vol. 47, no. 2, pp. 438–445, Apr. 2000. [Online]. Available: http://dx.doi.org/10.1109/23.846277 134

[201] R. Novotny, "Inorganic scintillators: a basic material for instrumentation in physics," *Nuclear Instruments and Methods, Section A*, vol. 537, pp. 1–5, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2004.07.221 134

[202] P. Guerra, "Contribuciones al diseño e implementación de un sistema de alta resolución para tomografía por emisión," Ph.D. dissertation, Universidad Politécnica de Madrid, 2007. 135

[203] R. A. Yotter and D. M. Wilson, "A review of photodetectors for sensing light-emitting reporters in biological systems," *IEEE Sensors Journal*, vol. 3, no. 3, pp. 288–303, Jun. 2003. [Online]. Available: http://dx.doi.org/10.1109/JSEN.2003.814651 136, 137, 139, 142, 340, 345

[204] R. Foord, R. Jones, C. J. Oliver, and E. R. Pike, "The use of photomultiplier tubes for photon counting," *Applied optics*, vol. 8, no. 10, pp. 1975–1989, 1969. [Online]. Available: http://dx.doi.org/10.1364/AO.8.001975 137

[205] M. Watanabe, T. Omura, H. Kyushima, Y. Hasegawa, and T. Yamashita, "A compact position-sensitive detector for PET," *IEEE Transactions on Nuclear Science*, vol. 42, no. 4, pp. 1090–1094, Aug. 1995. [Online]. Available: http://dx.doi.org/10.1109/23.467743 137

[206] D. Renker, "Photosensors," *Nuclear Instruments and Methods, Section A*, vol. 527, pp. 15–20, 2004. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2004.03.010 138, 141, 340

[207] ——, "Properties of avalanche photodiodes for applications in high energy physics, astrophysics and medical imaging," *Nuclear Instruments and Methods, Section A*, vol. 486, pp. 164–169, 2002. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(02)00696-4 139

[208] A. N. Otte, J. Barral, B. Dolgoshein, J. Hose, S. Klemin, E. Lorenz, R. Mirzoyan, E. Popova, and M. Teshima, "A test of silicon photomultipliers as readout for PET," *Nuclear Instruments and Methods, Section A*, vol. 545, pp. 705–715, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2005.02.014 141, 340

[209] A. W. Lightstone, R. J. McIntyre, R. Lecomte, and D. Schmitt, "A bismuth germanate-avalanche photodiode module designed for use in high resolution positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 33, no. 1, pp. 456–459, Feb. 1986. [Online]. Available: http://dx.doi.org/10.1109/TNS.1986.4337142 141

[210] P. Buzhan, B. Dolgoshein, L. Filatov, A. Ilyin, V. Kantzerov, V. Kaplin, A. Karakash, F. Kayumov, S. Klemin, E. Popova, and S. Smirnov, "Silicon photomultiplier and its possible applications," *Nuclear Instruments and Methods, Section A*, vol. 504, pp. 48–52, 2003. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(03)00749-6 141

[211] A. Nassalski, M. Moszyński, A. Syntfeld-Każuch, T. Szczęśniak, Ł. Świdersli, D. Wolski, T. Batsch, and J. Baszak, "Multi pixel photon counters (MPPC) as an alternative to APD in PET applications," *IEEE Transactions on Nuclear Science*,

vol. 57, no. 3, pp. 1008–1014, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2044586 141

[212] A. del Guerra, N. Belcari, M. G. Bisogni, G. Llosá, S. Marcatili, and S. Moehrs, “Advances in position-sensitive photodetectors for PET applications,” *Nuclear Instruments and Methods, Section A*, vol. 604, pp. 319–322, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.01.034 141, 142, 345

[213] W. J. Ashmanskas, B. C. LeGeyt, F. M. Newcomer, J. V. Panetta, W. A. Ryan, R. van Berg, R. I. Wiener, and J. S. Karp, “Waveform-sampling electronics for a whole-body time-of-flight PET scanner,” *IEEE Transactions on Nuclear Science*, vol. 61, no. 3, pp. 1174–1181, Jun. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNS.2014.2303119 141, 198, 202, 204

[214] V. Bettinardi, L. Presotto, E. Rapisarda, M. Picchio, L. Gianolli, and M. C. Gilardi, “Physical performance of the new hybrid PET/CT Discovery-690,” *Medical physics*, vol. 38, no. 10, pp. 5394–5411, 2010. [Online]. Available: http://dx.doi.org/10.1118/1.3635220 141, 170

[215] J. E. Mackewn, C. W. Lerche, B. Weissler, K. Sunassee, R. T. M. de Rosales, A. Phinikaridou, A. Salomon, R. Ayres, C. Tsoumpas, G. M. Soultanidis, P. Gebhardt, T. Schaeffter, P. K. Marsden, and V. Schulz, “PET performance evaluation of a pre-clinical SiPM-based MR-compatible PET scanner,” *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 784–790, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNS.2015.2392560 141

[216] C. Bauer, A. Stolin, J. Proffitt, P. Martone, J. Brefczynski-Lewis, J. Lewis, J. Hankiewicz, R. Raylman, and S. Majewski, “Development of a ring PET insert for MRI,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Seoul, South Korea, Oct. 2013, pp. 1–9. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2013.6829135 141

[217] M. Budassi, M. L. Purschke, J. Fried, T. Cao, S. Stoll, E. Gualtieri, J. S. Karp, P. O’Connor, D. J. Schlyer, C. L. Woody, and P. Vaska, “First results from the BNL/Penn PET-MRI system for whole body rodent imaging at 9.4T,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 2753–2755. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551626 141, 204, 207

[218] V. C. Spanoudaki and C. S. Levin, “Photo-detectors for time of flight positron emission tomography (ToF-PET),” *Sensors*, vol. 10, pp. 10 484–10 505, 2010. [Online]. Available: http://dx.doi.org/10.3390/s101110484 141, 142, 345

[219] T. Frach, G. Prescher, C. Degenhardt, R. de Gruyter, A. Schmitz, and R. Ballizany, “The digital silicon photomultiplier - principle of operation and intrinsic detector performance,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Orlando, FL, USA, Oct. 2009, pp. 1959–1965. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2009.5402143 141

[220] L. H. C. Braga, L. Gasparini, L. Grant, R. K. Henderson, N. Massari, M. Perenzoni, D. Stoppa, and R. Walker, “A fully digital 8 × 16 SiPM array for PET applications with per-pixel TDCs and real-time energy output,” *IEEE Journal of Solid-State Circuits*, vol. 49, no. 1, pp. 301–314, Jan. 2014. [Online]. Available: http://dx.doi.org/10.1109/JSSC.2013.2284351 142, 204, 205

[221] B. Weissler, P. Gebhardt, P. Düppenbecker, B. Goldschmidt, A. Salomon, D. Schug, J. Wehner, C. Lerche, D. Wirtz, W. Renz, K. Schumacher, B. Zwaans, P. Marsden, F. Kiessling, and V. Schulz, "Design concept of world's first preclinical PET/MR insert with fully digital silicon photomultiplier technology," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 2113–2116. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551484 142

[222] W. W. Moses, P. R. G. Virador, S. E. Derenzo, R. H. Huesman, and T. F. Budinger, "Design of a high-resolution, high-sensitivity PET camera for human brains and small animals," *IEEE Transactions on Nuclear Science*, vol. 44, no. 4, pp. 1487–1491, Aug. 1997. [Online]. Available: http://dx.doi.org/10.1109/23.632691 144

[223] J. L. Humm, A. Rosenfeld, and A. del Guerra, "From PET detectors to PET scanners," *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 30, no. 11, pp. 1574–1597, Nov. 2003. [Online]. Available: http://dx.doi.org/10.1007/s00259-003-1266-2 144

[224] M. E. Casey and R. Nutt, "A multicrystal 2-dimensional BGO detector system for positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 33, no. 1, pp. 460–463, Feb. 1986. [Online]. Available: http://dx.doi.org/10.1109/TNS.1986.4337143 145, 146, 196, 205

[225] E. N. Giménez, J. M. Benlloch, M. Giménez, C. W. Lerche, M. Fernández, N. Pavón, M. Rafecas, F. Sánchez, A. Sebastiá, R. Esteve, J. D. Martínez, and J. Toledo, "Detector optimization of a small animal PET camera based on continuous LSO crystals and flat panel PS-PMTs," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Rome, Italy, Oct. 2004, pp. 3885–3889. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2004.1466727 145

[226] L. R. Furenlid, E. Clarkson, D. G. Marks, and H. H. Barrett, "Spatial pileup considerations for pixellated gamma-ray detectors," *IEEE Transactions on Nuclear Science*, vol. 47, no. 4, pp. 1399–1403, Aug. 2000. [Online]. Available: http://dx.doi.org/10.1109/23.872985 146

[227] P. Bruyndonckx, C. Lemaître, S. Léonard, D. R. Schaart, D. J. van der Laan, M. C. Maas, O. Devroede, Y. Wu, M. Krieguer, and S. Tavernier, "Initial characterization of a nonpixelated scintillator detector in a PET prototype demonstrator," *IEEE Transactions on Nuclear Science*, vol. 53, no. 5, pp. 2543–2548, Oct. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.875998 146, 197

[228] G. Landi, "Properties of the center of gravity as an algorithm for position measurements," *Nuclear Instruments and Methods, Section A*, vol. 485, pp. 698–719, 2002. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(01)02071-X 148, 150

[229] ——, "Properties of the center of gravity as an algorithm for position measurements: two-dimensional geometry," *Nuclear Instruments and Methods, Section A*, vol. 497, pp. 511–534, 2003. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(02)01822-3 148, 150

[230] C. W. Lerche, M. Döring, A. Ros, V. Herrero, R. Gadea, R. J. Aliaga, R. Colom, F. Mateo, J. M. Monzó, N. Ferrando, J. F. Toledo, J. D. Martínez, A. Sebastiá, F. Sanchez, and J. M. Benlloch, "Depth of interaction detection for γ-ray imaging," *Nuclear Instruments and Methods, Section A*, vol. 600, pp. 624–634, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2008.11.151 148, 149, 156, 157

[231] S. Siegel, R. W. Silverman, Y. Shao, and S. R. Cherry, "Simple charge division readouts for imaging scintillator arrays using a multi-channel PMT," *IEEE Transactions on Nuclear Science*, vol. 43, no. 3, pp. 1634–1641, Jun. 1996. [Online]. Available: http://dx.doi.org/10.1109/23.507162 149, 196

[232] D. Clément, R. Frei, J.-F. Loude, and C. Morel, "Development of a 3D position sensitive scintillation detector using neural networks," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Toronto, Canada, Nov. 1998, pp. 1448–1452. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.1998.773818 152, 155

[233] A. M. Bronstein, M. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi, "Optimal nonlinear line-of-flight estimation in positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 50, no. 3, pp. 421–426, Jun. 2003. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.812434 152

[234] P. Bruyndonckx, C. Lemaître, D. J. van der Laan, M. Maas, D. Schaart, W. Yonggang, Z. Li, M. Krieguer, and S. Tavernier, "Evaluation of machine learning algorithms for localization of photons in undivided scintillator blocks for PET detectors," *IEEE Transactions on Nuclear Science*, vol. 55, no. 3, pp. 918–924, Jun. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.922811 152, 196

[235] R. J. Aliaga, J. D. Martínez, R. Gadea, A. Sebastiá, J. M. Benlloch, F. Sánchez, N. Pavón, and C. W. Lerche, "Corrected position estimation in PET detector modules with multi-anode PMTs using neural networks," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 776–783, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.875438 152, 154, 340

[236] F. Mateo, R. J. Aliaga, R. Gadea, J. D. Martínez, and J. M. Monzó, "Incidence position estimation in a PET detector using a discretized positioning circuit and neural networks," *Computational and Ambient Intelligence*, vol. 4507, pp. 684–691, 2007. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-73007-1_82 152

[237] F. Mateo, R. J. Aliaga, N. Ferrando, J. D. Martínez, V. Herrero, C. W. Lerche, R. J. Colom, J. M. Monzó, A. Sebastiá, and R. Gadea, "High-precision position estimation in PET using artificial neural networks," *Nuclear Instruments and Methods, Section A*, vol. 604, pp. 366–369, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.01.058 152

[238] R. J. Aliaga, R. Gadea, R. J. Colom, J. M. Monzó, C. W. Lerche, and J. D. Martínez, "System-on-chip implementation of neural network training on FPGA," *International Journal on Advances in Systems and Measurements*, vol. 2, no. 1, pp. 44–55, 2009. [Online]. Available: http://www.thinkmind.org/download.php?articleid=sysmea_v2_n1_2009_4 152

[239] C. W. Lerche, A. Ros, J. M. Monzó, R. J. Aliaga, N. Ferrando, J. D. Martínez, V. Herrero, R. Esteve, R. Gadea, R. J. Colom, J. Toledo, F. Mateo, A. Sebastiá, F. Sanchez, and J. M. Benlloch, "Maximum likelihood positioning for gamma-ray imaging detectors with depth of interaction measurements," *Nuclear Instruments and Methods, Section A*, vol. 604, pp. 359–362, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.01.060 153, 154, 340

[240] W. C. J. Hunter, H. H. Barrett, and L. R. Furenlid, "Calibration method for ML estimation of 3D interaction position in a thick gamma-ray detector," *IEEE Transactions on Nuclear Science*, vol. 56, no. 1, pp. 189–196, Feb. 2009. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.2010704 153

[241] Y. H. Chung, Y. Choi, T. Y. Song, J. H. Jung, G. Cho, Y. S. Choe, K.-H. Lee, S. E. Kim, and B.-T. Kim, "Evaluation of maximum-likelihood position estimation with Poisson and Gaussian noise models in a small gamma camera," *IEEE Transactions on Nuclear Science*, vol. 51, no. 1, pp. 101–104, Feb. 2004. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.823053 153

[242] J. Joung, R. S. Miyaoka, S. G. Kohlmyer, and T. K. Lewellen, "Investigation of bias-free positioning estimators for the scintillation cameras," *IEEE Transactions on Nuclear Science*, vol. 48, no. 3, pp. 715–719, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.940152 153

[243] W. W. Moses and S. E. Derenzo, "Effect of depth of interaction measurement resolution on radial elongation in PET," *Journal of Nuclear Medicine*, vol. 31, p. 749, 1990. 154

[244] J. Seidel, J. J. Vaquero, S. Siegel, W. R. Gandler, and M. V. Green, "Depth identification accuracy of a three layer phoswich PET detector module," *IEEE Transactions on Nuclear Science*, vol. 46, no. 3, pp. 485–490, Jun. 1999. [Online]. Available: http://dx.doi.org/10.1109/23.775567 155

[245] Y. Wang, J. Seidel, B. M. W. Tsui, J. J. Vaquero, and M. G. Pomper, "Performance evaluation of the GE Healthcare eXplore VISTA dual-ring small-animal PET scanner," *Journal of Nuclear Medicine*, vol. 47, pp. 1891–1900, 2006. [Online]. Available: http://jnm.snmjournals.org/content/47/11/1891.short 155, 205

[246] V. V. Nagarkar, V. Gaysinskiy, V. Gelfandbein, S. Miller, S. Cool, H. Kudrolli, H. B. Barber, K. Haston, P. M. Kain, and V. Bora, "Continuous Phoswich$^{\text{TM}}$detector for molecular imaging," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Knoxville, TN, USA, Nov. 2010, pp. 4–9. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2010.5873707 155

[247] H. Liu, T. Omura, M. Watanabe, and T. Yamashita, "Development of a depth of interaction detector for γ-rays," *Nuclear Instruments and Methods, Section A*, vol. 459, pp. 182–190, 2001. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(00)00939-6 155

[248] M. Rafecas, G. Böning, B. J. Pichler, E. Lorenz, M. Schwaiger, and S. I. Ziegler, "A Monte Carlo study of high-resolution PET with granulated dual-layer detectors," *IEEE Transactions on Nuclear Science*, vol. 48, no. 4, pp. 1490–1495, Aug. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.958385 155

[249] J. S. Huber, W. W. Moses, M. S. Andreaco, and O. Petterson, "An LSO scintillator array for a PET detector module with depth of interaction measurement," *IEEE Transactions on Nuclear Science*, vol. 48, no. 3, pp. 684–688, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1109/23.940147 155

[250] P. A. Dokhale, R. W. Silverman, K. S. Shah, R. Farrell, M. A. McClish, G. Entine, and S. R. Cherry, "Intrinsic spatial resolution and parallax correction using depth-endoded PET detector modules based on position-sensitive APD readout," *IEEE Transactions on Nuclear Science*, vol. 53, no. 5, pp. 2666–2670, Oct. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.882807 155

[251] C. W. Lerche, J. M. Benlloch, F. Sánchez, N. Pavón, N. Giménez, M. Fernández, M. Giménez, A. Sebastiá, J. Martínez, and F. J. Mora, "Depth of interaction detection with enhanced position-sensitive proportional resistor network," *Nuclear Instruments and Methods, Section A*, vol. 537, pp. 326–330, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2004.08.036 156

[252] C. W. Lerche, J. M. Benlloch, F. Sánchez, N. Pavón, B. Escat, E. N. Gimenez, M. Fernández, I. Torres, M. Giménez, A. Sebastiá, and J. Martínez, "Depth of γ-ray interaction within continuous crystals from the width of its scintillation light-distribution," *IEEE Transactions on Nuclear Science*, vol. 52, no. 3, pp. 560–572, Jun. 2005. [Online]. Available: http://dx.doi.org/10.1109/TNS.2005.851424 156

[253] V. Herrero, R. Colom, R. Gadea, C. W. Lerche, J. Cerdá, A. Sebastiá, and J. M. Benlloch, "Front-end circuit for position sensitive silicon and vacuum tube photomultipliers with gain control and depth of interaction measurement," *Nuclear Instruments and Methods, Section A*, vol. 576, pp. 118–122, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2007.01.181 157, 196

[254] V. Herrero-Bosch, R. J. Colom, R. Gadea, J. Espinosa, J. M. Monzó, R. Esteve, A. Sebastiá, C. W. Lerche, and J. M. Benlloch, "PESIC: an integrated front-end for PET applications," *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 27–33, Feb. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2007.909902 157

[255] C. W. Lerche, V. Herrero-Bosch, M. Spaggiari, F. Mateo-Jimenez, J. M. Monzó-Ferrer, R. J. Colom-Palero, and F. Mora-Mas, "Fast circuit topology for spatial signal distribution analysis," in *IEEE-NPSS Real Time Conference*, Lisbon, Portugal, May 2010, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1109/RTC.2010.5750391 158

[256] V. Herrero, C. W. Lerche, M. Spaggiari, R. J. Aliaga, N. Ferrando, and R. J. Colom, "AMIC: An expandable front-end for gamma-ray detectors with light distribution analysis capabilities," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1641–1646, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2152855 158, 196

[257] M. Spaggiari, V. Herrero, C. W. Lerche, R. J. Aliaga, J. M. Monzó, and R. Gadea, "AMIC: An expandable integrated analog front-end for light distribution moment analysis," *Journal of Instrumentation*, vol. 6, pp. 1010–1020, Jan. 2011. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/6/01/C01094 158

[258] A. Ros, R. J. Aliaga, V. Herrero-Bosch, J. M. Monzo, A. Gonzalez, R. J. Colom, F. J. Mora, and J. M. Benlloch, "Expandable programmable integrated front-end for scintillator based photodetectors," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 3196–3200. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551729 158

[259] V. Herrero-Bosch, J. M. Monzo, A. Ros, R. J. Aliaga, A. González, C. Montoliu, R. J. Colom-Palero, and J. M. Benlloch, "Programmable integrated front-end for SiPM/PMT PET detectors with continuous scintillating crystal," *Journal of Instrumentation*, vol. 7, p. C12021, Dec. 2012. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/7/12/C12021 158, 213

[260] W. W. Moses and C. J. Thompson, "Timing calibration in PET using a time alignment probe," *IEEE Transactions on Nuclear Science*, vol. 53, no. 5, pp. 2660–2665, Oct. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.882797 160

[261] W. W. Moses, "Time of flight in PET revisited," *IEEE Transactions on Nuclear Science*, vol. 50, no. 5, pp. 1325–1330, Oct. 2003. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.817319 160, 170, 205, 341

[262] W. W. Moses and M. Ullisch, "Factors influencing timing resolution in a commercial LSO PET camera," *IEEE Transactions on Nuclear Science*, vol. 53, no. 1, pp. 78–85, Feb. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2005.862980 161, 163

[263] R. F. Post and L. I. Schiff, "Statistical limitations on the resolving time of a scintillation counter," *Physical Review*, vol. 80, no. 6, p. 1113, 1950. [Online]. Available: http://dx.doi.org/10.1103/PhysRev.80.1113 162

[264] M. Kelbert, I. Sazonov, and A. G. Wright, "Exact expression for the variance of the photon emission process in scintillation counters," *Nuclear Instruments and Methods, Section A*, vol. 564, pp. 185–189, 2006. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2006.04.049 162

[265] Y. Shao, "A new timing model for calculating the intrinsic timing resolution of a scintillator detector," *Physics in Medicine and Biology*, vol. 52, no. 4, pp. 1103–1117, 2007. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/52/4/016 162

[266] F. Powolny, "Characterization of time resolved photodetector systems for Positron Emission Tomography," Ph.D. dissertation, Université de Neuchâtel, 2009. 162, 163

[267] H. H. Barrett and K. J. Myers, *Foundations of image science.* John Wiley & Sons, 2003. 163

[268] K. Pauwels, E. Auffray, S. Gundacker, A. Knapitsch, and P. Lecoq, "Effect of aspect ratio on the light output of scintillators," *IEEE Transactions on Nuclear Science*, vol. 59, no. 5, pp. 2340–2345, Oct. 2012. [Online]. Available: http://dx.doi.org/10.1109/TNS.2012.2183890 163

[269] A. Ros, C. W. Lerche, A. Sebastiá, F. Sánchez, and J. M. Benlloch, "Retroreflector arrays for better light collection efficiency of γ-ray imaging

detectors with continuous scintillation crystals without DOI misestimation," *Journal of Instrumentation*, vol. 9, p. P04009, Apr. 2014. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/9/04/P04009 163

[270] J. M. Monzó, R. J. Aliaga, V. Herrero, J. D. Martínez, F. Mateo, A. Sebastiá, F. J. Mora, J. M. Benlloch, and N. Pavón, "Accurate simulation testbench for nuclear imaging systems," *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 421–428, Feb. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2007.912878 163

[271] R. Vinke, H. Löhner, D. R. Schaart, H. T. van Dam, S. Seifert, F. J. Beekman, and P. Dendooven, "Optimizing the timing resolution of SiPM sensors for use in TOF-PET detectors," *Nuclear Instruments and Methods, Section A*, vol. 610, pp. 188–191, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.05.068 163, 198

[272] T. J. Paulus, "Timing electronics and fast timing methods with scintillation detectors," *IEEE Transactions on Nuclear Science*, vol. 32, no. 3, pp. 1242–1249, Jun. 1985. [Online]. Available: http://dx.doi.org/10.1109/TNS.1985.4337024 164, 165

[273] D. A. Gedcke and W. J. McDonald, "A constant fraction of pulse height trigger for optimum time resolution," *Nuclear Instruments and Methods, Section A*, vol. 55, pp. 377–380, 1967. [Online]. Available: http://dx.doi.org/10.1016/0029-554X(67)90145-0 165

[274] Z. H. Cho and R. L. Chase, "Improved amplitude and rise time compensated timing with Ge detectors," *IEEE Transactions on Nuclear Science*, vol. 19, no. 1, pp. 451–460, 1972. [Online]. Available: http://dx.doi.org/10.1109/TNS.1972.4326545 166

[275] A. Fallu-Labruyere, H. Tan, W. Hennig, and W. K. Warburton, "Time resolution studies using digital constant fraction discrimination," *Nuclear Instruments and Methods, Section A*, vol. 579, pp. 247–251, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2007.04.048 166

[276] J. M. Monzó, C. W. Lerche, J. D. Martínez, R. Esteve, J. F. Toledo, R. Gadea, R. J. Colom, N. Ferrando, R. J. Aliaga, F. Mateo, F. Sánchez, F. J. Mora, J. M. Benlloch, and A. Sebastiá, "Analysis of time resolution in a dual head LSO+PSPMT PET system using low pass filter interpolation and digital constant fraction discriminator techniques," *Nuclear Instruments and Methods, Section A*, vol. 604, pp. 347–350, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.01.062 167

[277] H. Semmaoui, M.-A. Tétrault, R. Lecomte, and R. Fontaine, "Signal deconvolution concept combined with cubic spline interpolation to improve timing with phoswitch PET detectors," *IEEE Transactions on Nuclear Science*, vol. 56, no. 3, pp. 581–587, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2015946 167

[278] J. M. Monzó, R. Esteve, C. W. Lerche, N. Ferrando, J. Toledo, R. J. Aliaga, V. Herrero, and F. J. Mora, "Digital signal processing techniques to improve time resolution in positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1613–1620, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2140382 167, 168, 288, 292

[279] R. Vinke, H. Löhner, D. R. Schaart, H. T. van Dam, S. Seifert, F. J. Beekman, and P. Dendooven, "Time walk correction for TOF-PET detectors based on a monolithic scintillation crystal coupled to a photosensor array," *Nuclear Instruments and Methods, Section A*, vol. 621, pp. 595–604, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2010.05.034 167

[280] J.-D. Leroux, M. A. Tétrault, D. Rouleau, C. M. Pepin, J.-B. Michaud, J. Cadorette, R. Fontaine, and R. Lecomte, "Time discrimination techniques using artificial neural networks for positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 56, no. 3, pp. 588–595, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2021428 167

[281] A. Mann, B. Grube, I. Konorov, S. Paul, L. Schmitt, D. P. McElroy, and S. I. Ziegler, "A sampling ADC data acquisition system for positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 53, no. 1, pp. 297–303, Feb. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.869830 167, 206

[282] B. Joly, G. Montarou, J. Lecoq, G. Bohner, M. Crouau, M. Brossard, and P.-E. Vert, "An optimal filter based algorithm for PET detectors with digital sampling front-end," *IEEE Transactions on Nuclear Science*, vol. 57, no. 1, pp. 63–70, Feb. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2031871 167, 168

[283] P. Stenstrom, A. Rillbert, F. Habte, C. Bohm, and S. A. Larsson, "Evaluation of a data acquisition system for SPECT (PET)," *IEEE Transactions on Nuclear Science*, vol. 47, no. 4, pp. 1655–1659, Aug. 2000. [Online]. Available: http://dx.doi.org/10.1109/23.873030 167

[284] J. G. Proakis and D. G. Manolakis, *Digital signal processing: principles, algorithms & applications*, 4th ed. Prentice Hall, 1996. 167

[285] M. Streun, G. Brandenburg, H. Larue, E. Zimmermann, K. Ziemons, and H. Halling, "Coincidence detection by digital processing of free-running sampled pulses," *Nuclear Instruments and Methods, Section A*, vol. 487, pp. 530–534, 2002. [Online]. Available: http://dx.doi.org/10.1016/S0168-9002(02)00401-1 168

[286] L. L. Ruckman and G. S. Varner, "Sub-10 ps monolithic and low-power photodetector readout," *Nuclear Instruments and Methods, Section A*, vol. 602, pp. 438–445, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2009.01.217 168

[287] E. Kim, K. J. Hong, J. Y. Yeom, P. D. Olcott, and C. S. Levin, "Trends of data path topologies for data acquisition systems in positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 60, no. 5, pp. 3746–3757, Oct. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2281419 168, 197, 201, 202, 210, 341

[288] W. W. Moses, "Recent advances and future advances in time-of-flight PET," *Nuclear Instruments and Methods, Section A*, vol. 580, no. 2, pp. 919–924, Oct. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2007.06.038 169, 287

[289] G. L. Brownell, C. A. Burnham, S. Wilensky, S. Aronow, H. Kazemi, and D. Strieder, "New developments in positron scintigraphy and the application of cyclotron-produced positron emitters," in *VI Proceedings of a Symposium on Medical Radioisotope Scintigraphy*, Salzburg, Austria, Aug. 1969, pp. 163–176. 169

[290] R. Gariod, R. Allemand, E. Cormoreche, M. Laval, and M. Moszynski, "The LETI positron tomograph architecture and time of flight improvements," in *Proceedings of the Workshop on Time-of-Flight Tomography*, St. Louis, MO, USA, May 1982, pp. 25–29. 169

[291] M. M. Ter-Pogossian, D. C. Ficke, M. Yamamoto, and J. T. Hood, "Super PETT I: A positron emission tomograph utilizing photon time-of-flight information," *IEEE Transactions on Medical Imaging*, vol. 1, no. 3, pp. 179–187, Nov. 1982. [Online]. Available: http://dx.doi.org/10.1109/TMI.1982.4307570 169

[292] W. H. Wong, N. A. Mullani, E. A. Phillipe, R. Hartz, and K. L. Gould, "Image improvement and design optimization of the time-of-flight PET," *Journal of Nuclear Medicine*, vol. 24, pp. 52–60, Jan. 1983. 169

[293] M. Conti, "State of the art and challenges of time-of-flight PET," *Physica Medica*, vol. 25, pp. 1–11, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.ejmp.2008.10.001 170

[294] R. F. Muzic and J. A. Kolthammer, "PET performance of the GEMINI TF: a time-of-flight PET/CT scanner," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, San Diego, CA, USA, Oct. 2006, pp. 1940–1944. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2006.354274 170

[295] B. W. Jakoby, Y. Bercier, M. Conti, M. E. Casey, T. Gremillion, C. Hayden, B. Bendriem, and D. W. Townsend, "Performance investigation of a time-of-flight PET/CT scanner," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Dresden, Germany, Oct. 2008, pp. 3738–3743. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2008.4774221 170

[296] M. Conti, L. Eriksson, and V. Westerwoudt, "Estimating image quality for future generations of TOF PET scanners," *IEEE Transactions on Nuclear Science*, vol. 60, no. 1, pp. 87–94, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2012.2233214 170

[297] T. F. Budinger, "Time-of-flight positron emission tomography: Status relative to conventional PET," *Journal of Nuclear Medicine*, vol. 24, pp. 73–78, Jan. 1983. 170

[298] D. W. Townsend and B. Bendriem, "Introduction to 3D PET," in *The theory and practice of 3D PET*, B. Bendriem and D. W. Townsend, Eds. Kluwer Academic Publishers, 1998, pp. 1–10. 172

[299] D. L. Bailey, "Quantitative procedures in 3D PET," in *The theory and practice of 3D PET*, B. Bendriem and D. W. Townsend, Eds. Kluwer Academic Publishers, 1998, pp. 55–109. 174

[300] P. R. Edholm and G. T. Herman, "Linograms in image reconstruction from projections," *IEEE Transactions on Medical Imaging*, vol. 6, no. 4, pp. 301–307, Dec. 1987. [Online]. Available: http://dx.doi.org/10.1109/TMI.1987.4307847 176

[301] J. Radon, "On the determination of functions from their integral values along certain manifolds," *IEEE Transactions on Medical Imaging*, vol. 5, no. 4, pp. 170–176, Dec. 1986, (translation of the original, published in 1917). [Online]. Available: http://dx.doi.org/10.1109/TMI.1986.4307775 176

[302] F. Natterer, *The mathematics of computerized tomography.* Society for Industrial and Applied Mathematics, 2001. 177, 179, 185

[303] W. K. Cheung and R. M. Lewitt, "Modified Fourier reconstruction method using shifted transform samples," *Physics in Medicine and Biology*, vol. 36, no. 2, pp. 269–277, 1991. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/36/2/010 177

[304] M. Defrise and P. Kinahan, "Data acquisition and image reconstruction for 3D PET," in *The theory and practice of 3D PET*, B. Bendriem and D. W. Townsend, Eds. Kluwer Academic Publishers, 1998, pp. 11–53. 178, 179, 182, 183

[305] D. L. Snyder, L. J. Thomas, and M. M. Ter-Pogossian, "A mathematical model for positron emission tomography systems having time-of-flight measurements," *IEEE Transactions on Nuclear Science*, vol. 28, no. 3, pp. 3575–3583, Jun. 1981. [Online]. Available: http://dx.doi.org/10.1109/TNS.1981.4332168 180

[306] J. G. Colsher, "Fully three-dimensional positron emission tomography," *Physics in Medicine and Biology*, vol. 25, no. 1, pp. 103–115, 1980. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/25/1/010 184

[307] K. Ishii, H. Orihara, and T. Matsuzawa, "Construction function for three-dimensional sinograms of the time-of-flight positron emission tomography," *Review of Scientific Instruments*, vol. 58, pp. 1699–1701, Sep. 1987. [Online]. Available: http://dx.doi.org/10.1063/1.1139370 185

[308] P. E. Kinahan and J. G. Rogers, "Analytic 3D image reconstruction using all detected events," *IEEE Transactions on Nuclear Science*, vol. 36, no. 1, pp. 964–968, Feb. 1989. [Online]. Available: http://dx.doi.org/10.1109/23.34585 185

[309] M. E. Daube-Witherspoon and G. Muehllehner, "Treatment of axial data in three-dimensional PET," *Journal of Nuclear Medicine*, vol. 28, no. 11, pp. 1717–1724, Nov. 1987. 186

[310] M. Defrise, P. E. Kinahan, D. W. Townsend, C. Michel, M. Sibomana, and D. Newport, "Exact and approximate rebinning algorithms for 3D PET data," *IEEE Transactions on Medical Imaging*, vol. 16, no. 2, pp. 145–158, Apr. 1997. [Online]. Available: http://dx.doi.org/10.1109/42.563660 186

[311] K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *Journal of Computer Assisted Tomography*, vol. 8, no. 2, pp. 306–316, Jan. 1984. 187

[312] J. Llacer, E. Veklerov, L. R. Baxter, S. T. Grafton, L. K. Griffeth, R. A. Hawkins, C. K. Hoh, J. C. Mazziotta, E. J. Hoffman, and C. E. Metz, "Results of a clinical receiver operating characteristic study comparing filtered backprojection and maximum likelihood estimator images in FDG PET studies," *Journal of Nuclear Medicine*, vol. 34, no. 7, pp. 1198–1203, 1993. 188

[313] C.-T. Chen, X. Ouyang, W. H. Wong, X. Hu, V. E. Johnson, C. Ordonez, and C. E. Metz, "Sensor fusion in image reconstruction," *IEEE Transactions on Nuclear Science*, vol. 38, no. 2, pp. 687–692, Apr. 1991. [Online]. Available: http://dx.doi.org/10.1109/23.289375 188

[314] T. J. Hebert and R. Leahy, "Fast methods for including attenuation in the EM algorithm," *IEEE Transactions on Nuclear Science*, vol. 37, no. 2, pp. 754–758, Apr. 1990. [Online]. Available: http://dx.doi.org/10.1109/23.106710 188

[315] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 113–119, Oct. 1982. [Online]. Available: http://dx.doi.org/10.1109/TMI.1982.4307558 189, 190

[316] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximisation algorithm," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2664–2677, Oct. 1994. 190

[317] H. Hudson and R. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE Transactions on Medical Imaging*, vol. 13, no. 4, pp. 601–609, Dec. 1994. [Online]. Available: http://dx.doi.org/10.1109/42.363108 190

[318] A. Gaitanis, G. Kontaxakis, G. Spyrou, G. Panayiotakis, and G. Tzanakos, "PET image reconstruction: A stopping rule for the MLEM algorithm based on properties of the updating coefficients," *Computerized Medical Imaging and Graphics*, vol. 34, pp. 131–141, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.compmedimag.2009.07.006 191

[319] K. Lange, "Convergence of EM image reconstruction algorithms with Gibbs smoothing," *IEEE Transactions on Medical Imaging*, vol. 9, no. 4, pp. 439–446, Dec. 1990. [Online]. Available: http://dx.doi.org/10.1109/42.61759 191

[320] S. Alenius and U. Ruotsalainen, "Bayesian image reconstruction for emission tomography based on median root prior," *European Journal of Nuclear Medicine*, vol. 24, no. 3, pp. 258–265, Mar. 1997. [Online]. Available: http://dx.doi.org/10.1007/BF01728761 191

[321] R. K. Bock and W. Krischer, *The data analysis briefbook.* Springer Verlag, 1998. 191

[322] H. Du, Y. Yang, and S. R. Cherry, "Comparison of four depth-encoding PET detector modules with wavelength shifting (WLS) and optical fiber read-out," *Physics in Medicine and Biology*, vol. 53, no. 7, pp. 1829–1842, 2008. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/53/7/002 196

[323] T. K. Lewellen, D. DeWitt, R. S. Miyaoka, and S. Hauck, "A building block for nuclear medicine imaging systems data acquisition," *IEEE Transactions on Nuclear Science*, vol. 61, no. 1, pp. 79–87, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2295037 196, 197, 203, 204, 205

[324] I. Valastyán, J. Gál, G. Hegyesi, G. Kalinka, F. Nagy, B. Király, J. Imrek, J. Molnár, M. Colarieti-Tosti, Z. Szabo, and L. Balkay, "Novel time over threshold based readout method for MRI compatible small animal PET detector," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 1295–1299. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551317 196, 197, 204

[325] M. Streun, G. Brandenburg, H. Larue, C. Parl, and K. Ziemons, "The data acquisition system of ClearPET Neuro - a small animal PET scanner," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 700–703, Jun. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.875051 196, 204

[326] P. D. Olcott, G. Chinn, and C. S. Levin, "Compressed sensing for the multiplexing of PET detectors," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Valencia, Spain, Oct. 2011, pp. 3224–3226. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6153661 196

[327] M. D. Haselman, J. Pasko, S. Hauck, T. K. Lewellen, and R. S. Miyaoka, "FPGA-based pulse pile-up correction with energy and timing recovery," *IEEE Transactions on Nuclear Science*, vol. 59, no. 5, pp. 1823–1830, Oct. 2012. [Online]. Available: http://dx.doi.org/10.1109/TNS.2012.2207403 197, 205

[328] Z. Deng and Q. Xie, "Quadratic programming time pickoff method for multivoltage threshold digitizer in PET," *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 805–813, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNS.2015.2416349 197

[329] J. D. Martinez, J. M. Benlloch, J. Cerdá, C. W. Lerche, N. Pavón, and A. Sebastiá, "High-speed data acquisition and digital signal processing system for PET imaging techniques applied to mammography," *IEEE Transactions on Nuclear Science*, vol. 51, no. 3, pp. 407–412, Jun. 2004. [Online]. Available: http://dx.doi.org/10.1109/TNS.2004.828531 198, 204

[330] G. Sportelli, N. Belcari, P. Guerra, F. Spinella, G. Franchi, F. Attanasi, S. Moehrs, V. Rosso, A. Santos, and A. del Guerra, "Reprogrammable acquisition architecture for dedicated positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 58, no. 3, pp. 695–702, Jun. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2113193 199, 202, 204

[331] G. Sportelli, N. Belcari, P. Guerra, and A. Santos, "Low-resource synchronous coincidence processor for positron emission tomography," *Nuclear Instruments and Methods, Section A*, vol. 648, pp. S199–S201, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2010.11.170 199, 200, 201

[332] C. Wang, H. Li, H. Baghaei, Y. Zhang, R. A. Ramirez, S. Liu, S. An, and W.-H. Wong, "A low-cost coincidence system with capability of multiples coincidence for high count-rate TOF or non-TOF PET cameras using hybrid method combining AND-logic and time-mark technology," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Orlando, FL, USA, Oct. 2009, pp. 3633–3638. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2009.5401842 199, 200, 205

[333] B. Collinge, A. W. Merrison, and D. Eccleshall, "A fast multiple coincidence circuit," *Journal of Scientific Instruments*, vol. 33, no. 2, pp. 72–74, 1956. [Online]. Available: http://dx.doi.org/10.1088/0950-7671/33/2/308 199

[334] F. M. C. Clemêncio, C. F. M. Loureiro, P. Fonte, and J. Landeck, "An all-digital coincidence selection and coincidence-trigger generation for a small animal RPC-PET camera," *IEEE Transactions on Nuclear Science*, vol. 60, no. 4, pp. 2912–2917, Aug. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2273416 199, 200

[335] M.-A. Tétrault, J. F. Oliver, M. Bergeron, R. Lecomte, and R. Fontaine, "Real time coincidence detection engine for high count rate timestamp based PET," *IEEE Transactions on Nuclear Science*, vol. 57, no. 1, pp. 117–124, Feb. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2009.2038055 199, 202, 205

[336] R. J. Smith and J. S. Karp, “A practical method for randoms subtraction in volume imaging PET from detector singles countrate measurements,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, San Francisco, CA, USA, Oct. 1995, pp. 992–996. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.1995.510433 199

[337] G. Hesz, D. Völgyes, B. Benyó, T. Bükki, and P. Major, “Timing calibration method for NanoPET$^{\mathrm{TM}}$/CT system,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Knoxville, TN, USA, Nov. 2010, pp. 2848–2850. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2010.5874314 200

[338] B. E. Atkins, D. R. Pressley, M. W. Lenox, B. K. Swann, D. F. Newport, and S. B. Siegel, “A data acquisition, event processing and coincidence determination module for a distributed parallel processing architecture for PET and SPECT imaging,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, San Diego, CA, USA, Oct. 2006, pp. 2439–2442. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2006.354404 200, 205

[339] J.-B. Michaud, C.-A. Brunet, M. Rafecas, R. Lecomte, and R. Fontaine, “Sensitivity in PET: neural networks as an alternative to Compton photons LOR analysis,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Honolulu, HI, USA, Oct. 2007, pp. 3594–3600. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2007.4436902 200

[340] Y. Wang, H. Li, Y. Liu, T. Xing, J. Uribe, H. Baghaei, R. Farrell, and W.-H. Wong, “A modular low dead-time coincidence system for high-resolution PET cameras,” *IEEE Transactions on Nuclear Science*, vol. 50, no. 5, pp. 1386–1391, Oct. 2003. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.817309 200, 207

[341] J. Proffitt, W. Hammond, S. Majewski, V. Popov, R. R. Raylman, A. G. Weisenberger, and R. Wojcik, “A flexible high-rate USB2 data acquisition system for PET and SPECT imaging,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Fajardo, PR, USA, Oct. 2005, pp. 2971–2975. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2005.1596955 201

[342] C. Veerappan, C. Bruschini, and E. Charbon, “Distributed coincidence detection for multi-ring based PET systems,” in *IEEE-NPSS Real Time Conference*, Nara, Japan, May 2014, pp. 1–2. [Online]. Available: http://dx.doi.org/10.1109/RTC.2014.7097478 201, 204

[343] T. Omura, T. Moriya, R. Yamada, H. Yamauchi, A. Saito, T. Sakai, T. Miwa, and M. Watanabe, “Development of a high-resolution four-layer DOI detector using MPPCs for brain PET,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 3560–3563. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551815 202, 204

[344] V. G. Zavarzin and W. A. Earle, “A 500k event/sec 12-bit ADC system with high-speed buffered PCI interface,” *IEEE Transactions on Nuclear Science*, vol. 46, no. 3, pp. 414–416, Jun. 1999. [Online]. Available: http://dx.doi.org/10.1109/23.775554 202

[345] G. Hegyesi, J. Imrek, G. Kalinka, J. Molnár, D. Novák, J. V'egh, L. Balkay, M. Emri, S. A. Kis, G. Molnár, L. Trón, I. Valastyán, I. Bagaméry, T. Bükki, S. Rósza, Z. Szabó, and A. Kerek, "Ethernet based distributed data acquisition system for a small animal PET," *IEEE Transactions on Nuclear Science*, vol. 53, no. 4, pp. 2112–2117, Aug. 2006. [Online]. Available: http://dx.doi.org/10.1109/TNS.2006.878128 202

[346] T. K. Lewellen, R. S. Miyaoka, L. R. MacDonald, M. Haselman, D. DeWitt, W. Hunter, and S. Hauck, "Design of a second generation Firewire based data acquisition system for small animal PET scanners," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Dresden, Germany, Oct. 2008, pp. 5023–5028. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2008.4774369 202

[347] E. Kim, P. Olcott, and C. Levin, "Optical network-based PET DAQ system: one fiber optical connection," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Knoxville, TN, USA, Nov. 2010, pp. 2020–2025. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2010.5874131 203

[348] M. Nakazawa, J. Ohi, T. Furumiya, T. Tsuda, M. Furuta, M. Sato, and K. Kitamura, "PET data acquisition (DAQ) system having scalability for the number of detector," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 2475–2478. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551564 203, 210

[349] C. Veerappan, C. Bruschini, and E. Charbon, "Sensor network architecture for a fully digital and scalable SPAD based PET system," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 1115–1118. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551280 203, 210

[350] M. S. Musrock, J. W. Young, J. C. Moyers, J. E. Breeding, M. E. Casey, J. M. Rochelle, D. M. Binkley, and B. K. Swann, "Performance characteristics of a new generation of processing circuits for PET applications," *IEEE Transactions on Nuclear Science*, vol. 50, no. 4, pp. 974–978, Aug. 2003. [Online]. Available: http://dx.doi.org/10.1109/TNS.2003.815307 204, 206

[351] C. J. Thompson and A. L. Goertzen, "A method for determination of the timing stability of PET scanners," *IEEE Transactions on Medical Imaging*, vol. 24, no. 8, pp. 1053–1057, Aug. 2005. [Online]. Available: http://dx.doi.org/10.1109/TMI.2005.852072 204, 206

[352] K. Ziemons, E. Auffray, R. Barbier, G. Brandenburg, P. Bruyndonckx, Y. Choi, D. Christ, N. Costes, Y. Declais, O. Devroede, C. Dujardin, A. Fedorovd, U. Heinrichs, M. Korjik, M. Krieguer, C. Kuntner, G. Largeron, C. Lartizien, H. Larue, P. Lecoq, S. Leonard, J. Marteau, C. Morel, J. B. Mosset, C. Parl, C. Pedrini, A. G. Petrosyan, U. Pietrzyk, M. Rey, S. Saladino, D. Sappey-Marinier, L. Simon, M. Streun, S. Tavernier, and J. M. Vieira, "The ClearPET$^{\mathrm{TM}}$project: development of a 2$^{\mathrm{nd}}$ generation high-performance small animal PET scanner," *Nuclear Instruments and Methods, Section A*, vol. 537, pp. 307–311, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2004.08.032 204

[353] D. P. McElroy, M. Hoose, W. Pimpl, V. Spanoudaki, T. Schüler, and S. I. Ziegler, "A true singles list-mode data acquisition system for a small animal PET scanner

with independent crystal readout," *Physics in Medicine and Biology*, vol. 50, no. 14, pp. 3323–3335, 2005. [Online]. Available: http://dx.doi.org/10.1088/0031-9155/50/14/009 204

[354] D. P. McElroy, W. Pimpl, B. J. Pichler, M. Rafecas, T. Schüler, and S. I. Ziegler, "Characterization and readout of MADPET-II detector modules: validation of a unique design concept for high resolution small animal PET," *IEEE Transactions on Nuclear Science*, vol. 52, no. 1, pp. 199–204, Feb. 2005. [Online]. Available: http://dx.doi.org/10.1109/TNS.2004.843114 204, 205

[355] J.-F. Pratte, S. Junnarkar, G. Deptuch, J. Fried, P. O'Connor, V. Radeka, P. Vaska, C. Woody, D. Schlyer, S. Stoll, S. H. Maramraju, S. Krishnamoorthy, R. Lecomte, and R. Fontaine, "The RatCAP front-end ASIC," *IEEE Transactions on Nuclear Science*, vol. 55, no. 5, pp. 2727–2735, Oct. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.2004275 204

[356] R. Fontaine, F. Bélanger, N. Viscogliosi, H. Semmaoui, M.-A. T. J.-B. Michaud, C. Pepin, J. Cadorette, and R. Lecomte, "The hardware and signal processing architecture of LabPET$^{\text{TM}}$, a small animal APD-based digital PET scanner," *IEEE Transactions on Nuclear Science*, vol. 56, no. 1, pp. 3–9, Feb. 2009. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.2007485 204, 205

[357] L. Njejimana, M.-A. Tétrault, L. Arpin, A. Burghgraeve, P. Maillé, J.-C. Lavoie, C. Paulin, K. C. Koua, H. Bouziri, S. Panier, M. W. Ben Attouch, M. Abidi, J. Cadorette, J.-F. Pratte, R. Lecomte, and R. Fontaine, "Design of a real-time FPGA-based data acquisition architecture for the LabPET II: an APD-based scanner dedicated to small animal PET imaging," *IEEE Transactions on Nuclear Science*, vol. 60, no. 5, pp. 3633–3638, Oct. 2013. [Online]. Available: http://dx.doi.org/10.1109/TNS.2013.2250307 204, 205, 207

[358] J. Imrek, G. Hegyesi, G. Kalinka, B. Király, J. Molnár, F. Nagy, I. Valastyán, L. Balkay, and Z. Szabó, "Evaluation detector module of the miniPET-3 small animal PET scanner," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 3790–3793. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551870 204, 205

[359] Q. Xie, L. Wang, J. Zhu, Y. Chen, J. Liu, M. Niu, X. Chen, Z. Wu, D. Xi, Z. Hu, B. Li, Y. Zheng, and P. Xiao, "Development and initial performance measurements of Trans-PET BioCaliburn SH1.0," in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 3090–3092. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551704 204, 205

[360] R. S. Miyaoka, X. Li, W. Hunter, L. A. P. II, W. McDougald, P. E. Kinahan, and T. K. Lewellen, "Resolution properties of a prototype continuous miniature crystal element (cMiCE) scanner," *IEEE Transactions on Nuclear Science*, vol. 58, no. 5, pp. 2244–2249, Oct. 2011. [Online]. Available: http://dx.doi.org/10.1109/TNS.2011.2165296 204

[361] W. C. J. Hunter, R. S. Miyaoka, L. MacDonald, W. McDougald, and T. K. Lewellen, "Light-sharing interface for dMiCE detectors using sub-surface laser engraving," *IEEE Transactions on Nuclear Science*, vol. 62, no. 1, pp. 27–35, Feb. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNS.2014.2374075 204

[362] E. Fysikopoulos, M. Georgiou, N. Efthimiou, S. David, G. Loudos, and G. Matsopoulos, “Fully digital FPGA-based data acquisition system for dual head PET detectors,” *IEEE Transactions on Nuclear Science*, vol. 61, no. 5, pp. 2764–2770, Oct. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNS.2014.2354984 204

[363] E. Kim, K. J. Hong, P. D. Olcott, and C. S. Levin, “PET DAQ system for compressed sensing detector modules,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 2798–2801. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551638 204

[364] T. Isobe, R. Yamada, K. Shimizu, A. Saito, K. Ote, K. Sakai, T. Moriya, H. Yamauchi, T. Omura, and M. Watanabe, “Development of a new brain PET scanner based on single event data acquisition,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Anaheim, CA, USA, Nov. 2012, pp. 3540–3543. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2012.6551810 204

[365] A. G. Weisenberger, H. Dong, B. Kross, S. J. Lee, J. McKisson, J. E. McKisson, W. Xi, C. Zorn, C. R. Howell, A. S. Crowell, L. Cumberbatch, C. D. Reid, M. F. Smith, and A. Stolin, “Development of PhytoPET: a plant imaging PET system,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Valencia, Spain, Oct. 2011, pp. 275–278. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2011.6154496 204, 207

[366] E. Charbon, C. Bruschini, C. Veerappan, L. H. C. Braga, N. Massari, M. Perenzoni, L. Gasparini, D. Stoppa, R. Walker, A. Erdogan, R. K. Henderson, S. East, L. Grant, B. Játékos, F. Ujhelyi, G. Erdei, E. Lörincz, L. André, L. Maingault, V. Reboud, L. Verger, E. G. d’Aillon, P. Major, Z. Papp, and G. Németh, “SPADnet: a fully digital, networked approach to MRI compatible PET systems based on deep-submicron CMOS technology,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Seoul, South Korea, Oct. 2013, pp. 1–9. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2013.6829025 204, 205

[367] W. W. Moses, S. Buckley, C. Vu, Q. Peng, N. Pavlov, W.-S. Choong, J. Wu, and C. Jackson, “OpenPET: A flexible electronics system for radiotracer imaging,” *IEEE Transactions on Nuclear Science*, vol. 57, no. 5, pp. 2532–2537, Oct. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2058866 204, 205

[368] M.-A. Tétrault, E. Desaulniers Lamy, A. Boisvert, C. Thibaudeau, M. Kanoun, F. Dubois, R. Fontaine, and J.-F. Pratte, “Real-time discrete SPAD array readout architecture for time of flight PET,” *IEEE Transactions on Nuclear Science*, vol. 62, no. 3, pp. 1077–1082, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1109/TNS.2015.2409783 205

[369] I. Szanda, J. Mackewn, G. Patay, P. Major, K. Sunassee, G. E. Mullen, G. Nemeth, Y. Haemisch, P. J. Blower, and P. K. Marsden, “National Electrical Manufacturers Association NU-4 performance evaluation of the PET component of the NanoPET/CT preclinical PET/CT scanner,” *Journal of Nuclear Medicine*, vol. 52, no. 10, pp. 1741–1747, Oct. 2011. [Online]. Available: http://dx.doi.org/10.2967/jnumed.111.088260 205

[370] R. Fontaine, M.-A. Tétrault, F. Bélanger, N. Viscogliosi, P. Bérard, J. Cadorette, J.-D. Leroux, J.-B. Michaud, J.-F. P. C. Pepin, S. Robert, and R. Lecomte, “Preliminary results of a data acquisition sub-system for distributed, digital, computational, APD-based, dual-modality PET/CT architecture for small animal imaging,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Rome, Italy, Oct. 2004, pp. 2296–2300. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2004.1462718 206

[371] S. Weber, A. Terstegge, H. Herzog, R. Reinartz, P. Reinhart, F. Rongen, H. W. Müller-Gärtner, and H. Halling, “The design of an animal PET: flexible geometry for achieving optimal spatial resolution or high sensitivity,” *IEEE Transactions on Medical Imaging*, vol. 16, no. 5, pp. 684–689, Oct. 1997. [Online]. Available: http://dx.doi.org/10.1109/42.640759 207

[372] I. N. Weinberg, V. Zavarzin, P. Stepanov, D. Beiline, R. Pani, G. DeVincentes, J. C. Zeng, and L. P. Adler, “Flexible geometries for hand-held PET and SPECT cameras,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, San Diego, CA, USA, Nov. 2001, pp. 1333–1336. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2001.1009750 207

[373] M. Grkovski, K. Brzezinski, V. Cindro, N. H. Clinthorne, H. Kagan, C. Lacasta, M. Mikuž, C. Solaz, A. Studen, P. Weilhammer, and D. Žontar, “Evaluation of a high resolution silicon PET insert module,” *Nuclear Instruments and Methods, Section A*, vol. 788, pp. 86–94, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.nima.2015.03.078 207

[374] J. E. Mackewn, P. Halsted, G. Charles-Edwards, R. Page, J. J. Totman, K. Sunassee, D. Strul, W. A. Hallett, M. Jauregui-Osoro, P. Liepins, S. C. R. Williams, T. Schaeffter, S. F. Keevil, and P. K. Marsden, “Performance evaluation of an MRI-compatible pre-clinical PET system using long optical fibers,” *IEEE Transactions on Nuclear Science*, vol. 57, no. 3, pp. 1052–1062, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1109/TNS.2010.2044891 207

[375] D. F. Newport, S. B. Siegel, B. K. Swann, B. E. Atkins, A. R. McFarland, D. R. Pressley, M. W. Lenox, and R. E. Nutt, “QuickSilver$^{\mathrm{TM}}$: a flexible, extensible, and high-speed architecture for multi-modality imaging,” in *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, San Diego, CA, USA, Oct. 2006, pp. 2333–2334. [Online]. Available: http://dx.doi.org/10.1109/NSSMIC.2006.354381 210

[376] *H8500/H10966 Flat panel type multianode PMT assembly*, Hamamatsu Photonics, 2011. [Online]. Available: http://www.hamamatsu.com/resources/pdf/etd/H8500_H10966_TPMH1327E.pdf 213, 273

[377] D. Calvet, “A new interface technique for the acquisition of multiple multi-channel high speed ADCs,” *IEEE Transactions on Nuclear Science*, vol. 55, no. 5, pp. 2592–2597, Oct. 2008. [Online]. Available: http://dx.doi.org/10.1109/TNS.2008.2002080 214, 215, 216

[378] L. W. Ritchey, *Right the first time: a practical handbook on high speed PCB and system design.* Speeding Edge, 2003. 214

[379] *4-channel, 12-bit, 40MSPS analog-to-digital converter with serial LVDS interface*, ADS5240, Texas Instruments, 2009. [Online]. Available: http://www.ti.com/lit/ds/symlink/ads5240.pdf 214

[380] *Quad, 12-bit, 50/65 MSPS, serial, LVDS, 3 V A/D converter*, AD9229, Analog Devices, 2010. [Online]. Available: http://www.analog.com/media/en/technical-documentation/data-sheets/AD9229.pdf 214

[381] *8-channel, 12-bit, 70MSPS analog-to-digital converter with serial LVDS interface*, ADS5273, Texas Instruments, 2009. [Online]. Available: http://www.ti.com/lit/ds/symlink/ads5273.pdf 214

[382] *Quad, 8-bit, 100 MSPS, serial LVDS 1.8 V ADC*, AD9287, Analog Devices, 2015. [Online]. Available: http://www.analog.com/media/en/technical-documentation/data-sheets/AD9287.pdf 214

[383] *Quad channel, 12-bit, 125-MSPS ADC with serial LVDS interface*, ADS6425, Texas Instruments, 2009. [Online]. Available: http://www.ti.com/lit/ds/slws197b/slws197b.pdf 215

[384] *16-bit, 105 Msps serial output ADC*, LTC2274, Linear Technologies, 2008. [Online]. Available: http://cds.linear.com/docs/en/datasheet/2274fb.pdf 215

[385] "JESD204B survival guide," Technical Article, 2013. [Online]. Available: http://www.analog.com/media/en/technical-documentation/technical-articles/JESD204B-Survival-Guide.pdf 216

[386] *Media Access Control (MAC) Parameters, Physical Layer, and Management Parameters for 10 Gb/s Operation*, IEEE Std. 802.3ae-2002, 2002. 218

[387] *Precision clock conditioner with integrated PLL*, LMK02000, National Semiconductor, 2007. [Online]. Available: http://www.ti.com/lit/ds/symlink/lmk02000.pdf 219

[388] *Virtex-5 Family Overview*, DS100, Xilinx, 2015. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds100.pdf 221

[389] *Stratix IV Device Handbook, Volume 1*, SIV5V1-4.7, Altera, 2015. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/hb/stratix-iv/stratix4_handbook.pdf 221

[390] *Arria II Device Handbook, Volume 1*, AIIGX5V1-4.6, Altera, 2014. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/hb/arria-ii-gx/arria-ii-gx_handbook.pdf 221

[391] J. G. Proakis and M. Salehi, *Fundamentals of communication systems*, 2nd ed. Pearson Education International, 2014. 231

[392] J. Williams, *Minimizing switching regulator residue in linear regulator outputs*, Application Note AN 101, Linear Technologies, Jul. 2005. [Online]. Available: http://cds.linear.com/docs/en/application-note/an101f.pdf 234

[393] *Designing power isolation filters with ferrite beads for Altera FPGAs*, Application Note AN 583, Altera, Jul. 2009. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/an/an583.pdf 234

[394] *Printed Circuit Board (PCB) Power Delivery Network (PDN) design methodology*, Application Note AN 574, Altera, May 2009. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/an/an574.pdf 235

[395] F. Carrió, V. González, E. Sanchis, D. Barrientos, J. M. Blasco, and F. J. Egea, "A capacitor selector tool for on-board PDN designs in multigigabit applications,"

in *IEEE International Symposium on Electromagnetic Compatibility (EMC)*, Long Beach, CA, USA, Aug. 2011, pp. 367–372. [Online]. Available: http://dx.doi.org/10.1109/ISEMC.2011.6038338 236, 237

[396] *Power Delivery Network (PDN) Tool User Guide*, Altera, 2009. [Online]. Available: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/ug/ug_pdn.pdf 237

[397] *Avalon Interface Specifications*, MNL-AVABUSREF, Altera, 2015. [Online]. Available: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/manual/mnl_avalon_spec.pdf 239

[398] *Triple Speed Ethernet MegaCore Function*, UG-01008, Altera, 2015. [Online]. Available: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/ug/ug_ethernet.pdf 240

[399] J. G. Proakis and M. Salehi, *Digital communications*, 5th ed. McGraw Hill, 2008. 242

[400] J. Stoer and R. Burlisch, *Introduction to numerical analysis*, 2nd ed. Springer Verlag, 1993. 254

[401] C. E. Cummings, D. Mills, and S. Golson, “Asynchronous and synchronous reset design techniques - part deux,” in *Synopsis Users Group conference*, Boston, MA, USA, 2003. [Online]. Available: http://www.sunburst-design.com/papers/CummingsSNUG2003Boston_Resets.pdf 257

[402] V. Álvarez *et al.*, “Design and characterization of the SiPM tracking system of NEXT-DEMO, a demonstrator prototype of the NEXT-100 experiment,” *Journal of Instrumentation*, vol. 8, p. T05002, May 2013. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/8/05/T05002 273

List of Figures

List of Tables

List of Acronyms

ADC Analog to Digital Converter 14, 15, 23, 24, 31, 50, 56–58, 168, 195, 197, 212–221, 223, 226–228, 231, 233, 239–248, 251, 259, 263–265, 272, 274–276, 341–343, 345

APD Avalanche Photodiode 135, 138–142, 163, 195, 204, 340

ARC Amplitude and Rise time Compensated discrimination 166, 167, 253

ASIC Application-Specific Integrated Circuit 27, 32, 41, 49, 90, 98, 139–141, 157, 158, 195, 206

ATCA Advanced Telecommunications Computing Architecture 34, 35

BGO Bismuth Germanate ($Bi_4Ge_3O_{12}$) 134, 135, 204

CDR Clock and Data Recovery 46, 47, 77

CFD Constant Fraction Discriminator 164–168, 197, 204, 253, 289

CoG Center of Gravity 148, 274

CPLD Complex Programmable Logic Device 28

CRU Clock Recovery Unit 44, 46, 47, 103, 106

CT Computed Tomography 122, 123, 339

DAC Digital to Analog Converter 104, 106, 223

DAQ Data Acquisition 1–5, 7, 15–22, 24–31, 33, 34, 37–39, 41, 49, 51, 52, 56–63, 70, 76, 89, 94–96, 100, 101, 167, 193, 195, 197, 198, 200–206, 208–212, 217, 218, 226, 230, 252, 255, 257, 263–265, 273, 281, 285, 286, 288–291, 337, 341

DCFD Digital Constant Fraction Discriminator 167, 204, 253, 271, 276, 288, 292

DDMTD Digital Dual-Mixer Time Difference 81–88, 96, 97, 99, 102, 106, 109, 114, 115, 257–259, 261, 266, 287, 290, 292–294, 338, 339, 345

DLL Delay-Locked Loop 29, 51, 83, 221

DMA Direct Memory Access 240, 292

DOI Depth of Interaction 145, 153–157, 340

DPC Discrete Positioning Circuit 147–150, 152, 156–158, 196, 340

dSiPM Digital Silicon Photomultiplier 141, 204

DSP Digital Signal Processor 29, 30

ECC Error-Correcting Code 241, 242, 245, 246

EDNA Electronic Design for Nuclear Applications group 1, 3, 5, 152, 155, 157, 158, 196, 204, 211, 212, 252, 267, 285, 288, 290

ESL Equivalent Series Inductance 235

ESR Equivalent Series Resistance 235

FBP Filtered Backprojection 178, 179, 184, 185, 187, 190

FDG Fluorodeoxyglucose 120–122

FEE Front-end Electronics 9, 20, 337

FIFO First-In First-Out 42, 43, 47, 49, 79, 91, 93, 240–242, 246, 247, 254, 255, 257, 259, 260, 337, 342

FIR Finite Impulse Response 244, 245, 343

FOV Field of View 142, 151, 154, 159, 194, 198

FPGA Field-Programmable Gate Array 28–30, 33, 35, 46, 47, 49, 56, 58, 73, 74, 81–84, 86, 89, 90, 93, 95, 97, 98, 102–105, 107, 114, 115, 152, 167, 205, 206, 213, 215, 216, 218–222, 224, 227, 230, 233, 236, 237, 239, 240, 244, 248, 251, 257, 266, 268, 270, 272, 281, 287, 290, 291, 337, 338, 342, 343, 345, 346

FRS Free-Running Sampling 15, 197, 198, 204, 289

FSM Finite State Machine 107, 110, 243, 246, 248, 254, 261, 339, 343

FWHM Full-Width at Half Maximum 114, 115, 117, 127, 144, 147, 160, 163, 166, 168, 169, 272, 280–282

FWTM Full-Width at Tenth of Maximum 278–281

GPS Global Positioning System 53, 62

GPU Graphics Processing Unit 30, 194

GSO Gadolinium Oxyorthosilicate (Gd_2SiO_5) 135

GTP GTP (Xilinx transceiver) 46, 90–93, 95, 97, 98, 103, 104, 106, 107, 109, 338

GXB Gigabit Transceiver Block (Altera transceiver) 46, 93, 103, 241–243, 342

HDL Hardware Description Language 258

HEP High-Energy Physics 8–11, 13, 16, 19, 21, 24–30, 34, 61, 63, 119, 167, 193, 195, 202, 205, 337, 345

IML Inter-Module Link 208, 209, 217–220, 223, 239, 240, 255, 258–260, 263, 264, 266, 291, 341–343, 346

IP Intellectual Property-protected core 29, 46, 79

LDO Low Dropout linear regulator 233, 234

LED Leading Edge Discriminator 164, 204

LFSR Linear Feedback Shift Register 44, 243

LHC Large Hadron Collider 22, 34, 37, 95, 98, 100

LOR Line of Response 120, 123, 127, 128, 132, 142–144, 152–155, 159, 161, 169–174, 179–181, 184–188, 190, 194, 195, 198–200, 205, 273, 340, 341

LSB Least Significant Bit 78, 92, 274, 275

LSO Lutetium Oxyorthosilicate (Lu_2SiO_5) 134, 135, 162, 163, 170, 204, 205, 273

LUT Lookup Table 21, 60, 167, 200

LVDS Low Voltage Differential Signaling 27, 29

LYSO Lutetium-Yttrium Oxyorthosilicate ($Lu_{2(1-x)}Y_{2x}SiO_5$) 135, 204, 273

MAC Medium Access Control 69, 240

MGT Multi-Gigabit Transceiver (Xilinx transceiver) 90, 95, 97

ML Maximum Likelihood 152, 153, 188, 189

ML-EM Maximum Likelihood Expectation Maximization 189–191

MPPC Multi-Pixel Photon Counter 140

MR Magnetic Resonance 122, 127, 141, 142, 194, 204

MSB Most Significant Bit 77, 78, 338

MVT Multi-Voltage Threshold 197, 204

NTP Network Time Protocol 62, 63, 65, 66, 69

PCB Printed Circuit Board 40–42, 49, 76, 204, 206, 207, 210, 215, 223, 233, 235, 241

PCS Physical Coding Sublayer 47, 337

PDN Power Distribution Network 232, 235–237, 342

PEM Positron Emission Mammography 121, 199, 204

PET Positron Emission Tomography 1–5, 7–9, 12, 61, 119–130, 132–135, 140–147, 151, 153, 159–161, 167–171, 173, 174, 176–182, 185–188, 193–195, 198–208, 210–212, 285–291, 339–341, 345

PFD Phase-Frequency Detector 81, 104, 106

PLL Phase-Locked Loop 29, 38, 41, 44, 46, 47, 51, 77–79, 81, 83, 92, 93, 97, 98, 104, 106, 108, 109, 219–221, 239, 257, 261, 266, 342

PMT Photomultiplier Tube 13, 135–142, 158, 163, 195, 196, 204, 212, 213, 271, 273–283, 288, 344

PPS Pulse Per Second signal 110

PSF Point Spread Function 147, 151, 152, 340

PSPMT Position Sensitive Photomultiplier Tube 137, 145, 147, 204, 273

PTP Precision Time Protocol 63, 65, 96

RAM Random Access Memory 28, 221, 239, 240

RDAC Resistive Digital to Analog Converter 223, 226, 239, 263, 274, 275

rms Root Mean Square 51, 63, 117, 160, 219, 274, 287

RTT Round-Trip Time 64–66, 68, 70, 97, 98, 114, 117, 268, 269

SBP Statistics-Based Positioning 152–154, 204, 340

SFP Small Form-Factor Pluggable transceptor 218, 220, 223

SiPM Silicon Photomultiplier 135, 140–142, 147, 158, 195, 204, 206, 212–214, 273–283, 288, 291, 340, 341, 344

SNR Signal to Noise Ratio 138, 139, 195, 197, 226

SoC System on Chip 29, 30, 239, 240, 263, 264, 342

SPAD Single Photon Avalanche Diode 140

SPECT Single Photon Emission Computed Tomography 122, 132

SPR Single Photon Response 163

SPS Samples per Second 104, 116

TDC Time to Digital Converter 14, 23, 97, 100, 141, 197, 199, 204, 289

ToF Time of Flight 144, 169–171, 173, 174, 180, 185, 188, 194, 199, 204–206, 210, 287, 289, 291, 341

ToT Time over Threshold 197, 204

UI Unit Interval 44, 77, 92

VCO Voltage-Controlled Oscillator 41, 104, 106, 219

VCXO Voltage-Controlled Crystal Oscillator 104, 106, 219, 220, 222

Index