



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCUELA TÉCNICA  
SUPERIOR INGENIEROS  
INDUSTRIALES VALENCIA

**TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES**

# **DISEÑO E IMPLEMENTACIÓN DE UN EQUIPO DE MEDIDA DE TEMPERATURA DE ALTA RESOLUCIÓN BASADO EN EL ANALOG FRONT-END ADS1247**

AUTOR: Víctor Serrano Aladrén

TUTOR: Julio Gomis-Tena Dolz

**Curso Académico: 2018-19**

## **AGRADECIMIENTOS**

“A mi familia, mis compañeros y mi tutor,  
por su apoyo a lo largo de todo el grado y el TFG”

## **RESUMEN**

El objetivo del presente trabajo es el diseño de un equipo de reducidas dimensiones capaz de monitorizar la temperatura en un amplio rango (desde  $-200^{\circ}\text{C}$  hasta  $850^{\circ}\text{C}$ ), con una elevada resolución (inferior a las décimas de grado) y con capacidad de envío de la información a un ordenador. Para ello se ha escogido una configuración basada en un microcontrolador que se comunicará con el PC mediante comunicación USB, que configurará el circuito integrado de adquisición (el analog front-end ADS1247) y recibirá de éste los valores de las medidas. Todo ello se realizará bajo el compromiso de un reducido consumo (algo muy importante ya que la alimentación de todo el equipo provendrá del propio conector de la comunicación USB). Como sensor de temperatura se escoge una Pt100 (si bien el circuito integrado admite también otros tipos de sensores de temperatura, tales como termopares).

La enumeración de las acciones llevadas a cabo para la implementación del sistema han sido las siguientes: diseño del esquemático del circuito electrónico, diseño de la placa de circuito impreso, montaje de los componentes, mecanizado de la caja envolvente, programación del firmware del microcontrolador y selección del programa receptor de la información en el ordenador, para su posterior almacenamiento y procesado. Todo ello ha sido llevado a cabo de manera adecuada, disponiendo en la actualidad de un prototipo funcional que ha sido empleado como banco de ensayo para la comparación de distintas formas de medir la temperatura en configuración del sensor a 2 hilos, a 3 hilos y a 4 hilos.

**Palabras Clave:** Sensor de temperatura. RTD Pt100. Instrumentación electrónica.  
Bajo consumo. Compacto.

## RESUM

L'objectiu del present treball és el disseny d'un equip de reduïdes dimensions capaç de monitoritzar la temperatura en un ampli rang (-200°C ... 850°C), amb una elevada resolució (inferior a les dècimes de grau) i amb una capacitat d'enviament de la informació a un ordinador. És per tot això que s'ha escollit una configuració basada en un microcontrolador que es comunicarà amb el PC mitjançant comunicació USB, que configurarà el circuit integrat d'adquisició (l'analog front-end ADS1247) i rebrà d'aquest els valors de les mesures. Tot això es realitzarà sota el compromís d'un consum reduït (una cosa molt important ja que l'alimentació de tot l'equip provindrà del propi connector de la comunicació USB). Com sensor de temperatura es tria una Pt100 (si bé el circuit integrat admet també altres tipus de sensors de temperatura, com ara els termoparells).

L'enumeració de les accions dutes a terme per a la implementació del sistema han estat les següents: disseny de l'esquemàtic del circuit electrònic, disseny de la placa de circuit imprès, muntatge dels components, mecanització de la caixa envoltant, programació del firmware del microcontrolador i selecció del programa receptor de la informació a l'ordinador, per al seu posterior emmagatzematge i processat. Tot això ha estat dut a terme de manera adequada, disposant en l'actualitat d'un prototip funcional que ha estat emprat com a banc d'assaig per a la comparació de diferents formes de mesurar la temperatura en configuració del sensor a 2 fils, a 3 fils i a 4 fils.

**Paraules clau:** Sensor de temperatura. RTD Pt100. Instrumentació electrònica. Baix Consum. Compacte.

## **ABSTRACT**

The objective of the present work is the design of a device of reduced dimensions capable of monitoring the temperature in a wide range (-200°C ... 850°C), with a high resolution (less than tenths of a degree) and with the ability to send information to a computer. For this purpose, a configuration based on a microcontroller has been chosen that will communicate with the PC through USB communication, which will configure the integrated acquisition circuit (the ADS1247 front-end analog) and will receive the measurement values from it. All this will be done under the commitment of a reduced consumption (something very important since the power of all the equipment will come from the connector of the USB communication itself). A Pt100 is chosen as the temperature sensor (although the integrated circuit also supports other types of temperature sensors, such as thermocouples).

The enumeration of the actions carried out for the implementation of the system have been the following: design of the schematic of the electronic circuit, design of the printed circuit board, assembly of the components, machining of the enclosure, programming of the firmware of the microcontroller and selection of the program receiving the information in the computer, for its later storage and processing. All this has been carried out in an appropriate way, currently having a functional prototype that has been used as a test bench for the comparison of different ways of measuring the temperature in the configuration of the sensor with 2-wire, 3-wire and 4-wire connection.

**Keywords:** Temperature Sensor. Pt100. Electronic instrumentation. Low power. Compact.

# ÍNDICE

## Documentos contenidos en el TFG

- Memoria
- Presupuesto
- Planos
- Anexo

## Índice de la Memoria

CAPÍTULO 1. INTRODUCCIÓN.....	1
1.1. Objetivos del documento .....	1
1.1.1 Objetivo general.....	1
1.1.2 Objetivos específicos.....	1
1.1.3 Motivación.....	2
1.1.4 Diagrama de bloques del sistema .....	2
1.2 Análisis de los distintos tipos de sensores de temperatura.....	3
1.2.1 Termopares .....	3
1.2.2 Termorresistencias (RTD) .....	4
1.2.3 Termistores .....	5
1.2.4 Tipo de sensor empleado como base del diseño .....	6
1.3 Análisis en cuanto a la electrónica de adquisición de datos y microcontrolador .....	6
CAPÍTULO 2. DISEÑO DEL ESQUEMA ELECTRÓNICO .....	8
2.1 Diseño del esquema electrónico del analog front-end para la medida del voltaje de la Pt100 .....	8
2.1.1 Esquema electrónico para la lectura de una Pt100 de 2 hilos .....	9
2.1.2 Esquema electrónico para la lectura de una Pt100 de 3 hilos .....	10
2.1.3 Esquema electrónico para la lectura de una Pt100 de 4 hilos .....	11

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

---

2.1.4 Esquema electrónico para la lectura de cualquier tipo de Pt100 .....	12
2.1.5 Elección de los componentes siguiendo los requisitos del ADS1247.....	13
2.2 Diseño del esquema electrónico para la recepción del voltaje desde el ADS1247 y envío de la temperatura.....	16
2.3 Listado de componentes utilizados.....	17
CAPÍTULO 3. DISEÑO DEL CIRCUITO IMPRESO.....	18
CAPÍTULO 4. PROGRAMACIÓN DEL SOFTWARE.....	19
4.1 Partes del código presente en el archivo Pt100.c.....	19
4.1.1 Configuración del microcontrolador .....	19
4.1.2 Procesamiento de los datos provenientes del ADS1247 .....	21
4.1.3 Función principal.....	26
4.2 Partes del código presente en el archivo ADS1247.h .....	27
4.2.2 Definiciones para el ADS1247 .....	27
4.2.2 Función para la configuración del ADS1247 variable según el modo de funcionamiento.....	28
4.2.2 Funciones para la configuración del ADS1247 no dependientes del modo de funcionamiento.....	34
4.3 Entorno de trabajo .....	36
4.4 Visualización de los datos desde el ordenador .....	36
CAPÍTULO 5. DISEÑO DEL PROTOTIPO Y MONTAJE .....	37
CAPÍTULO 6. ENSAYOS Y AJUSTES .....	40
CAPÍTULO 7. CONCLUSIONES.....	45
REFERENCIAS.....	46

## Índice de figuras

Figura 1: Diagrama de bloques de los pasos necesarios para obtener la temperatura.....	2
Figura 2: Estructura básica de un termopar [1] .....	3
Figura 3: Curva característica de una RTD de tipo Pt100 según el documento Europeo EN 60751:2008 .....	4
Figura 4: Tipos de conexiones RTD [2] .....	5
Figura 5: Termistor de tipo NTC fabricado por DACHS [3] .....	5
Figura 6: Exterior del circuito integrado de alta precisión ADS1247 [4] .....	6
Figura 7: Exterior del microcontrolador PIC24FJ64GA004 [5] .....	7
Figura 8: Esquema de los componentes del ADS1247 [7].....	8
Figura 9: Esquema eléctrico para la conexión de una RTD de 2 hilos.....	9
Figura 10: Esquema eléctrico para la conexión de una RTD de 3 hilos.....	10
Figura 11: Esquema eléctrico para la conexión de una RTD de 4 hilos.....	11
Figura 12: Esquema electrónico del analog front-end compatible con todo tipo de Pt100 (extracto del plano 1).....	12
Figura 13: Esquema electrónico de la parte de procesamiento de información y comunicación (extracto del plano 1).....	16
Figura 14: Plano de conjunto del prototipo (extracto del plano 5) .....	37
Figura 15: Terminación del cable micro USB magnetizado Fonken de un metro .....	38
Figura 16: Imagen del prototipo con la tapa por el lado que se conecta el micro USB .....	38
Figura 17: Imagen del prototipo ya montado sin la tapa.....	39
Figura 18: Imagen del prototipo con la tapa por el lado que se conecta la Pt100 .....	39
Figura 19: Preparación del ensayo de la Pt100 de 4 hilos con agua a 0°C.....	40
Figura 20: Sensor de temperatura Pt100 4 hilos RS PRO de clase B.....	41
Figura 21: Sensor de temperatura Pt100 de 3 hilos.....	41
Figura 22: Ensayo con agua a 0°C de la Pt100 de 3 hilos.....	42
Figura 23: Preparación del ensayo de la Pt100 de 4 hilos con agua a 100°C.....	42

### Índice de tablas:

Tabla 1: Posición de los jumpers para cada configuración .....	13
Tabla 2: Resultados para el cálculo de la ganancia máxima del amplificador en cada configuración.....	14
Tabla 3: Resultados de la comprobación del voltaje en el nodo común de la RTD a 850°C.....	15
Tabla 4: Resultados de la comprobación del voltaje en el nodo común de la RTD a -200°C .....	15
Tabla 5: Componentes electrónicos necesarios para construir la placa de circuito impreso.....	17
Tabla 6: Rango de la escala del amplificador de ganancia variable del ADS1247 para cada configuración.....	25
Tabla 7: Bits del registro de control del multiplexador 0.....	28
Tabla 8: Bits del registro de control de la tensión de polarización .....	29
Tabla 9: Bits del registro de control del multiplexador 1.....	29
Tabla 10: Bits del registro de control del sistema 0 .....	30
Tabla 11: Bits del registro de control de IDAC 0.....	30
Tabla 12: Bits del registro de control de IDAC 1.....	31
Tabla 13: Ensayo del dispositivo midiendo agua a 0°C.....	43
Tabla 14: Ensayo del dispositivo midiendo agua a 100°C.....	43

### Índice del Presupuesto

1 Cuadros de precios .....	1
1.1 Materiales .....	1
1.2 Mano de obra .....	2
1.3 Equipos amortizados linealmente .....	2
2 Cuadro de precios descompuestos .....	2
3 Cuadro de precios unitarios .....	4
4 Cuadro de presupuesto base de licitación .....	5

### Índice de tablas:

Tabla 1: Cuadro de precios de los materiales.....	1
Tabla 2: Cuadro de precios de la mano de obra.....	2
Tabla 3: Cuadro de precios de los equipos de uso habitual.....	2
Tabla 4: Cuadro de precios descompuestos.....	2

Tabla 5: Cuadro de precios unitarios..... 4

Tabla 6: Cuadro de presupuesto base de licitación ..... 5

## Índice de los Planos

Esquema electrónico ..... 1

Situación de los componentes en la placa de circuito impreso..... 2

Pistas en la cara superior de la placa de circuito impreso..... 3

Pistas en la cara inferior de la placa de circuito impreso ..... 4

Ensamblaje de las piezas del prototipo..... 5

## Índice del Anexo

1 Pt100.c ..... 1

    1.1 Configuración 4 hilos  $V_{REF}$ ..... 1

    1.2 Modificaciones para el resto de las configuraciones ..... 6

2 ADS1247.h ..... 7

    2.1 Configuración 4 hilos  $V_{REF}$  ..... 7

    2.2 Modificaciones para el resto de las configuraciones ..... 10

        2.2.1 Función ADS\_init para la configuración 4 hilos  $R_{REF}$  ..... 10

        2.2.2 Función ADS\_init para la configuración 3 hilos  $V_{REF}$  (con y sin  $R_{COMP}$ ) ..... 11

        2.2.3 Función ADS\_init para la configuración 3 hilos  $V_{REF}$  (con y sin  $R_{COMP}$ ) ..... 11

---

DOCUMENTO N°1:

**MEMORIA**  
**DESCRIPTIVA**

---

# **CAPÍTULO 1. INTRODUCCIÓN**

## **1.1. Objetivos del documento**

### **1.1.1 Objetivo general**

El objetivo de este proyecto es el diseño de un equipo de dimensiones reducidas y bajo consumo capaz de monitorizar una temperatura en un entorno determinado, con una elevada resolución inferior a las décimas de grado y que pueda enviar esta información a un ordenador mediante USB. Se buscará además una gran flexibilidad en el mismo para que una vez construido el prototipo se puedan comparar distintas formas de medir la temperatura con él. Para todo ello, se deben tener en cuenta 4 objetivos específicos.

### **1.1.2 Objetivos específicos**

1. Selección de un tipo de sensor de temperatura que permita tanto un rango de temperatura medibles lo más amplio posible, preferiblemente desde  $-200^{\circ}\text{C}$  hasta  $850^{\circ}\text{C}$ , como una resolución del orden de centésimas de grado. Se buscará a su vez que su instalación sea fácil y rápida, pudiendo cambiarse fácilmente el dispositivo de entorno de trabajo.
2. Diseño de un circuito electrónico para la conversión de la señal analógica del sensor en una digital que podrá ser procesada por el microcontrolador para su posterior envío. El conjunto de componentes electrónicos que lo formen, así como los de la parte del microcontrolador, deberán de poder alimentarse con tensiones de 3,3 V generadas gracias a la energía procedente del USB del ordenador que recibe la información y deberán de ser suficientemente pequeños para que el equipo sea compacto. Debido a la amplia variabilidad de entornos en los que puede ser utilizado el dispositivo, es necesario que el sensor de temperatura pueda ser intercambiable por otros del mismo tipo pero con características adaptadas a distintos medios como su resistencia química o mecánica.
3. Elaboración de la programación para el microcontrolador elegido el cual es imprescindible para la transmisión de datos al ordenador. Se deberán realizar programas para el mayor número de sensores posibles y las distintas formas de tomar los datos de ellos. Gracias a este enfoque centrado en la flexibilidad del sistema, se podrán comparar entre ellos y optar por la solución más acertada en cada escenario.
4. Construcción de un prototipo funcional que incluya tanto la parte electrónica como una cobertura de protección. La caja exterior deberá de ser lo más ajustada posible a la electrónica resultante y deberá de estar personalizada para que todo encaje perfectamente sin haber elementos externos innecesarios que causen confusión.

### 1.1.3 Motivación

La electrónica es una rama de la ingeniería con un peso creciente en nuestra sociedad, necesaria para su funcionamiento, así como en continuo crecimiento. Para conseguir un acercamiento a esta disciplina a través de un Trabajo Final de Grado se ha escogido un proyecto que permita la aplicación práctica de diversos conceptos e ideas vistas a lo largo del grado en ingeniería en tecnologías industriales. Estos conocimientos en los que se quiere profundizar incluyen desde el propio diseño de circuitos electrónicos, pasando por la programación, hasta la ejecución de un proyecto físico. En estas actividades al estar estrechamente relacionadas con el mercado laboral la capacidad de resolución de los problemas se pone a prueba constantemente.

En concreto, se decidió realizar una exploración de distintas facetas de la instrumentación industrial hasta que finalmente se encontró que la detección de temperatura era la más idónea gracias a su importancia y su presencia tanto en el ámbito industrial como el doméstico. Realizar un dispositivo que cuantificase la temperatura gracias a un sensor no solo incluía todos los conocimientos anteriormente citados puestos en práctica, sino que también planteaba la necesidad de optimización tanto de la energía consumida como en el espacio usado, ambos factores claves en la electrónica actual y futura.

La importancia del control de la temperatura puede llegar a ser crítica en determinados procesos industriales, como por ejemplo en la conservación de alimentos o en la industria química. Otro caso muy relevante es el control de temperatura en el transporte de órganos para trasplantes. En cualquiera de estos casos, es necesario que la adquisición de la temperatura sea precisa y fiable, por lo que es necesario un amplio conocimiento de los aparatos que se usan y es igualmente fundamental poder elegir adecuadamente estos según los requerimientos concretos. En este sentido, en el presente TFG se opta por un enfoque orientado a la flexibilidad con unos ensayos entre distintos métodos a la hora de adquirir la temperatura para compararlos de primera mano y ver cómo se comportan, ayudando a una mejor comprensión de esta disciplina de la electrónica.

### 1.1.4 Diagrama de bloques del sistema

Para mostrar de forma más visual el funcionamiento del sistema en su conjunto, a continuación, se presenta un diagrama de bloques en la Figura 1. En él se presenta de forma ilustrativa los diferentes pasos que se dan para obtener la temperatura ambiente a partir del sensor.

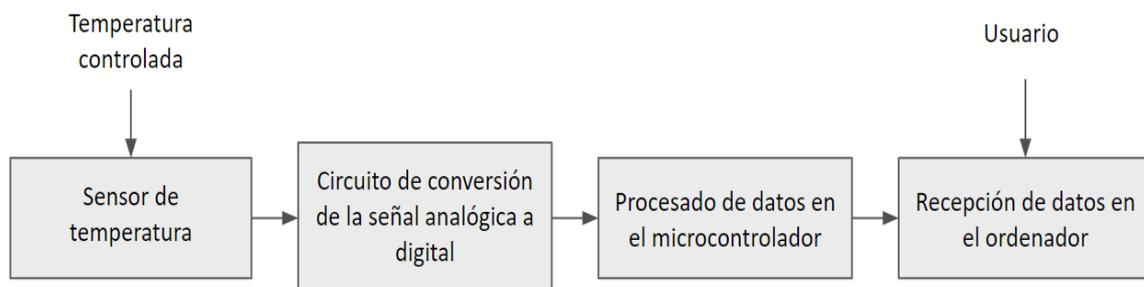


Figura 1: Diagrama de bloques de los pasos necesarios para obtener la temperatura

## 1.2 Análisis de los distintos tipos de sensores de temperatura

Un sensor de temperatura es aquel dispositivo que detecta la temperatura de un objeto o ambiente y es capaz de transformarla en una señal eléctrica que es posible interpretar con un sistema electrónico. Para el diseño de este sistema es fundamental escoger para que tipo de sensor térmico va a ir enfocado al requerir cada uno unos circuitos distintos para su correcto funcionamiento. Existen 3 categorías principales dentro de estos sensores según su funcionamiento: termopares, termorresistencias (RTD) y termistores.

### 1.2.1 Termopares

Los termopares son los sensores más empleados en la industria gracias a ser económicos, diversos rangos de temperatura y fácilmente intercambiables. Su funcionamiento se basa en la unión de dos hilos metálicos de distintos metales unidos por uno de los extremos llamado junta de medición o caliente, como se puede observar en la Figura 2. En el otro extremo, llamado junta fría, se genera un voltaje del orden de milivoltios debido a la temperatura, siendo este mayor cuando aumenta la misma.

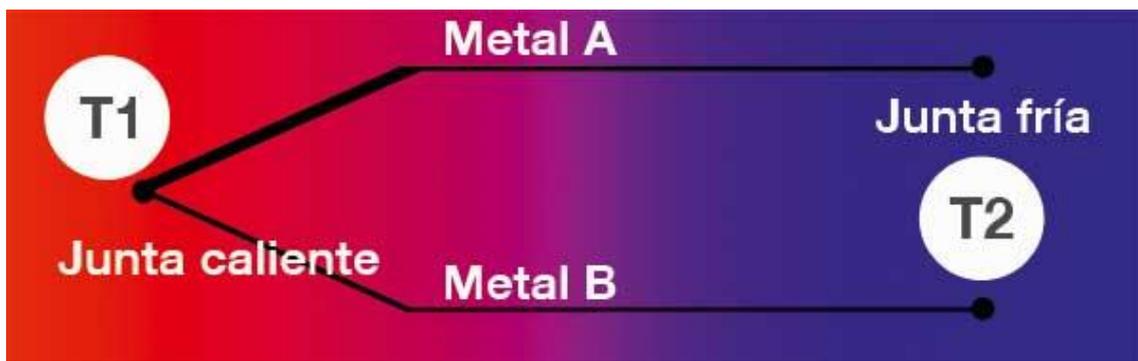


Figura 2: Estructura básica de un termopar [1]

Existen diversos tipos de termopares los cuales tienen diferentes características, entre ellos los más habituales son:

- Tipo J: Está formado por una combinación de hierro y constantan (aleación de cobre y níquel). Su rango de temperaturas se sitúa entre  $-40^{\circ}\text{C}$  y  $760^{\circ}\text{C}$ . Destaca por su bajo coste, pero tiene como desventajas su alta oxidación a más de  $500^{\circ}\text{C}$  y su fragilidad por debajo de  $0^{\circ}\text{C}$ .
- Tipo T: Al sustituir el hierro de los termopares Tipo J por cobre consigue un gran rendimiento en entornos con humedad, pero a cambio aumenta su coste. Su rango de temperaturas está entre  $-200^{\circ}\text{C}$  y  $350^{\circ}\text{C}$ .
- Tipo K: Compuesto de chromega (aleación de cromo y níquel) junto a alomega (aleación de aluminio y níquel). Es el más usado gracias a su amplio rango de temperatura (entre  $-200^{\circ}\text{C}$  y  $1250^{\circ}\text{C}$ , pese a que normalmente no se recomienda por debajo de  $300^{\circ}\text{C}$ ) y la posibilidad de usarlo en cualquier tipo de atmósfera. Su principal desventaja es que es más voluminoso que el resto.
- Tipo E: Formado por chromega y constantan y con un rango de entre  $-200^{\circ}\text{C}$  y  $900^{\circ}\text{C}$ , es muy recomendable usarlo para medir por debajo de  $0^{\circ}\text{C}$ .

La principal desventaja que los caracteriza es su limitada exactitud, lo cual propicia una gran cantidad de error al trabajar con precisiones inferiores al grado centígrado. Para minimizarlo lo máximo posible será necesaria la compensación de la unión fría. Entre las desventajas de los termopares también se encuentra su tiempo de respuesta, menor que los de las otras categorías.

### 1.2.2 Termorresistencias (RTD)

Un sensor de temperatura basado en una termorresistencia consiste en una resistencia formada por un material el cual varía su resistencia eléctrica en función de la temperatura. Su estructura suele estar compuesta por un alambre enrollado con un núcleo de cerámica o vidrio envolviéndolo. Destacan por su elevada adaptabilidad, tanto por los rangos de temperatura como por los cabezales. Una de las características que los hacen más atractivos para su uso industrial es su inmunidad ante el ruido eléctrico, siempre presente en las industrias con gran cantidad de aparatos eléctricos. Las termorresistencias más habituales son las compuestas de platino (Pt) al ser el material más exacto y estable consiguiendo una gran linealidad en su comportamiento entre  $-200^{\circ}\text{C}$  y  $850^{\circ}\text{C}$ , como se puede ver en la Figura 3.

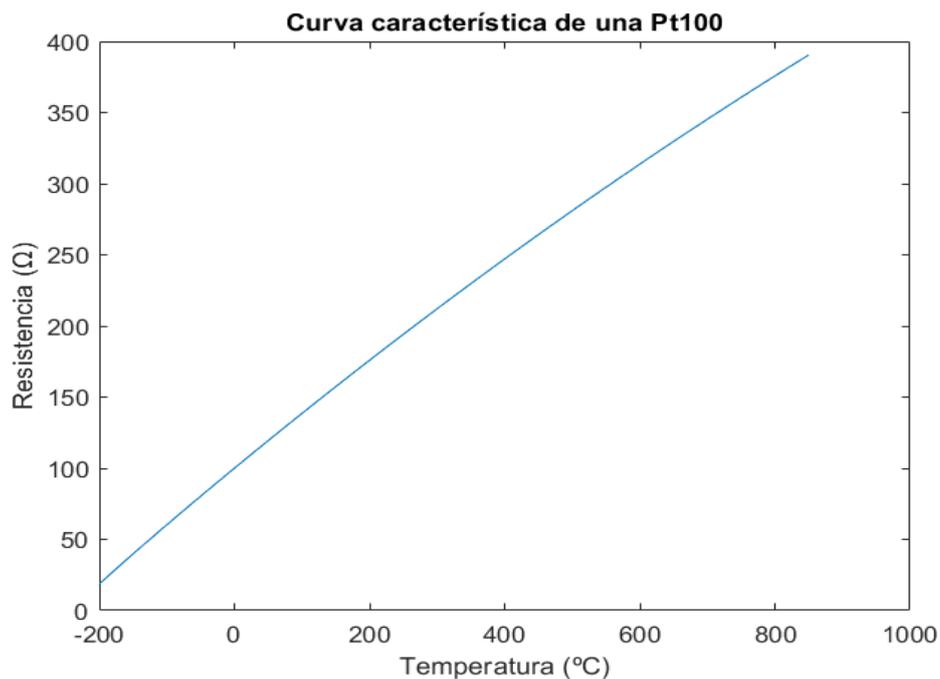


Figura 3: Curva característica de una RTD de tipo Pt100 según el documento Europeo EN 60751:2008

Las RTD suelen calibrarse para que a  $0^{\circ}\text{C}$  tengan  $100\Omega$ , de ahí que una termorresistencia de platino calibrada de esa manera se denomine Pt100. Existen otros materiales como el níquel y cobre para la construcción de termorresistencias. Sin embargo, su principal ventaja respecto a las de platino, que es su mayor linealidad, es fácilmente corregible empleando un vector de referencia en vez de una ecuación lineal para realizar la conversión de resistencia a temperatura. Por el contrario, tienen como desventaja un rango de temperaturas inferior.

Las RTD se pueden encontrar con 3 tipos de configuración según los cables que se conectan a la resistencia (Figura 4):

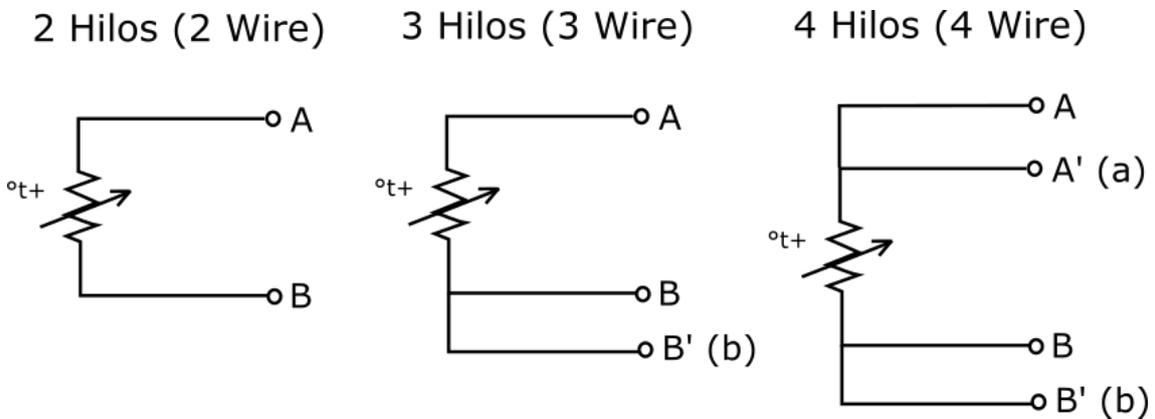


Figura 4: Tipos de conexiones RTD [2]

- Circuito bifilar: Es el más sencillo de todos, pero se deben emplear cables de conexión cortos para que su resistencia no altere el valor de la medida de manera variable con la temperatura.
- Circuito trifilar: Es el más extendido para cables de hasta 20 metros debido a la cancelación casi total de la resistencia de dichos cables siempre que sean de la misma longitud y estén a la misma temperatura.
- Circuito tetrafilar: El diseño más complejo y costoso de todos que garantiza una anulación de error por los cables de conexión prácticamente nulo incluso en instalaciones con distancias superiores a 20 metros

### 1.2.3 Termistores

Los termistores (Figura 5) son sensores térmicos que como las RTD son resistencias variables en función de la temperatura pero que al estar formados por semiconductores carecen de la linealidad característica de las termorresistencias. A cambio, poseen una respuesta rápida y son muy sensibles siempre que escojas rangos de temperaturas más limitados. Pueden ser de tipo NTC si su resistencia disminuye al aumentar con la temperatura o PTC si es al contrario.

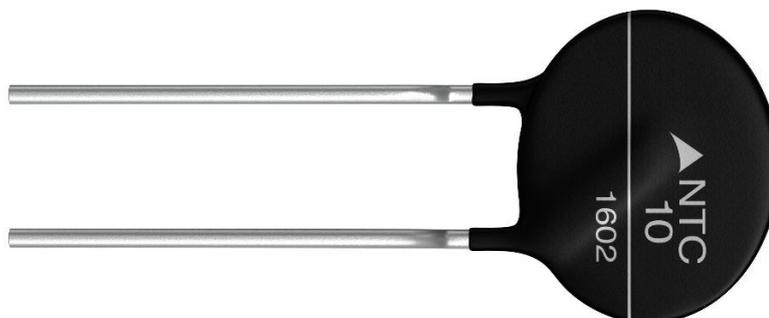


Figura 5: Termistor de tipo NTC fabricado por DACHS [3]

El uso más frecuente de los termistores de tipo NTC se sitúa en el rango de 100°C-150°C por ejemplo para alarmas de temperatura mientras que los de tipo PTC se utilizan en cámaras frigoríficas o donde la velocidad de respuesta sea muy importante. Todos los termistores se caracterizan por ocupar muy poco tamaño por lo que también pueden ser usados incorporados en placas electrónicas donde esta característica es fundamental.

#### 1.2.4 Tipo de sensor empleado como base del diseño

Tras comparar todos los tipos de sensores y viendo los campos donde más extendido está el uso de cada uno se ha concluido que el sistema deberá diseñarse para el uso de una RTD. Esto es debido principalmente a la gran flexibilidad de este tipo de sensores y al hecho de que se puede hacer el diseño fácilmente compatible con los 3 tipos de conexiones particulares a estos sensores. Por tanto, el diseño permitirá usar RTD con conexiones de 2, 3 y 4 hilos. En concreto, se ha seleccionado la Pt100 al ser la más utilizada y con unas posibilidades tanto de exactitud como de rango muy interesantes y que cubren prácticamente todos los casos posibles.

### 1.3 Análisis en cuanto a la electrónica de adquisición de datos y microcontrolador

A la hora de plantear un sistema de adquisición de temperaturas una opción sería partir de una placa de circuito impreso ya diseñada con su propio microprocesador como podría ser un Arduino. Esta solución, frente a un sistema diseñado para este uso en concreto empleando varios chips, pese a ser más sencilla y rápida de ejecutar se debe descartar al no cumplir con los objetivos deseados de tamaño, flexibilidad y exactitud de las medidas.

Por tanto, se necesita escoger en primer lugar un microchip que sea capaz de alimentar y medir el voltaje de la Pt100 para su posterior conversión y envío en formato digital. Entre todas las opciones del mercado se ha escogido el circuito integrado de alta precisión ADS1247 fabricado por Texas Instruments (Figura 6).

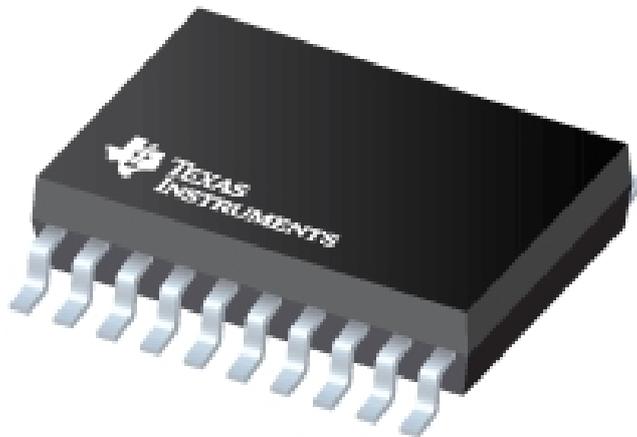


Figura 6: Exterior del circuito integrado de alta precisión ADS1247 [4]

Entre sus características destacan el amplificador de ganancia programable de bajo ruido, 4 entradas independientes, voltaje interno de referencia con poca variación en función de la temperatura (10ppm/°C) y un convertidor de analógico a digital con una resolución de hasta 24 bits. Todo esto dentro de un coste muy competitivo y un tamaño apropiado.

Para acompañarlo será necesario un microprocesador que gestione la información del voltaje de la Pt100 que reciba del ADS1247 y tras convertirla a temperatura la envíe por interfaz TTL al ordenador para su visualización. Finalmente se opta por un microcontrolador PIC24FJ64GA004 (Figura 7) fabricado por Microchip de 16 bits, capaz de controlar y configurar correctamente el ADS1247.

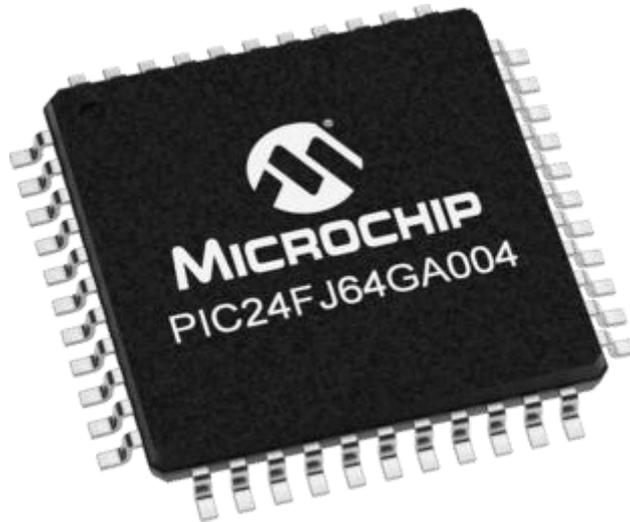


Figura 7: Exterior del microcontrolador PIC24FJ64GA004 [5]

Por último, para que el ordenador pueda recibir esta información por micro USB será necesario un convertidor de TTL a micro USB conectado al microcontrolador. Esto es debido a que el microcontrolador no tiene salida a micro USB de manera directa. Será desde este cable desde donde se alimente todo el dispositivo.

## **CAPÍTULO 2. DISEÑO DEL ESQUEMA ELECTRÓNICO**

El esquema electrónico del proyecto está compuesto por dos partes diferenciadas: la parte de adquisición del voltaje de la Pt100 y la parte del envío de la temperatura al ordenador realizada por el microcontrolador y el convertidor de TTL a micro USB. Ambas deberán de ser después unidas en la placa de circuito impreso. A continuación, se procede con el diseño de este circuito electrónico usando la herramienta gratuita EAGLE programada por Autodesk y que permite tanto el diseño del esquemático electrónico como la disposición de todos los componentes y uniones en el PCB.

### **2.1 Diseño del esquema electrónico del analog front-end para la medida del voltaje de la Pt100**

Como se ha expuesto previamente en los objetivos, se busca que el sistema tenga la máxima flexibilidad posible a la hora de usar distintos sensores según el escenario de uso que se manifieste. Por ello, tras seleccionar que el sensor de temperatura será una Pt100, el sistema deberá poder funcionar correctamente con las 3 conexiones posibles que puede usar una Pt100: 4, 3 y 2 hilos. Por lo tanto, en primer lugar se va a plantear cómo leer una Pt100 en cada uno de estos casos correctamente empleando el ADS1247. Para el envío de corrientes eléctricas y medida de voltaje en la Pt100 se emplearán las entradas AIN0, AIN1, AIN2 y AIN3 del ADS1247 (Figura 8)

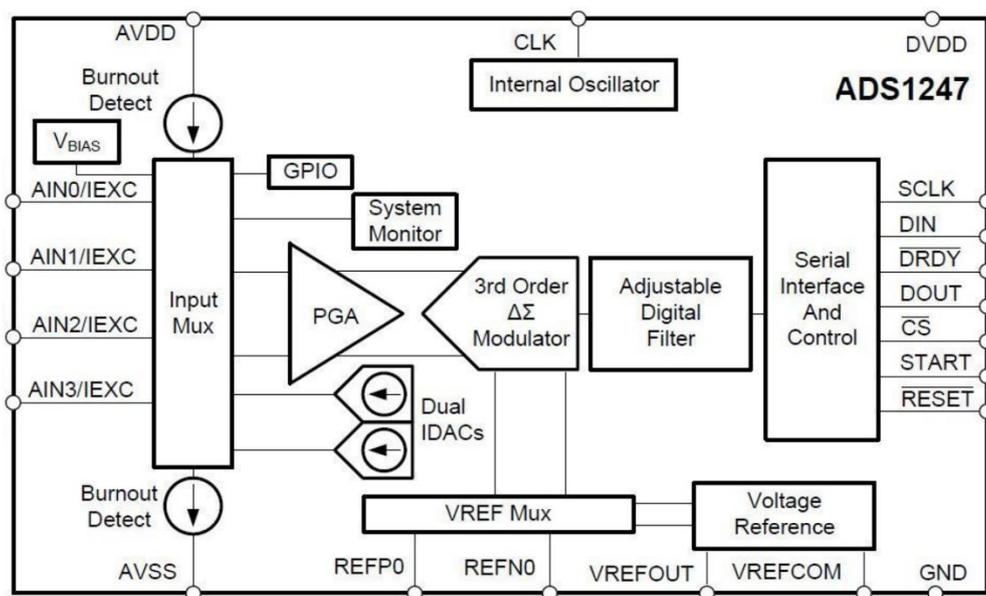


Figura 8: Esquema de los componentes del ADS1247 [7]

### 2.1.1 Esquema electrónico para la lectura de una Pt100 de 2 hilos

La estructura básica consiste en la circulación de una corriente eléctrica ( $I_{DAC}$ ) proveniente de AIN0 por la RTD. Esta corriente genera una diferencia de potencial que es igual a la que se genera entre AIN1 y AIN2, la cual deberá leer el ADS1247 usando su ADC (convertidor de analógico a digital) interno previo paso por el PGA (amplificador de ganancia programable).

Este método de medición no compensa el efecto de la resistencia de los cables, el cual es también variable con la temperatura del ambiente y por tanto añade más complejidad al error resultante. Sin embargo, es posible corregir otro suceso característico que produce error en la medición de voltajes como es el ruido eléctrico. El ruido eléctrico se manifiesta cuando, sea desde el propio sistema o sea desde el exterior, se introduce una energía eléctrica a ciertas frecuencias en el circuito que altera las mediciones por parte del ADC. Para solucionarlo se sitúa, entre la RTD y la conexiones donde se mide la diferencia de potencial de la misma, un circuito RC (Figura 9) que sea capaz de bloquear gran parte de estas frecuencias no deseadas.

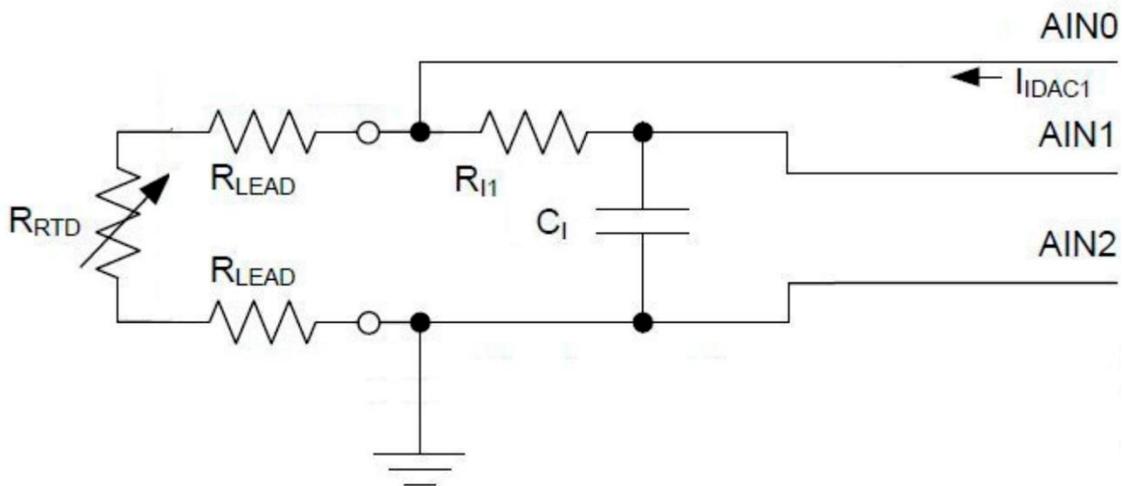


Figura 9: Esquema eléctrico para la conexión de una RTD de 2 hilos

El circuito RC en este caso solo cuenta con una resistencia y un condensador que forma un filtro diferencial capaz de mitigar ruidos que se tengan una entrada y una salida del sistema, pudiendo ser producidos tanto dentro del propio equipo como inducidos desde la alimentación de este. Según el tamaño del condensador filtrará frecuencias formando un filtro paso bajo permitiendo que la parte continua de la corriente eléctrica pase por la RTD y la parte con mayores frecuencias pase por el condensador.

### 2.1.2 Esquema electrónico para la lectura de una Pt100 de 3 hilos

La configuración de 3 hilos en Pt 100 permite, siempre que sus cables tengan una longitud y propiedades iguales, corregir en gran parte el error que estos producen (representado como la resistencia de los cables  $R_{LEAD}$ ) en la medida. Para ello es necesario que haya dos intensidades iguales provenientes de AIN0 y AIN3, solamente circulando por la RTD la proveniente de AIN0 (Figura 10). Como en la conexión anterior, el voltaje medido es el que existe entre AIN1 Y AIN2 que es igual al de la RTD.

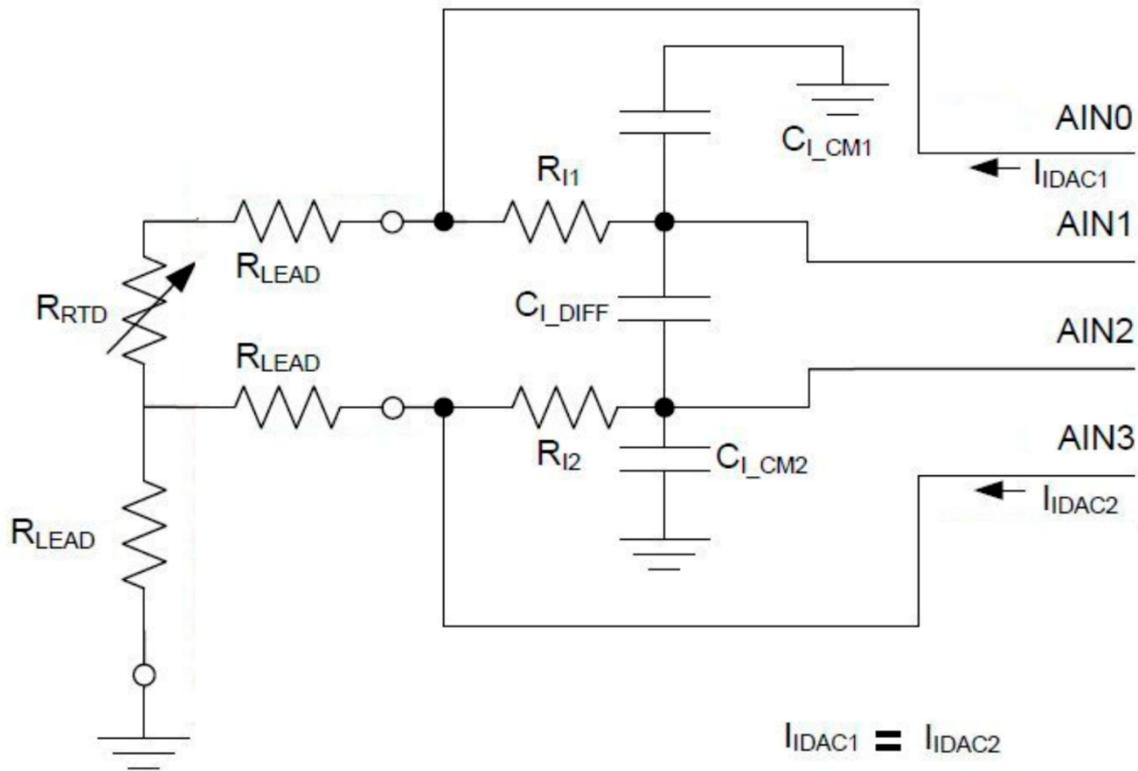


Figura 10: Esquema eléctrico para la conexión de una RTD de 3 hilos

En esta ocasión también se puede filtrar el ruido eléctrico y se ha optado por un circuito RC situado en la misma localización que con la conexión de 2 hilos pero que para generar un filtro de nodo común complementario al filtro diferencial cuenta con 2 resistencias y 3 condensadores. El filtro diferencial funciona igual que el anterior mientras que el filtro de nodo común al conectar dos condensadores a tierra, uno en cada línea, es capaz de eliminar el ruido eléctrico que entra al sistema desde el exterior (generados externamente fuera del equipo) por ambos sentidos y que deben de ser derivados a Tierra irremediablemente.

El filtro de nodo común, igual que el diferencial, funciona como uno de tipo paso bajo permitiendo según el tamaño del condensador el paso de todas las frecuencias inferiores a una determinada. Para el dispositivo se configurará una frecuencia de corte varias veces superior para el filtro de nodo común ya que este tipo de ruido es menos constante y si se presenta suele tener más frecuencia que el generado por el propio sistema o la alimentación.

### 2.1.3 Esquema electrónico para la lectura de una Pt100 de 4 hilos

Para la conexión de 4 hilos (Figura 11) se mantiene el circuito RC que forma el doble filtro (diferencial y de nudo común) pero se vuelve a tener únicamente una corriente que pasa por la RTD saliendo de AIN0. Es este caso no es necesario el uso de AIN3 al no necesitar una segunda corriente eléctrica. Este tipo de conexión permite corregir totalmente el error causado por la resistencia de los cables, permitiendo una mayor distancia entre el ADS1247 y la RTD.

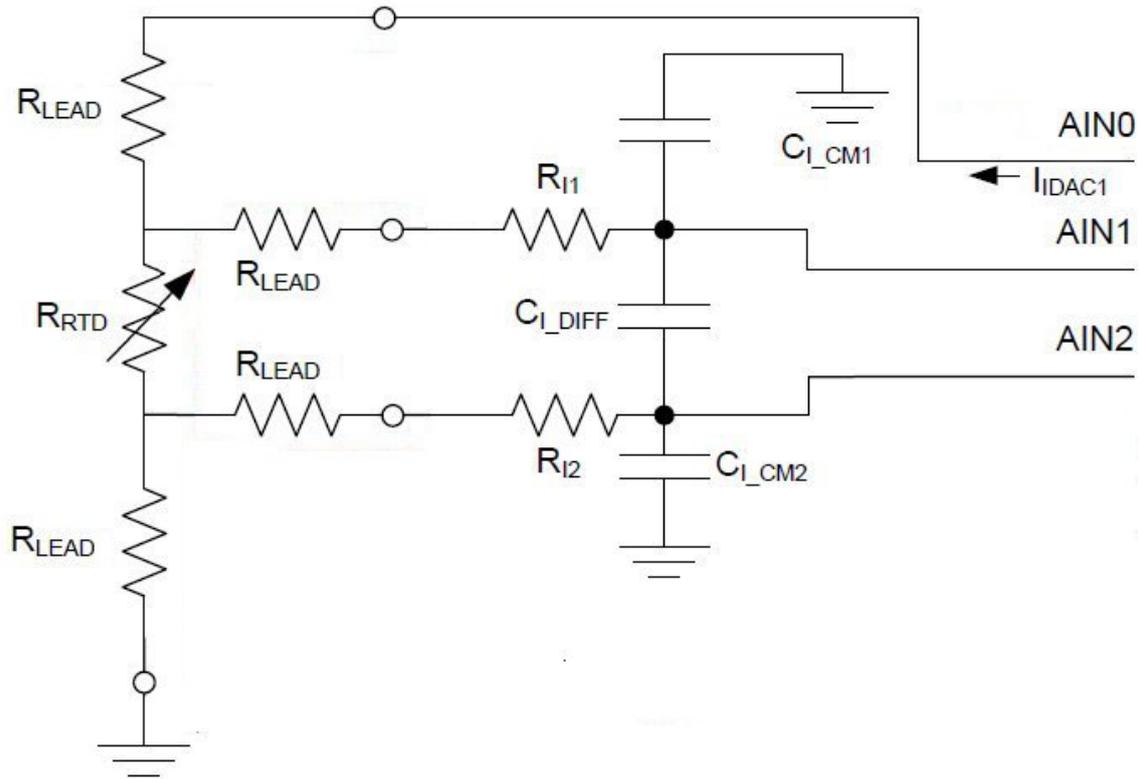


Figura 11: Esquema eléctrico para la conexión de una RTD de 4 hilos

### 2.1.4 Esquema electrónico para la lectura de cualquier tipo de Pt100

Una vez definido el esquema necesario para cada tipo de conexión ha llegado el momento de combinarlos todos en uno único el cual gracias a la presencia de jumpers permita de forma fácil y alternándolos. El esquema resultante se puede observar en la Figura 12, en el cual se incorpora adicionalmente una resistencia de compensación ( $R_{COMP}$ , en el esquemático representada como la unión de  $R_2$  y  $R_3$ ) para el caso de 3 hilos como forma adicional de medir el voltaje de este tipo de conexión. A su vez también se incorpora la posibilidad de que el ADS1247 emplee un voltaje de referencia interno ( $V_{REF}$ ) para la lectura del voltaje en el ADC o emplear un voltaje externo generado en una resistencia de referencia ( $R_{REF}$ , en el esquemático  $R_1$ ). Por tanto, finalmente el sistema podrá emplearse con 7 configuraciones distintas: 2 hilos  $V_{REF}$ , 3 hilos  $V_{REF}$ , 3 hilos  $R_{REF}$ , 3 hilos  $V_{REF}$   $R_{COMP}$ , 3 hilos  $R_{REF}$   $R_{COMP}$ , 4 hilos  $V_{REF}$  y 4 hilos  $R_{REF}$ .

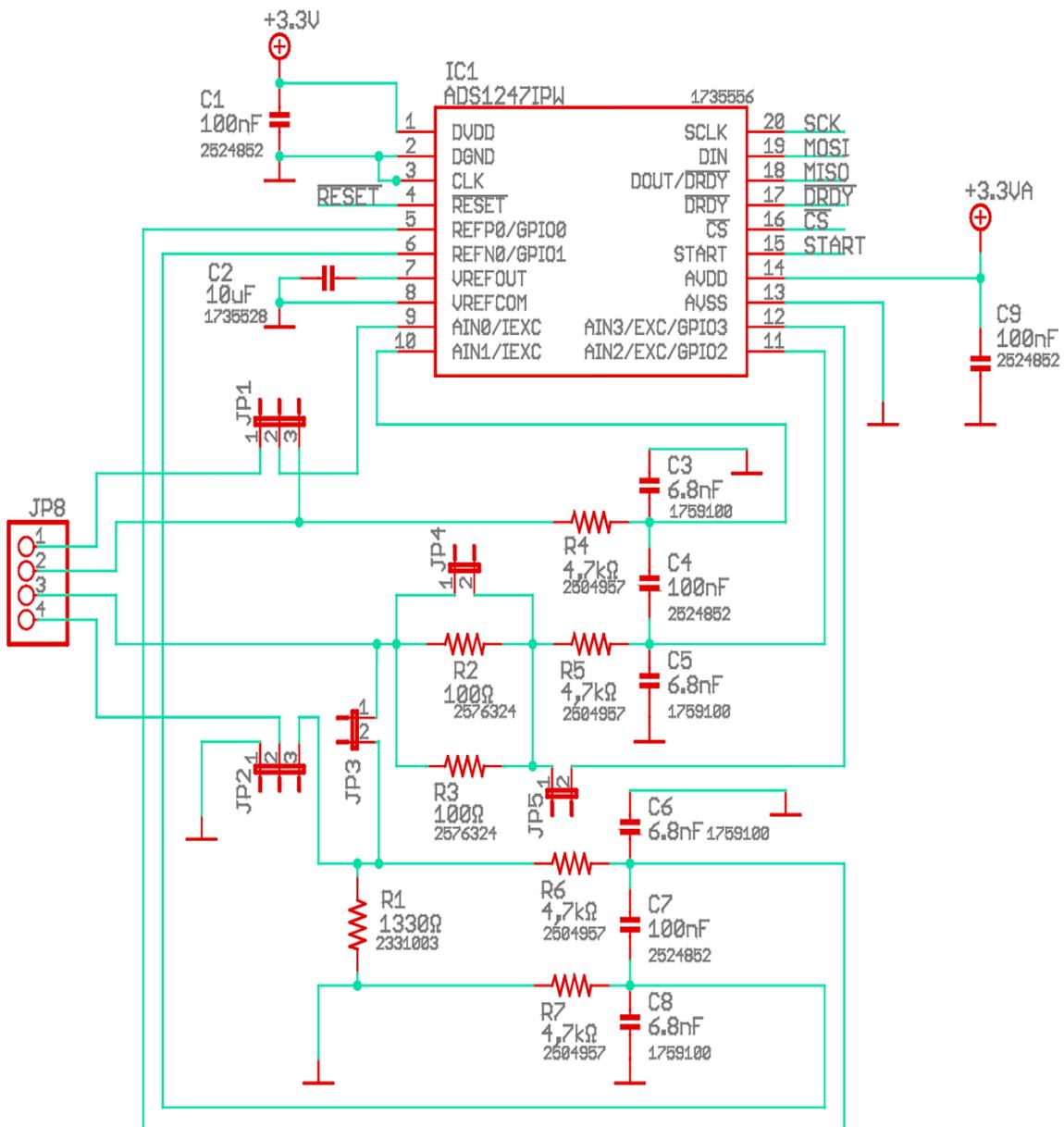


Figura 12: Esquema electrónico del analog front-end compatible con todo tipo de Pt100 (extracto del plano 1)

En el esquemático se puede ver la alimentación diferenciada de la parte analógico y digital del ADS1247, ambas de 3.3V y ambas conectadas a tierra a través de un condensador de 100 nF siguiendo las recomendaciones del fabricante en el datasheet del ADS1247 [7] para filtrar parte del ruido eléctrico que pueda venir por la alimentación. En la línea de lo anterior, el fabricante establece que los pines DGND,CLK,VREFCOM y AVSS van conectados directamente a tierra y el pin VREFOUT a un condensador que también va conectado a tierra. En el conector JP8 es donde irán conectados los cables de la Pt100 siguiendo el orden de ocupación de los del medio para la de 2 hilos, el 4 para la de 3 hilos y todos para la de 4 hilos.

La posición de los jumpers correspondientes a cada configuración tanto de distintos hilos de Pt100 como de forma de medir el voltaje son los siguientes:

Tabla 1: Posición de los jumpers para cada configuración

Configuración	JP1	JP2	JP3	JP4	JP5
2 hilos $V_{REF}$	23	23	Conectado	Conectado	No conectado
3 hilos $V_{REF}$	23	23	No conectado	Conectado	Conectado
3 hilos $R_{REF}$	23	23	No conectado	Conectado	Conectado
3 hilos $V_{REF} R_{COMP}$	23	23	No conectado	No conectado	Conectado
3 hilos $R_{REF} R_{COMP}$	23	23	No conectado	No conectado	Conectado
4 hilos $V_{REF}$	12	23	No conectado	Conectado	No conectado
4 hilos $R_{REF}$	12	23	No conectado	Conectado	No conectado

Aunque en principio se pretendía usar en algún caso la posición 12 del segundo jumper, por requerimientos de voltaje mínimo en la RTD para ser medida por el ADS1247 no se ha podido utilizar. Estos requerimientos serán fundamentales a la hora de seleccionar los componentes exactos utilizados en el esquema de la Figura 12.

### 2.1.5 Elección de los componentes siguiendo los requisitos del ADS1247

En primer lugar, es importante establecer el rango de operación en el que puede trabajar una Pt100 en cuanto a su resistencia interna. Según el documento Europeo EN 60751:2008 [6], el cual trata de los sensores térmicos basados en resistencias de platino, una Pt100 a  $-200^{\circ}\text{C}$  debe de tener  $18.52 \Omega$  y  $390.48 \Omega$  a  $850^{\circ}\text{C}$ . En segundo lugar, se establece que la intensidad de las corrientes eléctricas generadas por el ADS1247 (I<sub>IDAC</sub>) será de 0.5 mA en todos los casos, con lo que se maximizará el voltaje de la RTD manteniendo el autocalentamiento lo más bajo posible. Para que la corriente se pueda generar es necesario que el voltaje con el que se alimenta el ADS1247 de 3.3V sea inferior al voltaje máximo de la RTD y el de la resistencia de

referencia combinado. Se prueba con una resistencia de referencia de 1330  $\Omega$  empleando la ecuación (1). Para el caso más desfavorable, que sería con las dos corrientes de la conexión de 3 hilos, comprobándose el resultado es 1.53V y por tanto inferior a 3.3V.

$$V_{IDAC\ MAX} = R_{RTD\ MAX} \cdot I_{IDAC} + R_{REF} \cdot (2 \cdot I_{IDAC}) \quad (1)$$

A continuación, se procede a configurar el amplificador de ganancia programable. La ganancia máxima que puede tener viene determinada por el voltaje de referencia que se opte y por el voltaje máximo posible en la RTD, siguiendo la ecuación (2).

$$Ganancia_{MAX} = V_{REF} / V_{RTD\ MAX} \quad (2)$$

En la Tabla 2 se puede observar este cálculo para cada caso.

Tabla 2: Resultados para el cálculo de la ganancia máxima del amplificador en cada configuración

Configuración	$V_{REF}$	Ganancia máxima
2 hilos $V_{REF}$	2,048 V	10,49
3 hilos $V_{REF}$	2,048 V	10,49
3 hilos $R_{REF}$	1,33 V	6,81
3 hilos $V_{REF}$ $R_{COMP}$	2,048 V	10,49
3 hilos $R_{REF}$ $R_{COMP}$	1,33 V	6,81
4 hilos $V_{REF}$	2,048 V	10,49
4 hilos $R_{REF}$	0,665 V	3,4

Adicionalmente, es necesario comprobar que el voltaje del nodo común en la RTD está dentro de las especificaciones técnicas y es posible su medida correcta. Este voltaje se calcula con la ecuación (3) y se comprueba con la ecuación (4), tanto para el voltaje en RTD máximo como para el mínimo.

$$V_{CM} = (V_{RTD\ MAX} / 2) + V_{Rref} \quad (3)$$

$$AVSS + 0.1 + (V_{RTD\ MAX} \cdot Ganancia) / 2 \leq V_{CM} \leq AVDD - 0.1 - (V_{RTD\ MAX} \cdot Ganancia) / 2 \quad (4)$$

En el supuesto de que el voltaje en RTD es máximo se obtienen los resultados de la Tabla 3 y para el supuesto del voltaje con la resistencia RTD es mínima son obtenidos los datos de la Tabla 4. Cabe destacar que en el caso de 2 y 4 hilos empleando  $V_{REF}$  es en el único caso que no se ha podido cumplir la ecuación empleando la Ganancia disponible inmediatamente inferior, sino la siguiente. Esto es debido a que en el supuesto de RTD a 850°C (resistencia máxima), el voltaje del nodo común era menor al límite inferior (0,88 V > 0,76 V).

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

Tabla 3: Resultados de la comprobación del voltaje en el nodo común de la RTD a 850°C

Configuración	Límite inferior	V <sub>cm</sub>	Límite superior	Ganancia
2 hilos V <sub>REF</sub>	0,49 V	0,76 V	2,8 V	4
3 hilos V <sub>REF</sub>	0,88 V	1,43 V	2,42 V	8
3 hilos R <sub>REF</sub>	0,49 V	1,43 V	2,8 V	4
3 hilos V <sub>REF</sub> R <sub>COMP</sub>	0,88 V	1,43 V	2,42 V	8
3 hilos R <sub>REF</sub> R <sub>COMP</sub>	0,49 V	1,43 V	2,8 V	4
4 hilos V <sub>REF</sub>	0,49 V	1,43 V	2,8 V	4
4 hilos R <sub>REF</sub>	0,3 V	0,76 V	3 V	2

Tabla 4: Resultados de la comprobación del voltaje en el nodo común de la RTD a -200°C

Configuración	Límite inferior	V <sub>cm</sub>	Límite superior	Ganancia
2 hilos V <sub>REF</sub>	0,12 V	0,67 V	3,18 V	4
3 hilos V <sub>REF</sub>	0,14 V	1,33 V	3,16 V	8
3 hilos R <sub>REF</sub>	0,12 V	1,33 V	3,18 V	4
3 hilos V <sub>REF</sub> R <sub>COMP</sub>	0,14 V	1,33 V	3,16 V	8
3 hilos R <sub>REF</sub> R <sub>COMP</sub>	0,12 V	1,33 V	3,18 V	4
4 hilos V <sub>REF</sub>	0,12 V	0,67 V	3,18 V	4
4 hilos R <sub>REF</sub>	0,14 V	1,33 V	3,16 V	2

Por último, es necesario seleccionar los condensadores y resistencias que forman los filtros de nodo común y diferenciales tanto de la entrada del voltaje de la RTD al ADS1247 como del voltaje de la R<sub>REF</sub>. Las resistencias escogidas son unas de 4700 Ω. La medida del voltaje puede verse influida por las diferencias de tamaño entre estas resistencias por lo que no se recomienda escogerlas de mayor tamaño y se buscará una baja tolerancia para las mismas.

Los filtros diferenciales se diseñan para que tengan un margen respecto al ancho de banda del ADC, concretamente el punto donde filtran -3dB deberá de ser 10 veces más grande. Pese a que se va a usar inicialmente un muestreo de 5 muestras por segundo, para hacerlo compatible hasta 20 muestras por segundo (ancho de banda de 14,8 Hz) la frecuencia de corte se posiciona en 150 Hz. Aplicando la ecuación (5) sobre el filtro situado a la entrada del ADC (formado por C<sub>4</sub>, R<sub>4</sub> y R<sub>5</sub>) se obtiene como condensador para el filtro diferencial de 111 nF. El valor normalizado más cercano de condensador es de 100 nF con lo que se tiene una frecuencia de corte de 166 Hz suficiente para la aplicación. El filtro situado en la medida de voltaje de la resistencia de referencia (R<sub>1</sub>) se comporta de la misma manera.

$$f_{-3dB\_DIFF} = 1 / (2 \cdot \pi \cdot C_4 \cdot (R_4 + R_{RTD} + R_5)) \quad (5)$$

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

Siguiendo la recomendación del fabricante plasmada en el datasheet del ADS1247 [7], los condensadores del filtro común serán al menos 10 veces menores para evitar que diferencias entre los condensadores que forman el filtro provoquen diferencias en el voltaje medido. Por lo tanto, los condensadores del nodo común serán de 6.8 nF. Aplicando las ecuaciones (6) y (7) se obtienen unas frecuencias de corte de 3787 Hz y 3881 Hz respectivamente.

$$f_{-3dB\_CM+} = 1 / (2 \cdot \pi \cdot C_5 \cdot (R_4 + R_{RTD} + R_{REF})) \quad (6)$$

$$f_{-3dB\_CM-} = 1 / (2 \cdot \pi \cdot C_5 \cdot (R_4 + R_{REF})) \quad (7)$$

## 2.2 Diseño del esquema electrónico para la recepción del voltaje desde el ADS1247 y envío de la temperatura al ordenador

Para la comunicación se ha optado por el uso del microcontrolador PIC24FJ64GA004, el cual se configura (Figura 13) según su datasheet [8] alimentándolo por tres pines (dos VDD y AVDD) los cuales están conectados a tierra por condensadores de 1µF. También se encuentran conectados a tierra varios pines y especialmente el VDDCORE pasa previamente por un condensador de 10 µF.

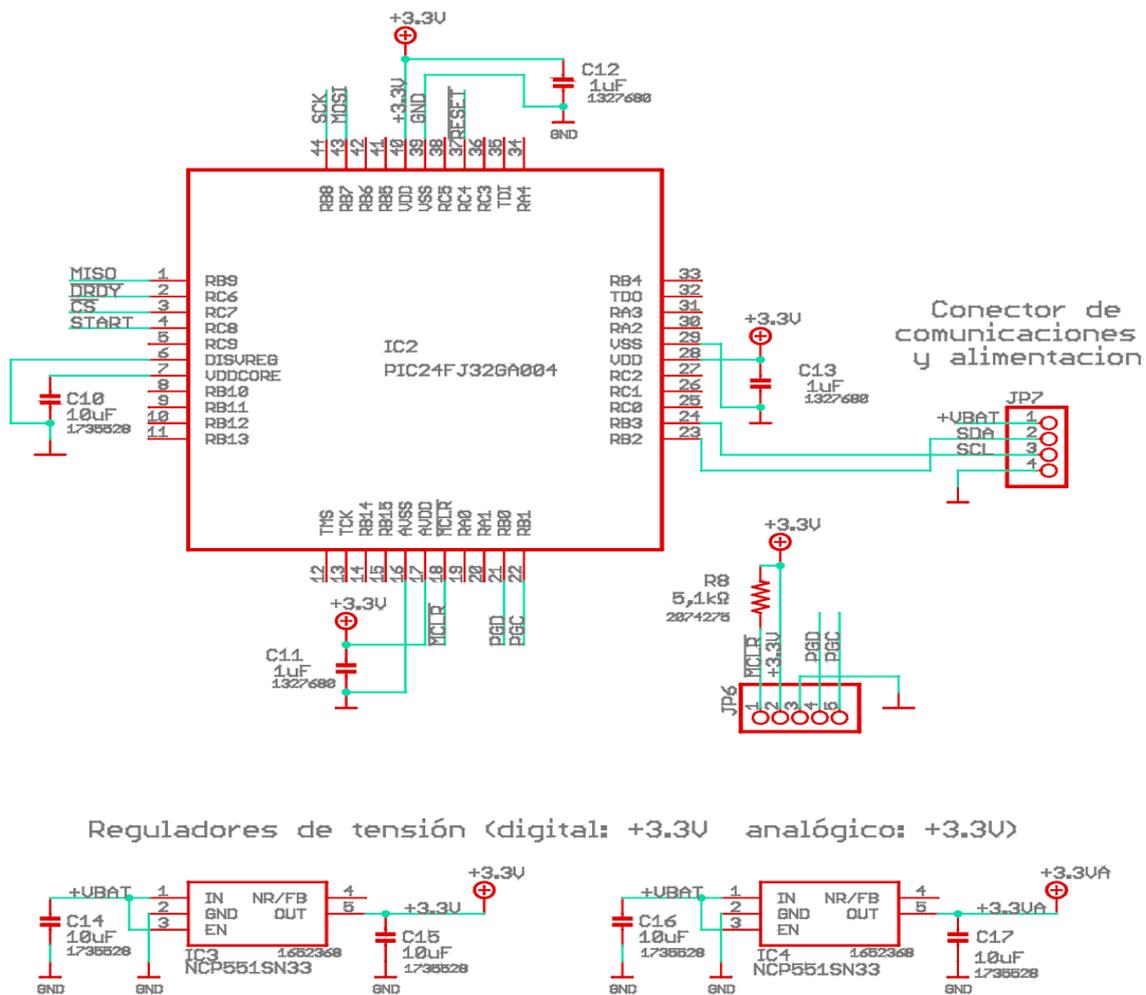


Figura 13: Esquema electrónico de la parte de procesamiento de información y comunicación (extracto del plano 1)

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

Para poder programar el microcontrolador se usa el conector JP6 alimentado también con 3.3V y conectado con 3 conexiones al microcontrolador. Para la comunicación con el ordenador es necesario primero una salida en formato TTL desde el microcontrolador por el conector JP7, el cual sirve también para alimentar todo el sistema. Esta alimentación se adapta a los requisitos con los reguladores de tensión fija NCP551SN33T1G , uno para la parte de alimentación digital y el otro para la parte analógica que solo se presenta en el ADS1247.

Cabe destacar el uso de condensadores de 10  $\mu$ F para mitigar cualquier posible ruido eléctrico y el hecho de que se tiene un sistema de alimentación independiente para la parte analógica, evitando que pueda haber interferencias transmitidas desde la parte digital.

La salida TTL pasa por un convertidor a micro USB el cual se conecta de la siguiente manera:

- +VBAT al conector del adaptador VCC.
- La transmisión del SDA (U1TX) al conector receptor del adaptador RX1.
- La recepción del SCL (U1RX) al conector emisor del adaptador TX0.
- Ambas tomas a tierra unidas.

## 2.3 Listado de componentes utilizados

En la Tabla 5 se encuentran todos los componentes electrónicos necesarios para realizar el diseño expuesto en el apartado anterior. Todos ellos serán comprados en la tienda Farnell.

Tabla 5: Componentes electrónicos necesarios para construir la placa de circuito impreso

Descripción	Cantidad
Texas Instruments ADS1247IPW (Referencia Farnell: 1735556)	1
MICROCHIP PIC24FJ64GA004-I/PT (Referencia Farnell: 1332299)	1
Regulador de tensión de 4,3V-12V a 3,3V NCP551SN33T1G (R. F.: 1652368)	2
Condensador GRM188R60J106ME47D 10 $\mu$ F (Referencia Farnell: 1735528)	6
Condensador 0603YD105KAT2A 1 $\mu$ F (Referencia Farnell: 1327680)	3
Condensador C0603X5R1A104K030BC 100 nF (Referencia Farnell: 2524852)	4
Condensador MC0603B682K500CT 6,8 nF (Referencia Farnell: 1759100)	4
Resistencia MCSR06X5101FTL 5.1 k $\Omega$ , tolerancia 1% (Referencia Farnell:2074275)	1
Resistencia ACPP06034K7BE 4,7 k $\Omega$ , tolerancia 0,1% (Referencia Farnell: 2504957)	4
Resistencia WINT0603LF111000A 100 $\Omega$ , 0.05% (Referencia Farnell: 2576324)	2
Resistencia CPF0603B1K33E1 1.33 k $\Omega$ , 0,1% (Referencia Farnell: 2331003)	1
Jumper M7581-05 con punto de prueba (Referencia Farnell: 2396303)	4

## **CAPÍTULO 3. DISEÑO DEL CIRCUITO IMPRESO**

Al trasladar el esquemático electrónico al diseño de la placa de circuito impreso se ha empleado un diseño de dos capas en el que la capa inferior es un plano de masa con algunas conexiones que no se podían realizar en la capa superior. El plano de masa simplifica el diseño al ser necesarias menos conexiones individuales y es lo suficientemente grande como para que las distintas conexiones a tierra se vean afectadas unas con las otras. Las interfaces con el exterior se han situado en 3 extremos diferentes de la placa:

- En el margen izquierdo se ha situado la conexión con la Pt100.
- En el margen inferior está la conexión con el programador del microcontrolador PIC.
- En el margen derecho se sitúa la conexión con el convertidor de TTL a micro USB.

En el diseño electrónico se ha decidido no colocar resistencias entre las conexiones del ADS1247 y el microcontrolador debido a la esperada cercanía que iba a existir entre ellos, eliminando la posibilidad de interferencias eléctricas relevantes. Por lo tanto, estos dos elementos se colocarán lo más cerca posible, resultando en un diseño que parte desde la izquierda con la conexión con la Pt100 y el ADS1247 con todos los jumpers necesarios. A continuación, en la parte superior se sitúa el microcontrolador junto a la conexión de programación y a la de transmisión de datos al ordenador. Finalmente, en la parte inferior centrado en mitad de la placa se encuentran los dos reguladores de tensión.

La situación de los componentes definitiva se puede observar en el plano 2 y las conexiones entre ellos en los planos 3 y 4, cada uno destinado a una capa del diseño. El grosor del PCB es de 1,6 mm y la anchura de los cables que realizan las conexiones es de 0,4064 mm. El tamaño definitivo de la placa es de 36x24 mm y una altura de 8mm debido a la altura de los jumpers.

Para la fabricación de la placa se ha optado por la empresa china ITEAD la cual pese a tener plazos de entrega más largos que una opción española (como por ejemplo Lab circuits) el precio es mucho menor siempre que los requisitos de la placa no sean muy exigentes y sea de pequeño tamaño, como es el caso. El plazo de entrega se estima de 3 semanas, siendo crucial acabar el diseño del PCB pronto para mientras llega el pedido hacer otras partes del proyecto como la programación.

## **CAPÍTULO 4. PROGRAMACIÓN DEL SOFTWARE**

Para configurar el dispositivo y que responda como se espera es necesario realizar una serie de programación sobre el microcontrolador, existiendo la necesidad de adaptar una parte programa base a cada tipo de conexión que se va a soportar. Solo será necesario realizar programas para el microcontrolador ya que es el controla tanto la comunicación con el ordenador como la configuración del ADS1247 y el procesamiento de los datos. Para tener una cierta modularidad se divide el programa en dos archivos. En primer lugar, el principal Pt100.c es donde se configura el microcontrolador, la conversión de voltaje a temperatura, las interrupciones de envío y recepción de datos y la función principal la cual inicializar correctamente el ADS1247. En segundo lugar, un archivo de tipo header llamado ADS1247.h en el que se encuentran las funciones que configuran el ADS1247 usadas en la función principal. El programa es de tipo continuo, por lo que no se detendrá hasta que no se apague el dispositivo e iniciándose con el encendido del mismo. A continuación, se van a explicar las partes en las que se dividen ambos archivos internamente, partiendo como archivo base el destinado para la versión de 4 hilos usando voltaje de referencia el cual se modifica para el resto de las versiones.

### **4.1 Partes del código presente en el archivo Pt100.c**

#### **4.1.1 Configuración del microcontrolador**

En primer lugar, se encuentra el código destinado a configurar el microcontrolador. Este comienza incluyendo el archivo de tipo header específico del microcontrolador usado donde están todas las definiciones y funciones necesarias para poder configurarlo.

| 1. `#include <24FJ64GA004.h>`

Posteriormente se encuentran una serie de directivas de cabecera las cuales se introducen con la directiva `#fuses` y permiten activar o desactivar funciones específicas del microcontrolador. En el siguiente extracto de código se puede ver qué directivas son empleadas en este proyecto en particular:

```
| 2. #fuses NOJTAG // El puerto JTAG está deshabilitado
| 3. #fuses NOPROTECT // La protección de código está deshabilitada
| 4. #fuses NOWRT // Escrituras a la memoria del programa permitidas
| 5. #fuses ICSP1 // Las funciones del emulador son compartidas con PGEC1/PGED1
| 6. #fuses NOWDT // Watch Dog Timer está deshabilitado; SWDTEN bit está deshabilitado
| 7. #fuses WDT128 // WDT Prescaler Ratio de 1:128
```

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

```
8. #fuses WPOSTS16 // Watchdog Timer Postscaler Select bits-> 1:32,768 (No usado)
9. #fuses NOIESO // Modo IESO está deshabilitado
10. #fuses FRC_PS // RC Oscillator rápido con Postscaler (FRCDIV)
11. #fuses CKSNOFM // Clock switching habilitado, Fail -Safe Clock Monitor deshabilitado
12. #fuses OSCIO // OSCO/CLKO/RC15 funciona como puerto I/O (RC15)
13. #fuses NOPR // Modo Primary Oscillator está deshabilitado
14. // -----
```

También es necesario especificar la función de los pines configurables del microcontrolador encargados por una parte de controlar el ADS1247, empleando para ello la directiva #pin\_select. Los pines empleados realizan las siguientes funciones, todas ellas relacionadas con la comunicación de tipo SPI con el ADS1247:

- MISO: Salida de datos del Esclavo (ADS1247) y entrada al Máster.
- DRDY: Indica que los datos están listos.
- MOSI: Salida de datos del Máster (microcontrolador) y entrada de datos al Esclavo.
- SCLK: Empleado para el envío de los pulsos del reloj necesarios para la sincronización.

Así como los pines encargados de la comunicación de tipo UART entre el microcontrolador y el convertidor de TTL a micro USB:

- RB2: Empleado para la transmisión de datos desde el microcontrolador está unido al conector de la placa TX conectado al puerto del adaptador micro USB RX1.
- RB3: Empleado para la recepción de datos desde el ordenador está unido al conector de la placa RX conectado al puerto del adaptador a micro USB TX1.

Debido al uso también de algunos pines no configurables como CS\_ADS (selección de chip), STRT (inicio) o AD\_RESET (configuración por defecto), también es posible encontrar definiciones (empleando la directiva #define) las cuales al sustituir el número de pin por una palabra representativa facilita su uso en el resto del programa. En este extracto de código se puede observar todo lo anteriormente mencionado:

```
15. #use delay(clock=4000000)
16. // -----
17. #define ON( pin) output_high(pin)
18. #define OFF(pin) output_low( pin)
19. // -----
20. #pin_select SDI1 = PIN_B9 // MISO (Master Input Slave Output) ( 1)
21. #pin_select INT1 = PIN_C6 // DRDY: External Interrupt 1 ( 2)
22. #define CS_ADS PIN_C7 // ( 3)
23. #define STRT PIN_C8 // ( 4)
24. #pin_select U1TX = PIN_B2 // RB2 TX (23)
25. #pin_select U1RX = PIN_B3 // RB3 RX (24)
26. #define AD_RESET PIN_C4 // (37)
27. #pin_select SDO1 = PIN_B7 // MOSI (Master Output Slave Input) (43)
28. #pin_select SCK1OUT= PIN_B8 // SCLK (Clock) (44)
```

Posteriormente, se configura la conexión UART con la que se enviarán los datos al ordenador, empleando una tasa de envío de 9600 baudios, como se puede observar en el siguiente extracto de código:

```
30. #byte U1STA = 0x0222
31. #byte U1RXREG = 0x0226
32. #use rs232(baud=9600,UART1,stream=TTL) /* Envíos a 9600 baudios en el UART1
    (Universal Asynchronous Receiver-Transmitter) */
33. // -----
34. #byte SPI1BUF = 0x0248
35. #include "ADS1247.h"
```

#### 4.1.2 Procesamiento de los datos provenientes del ADS1247

Para realizar la conversión de voltaje presente en la Pt100 a temperatura es necesario emplear un vector en la que estén los valores de resistencia que representan cada una de las temperaturas, desde -200°C hasta 850°C. La matriz resultante se define de la siguiente manera, estando formada por 1051 valores procedentes de la Tabla 1 del documento Europeo EN 60751:2008 [6]:

```
37. /*Vector para obtener la Temperatura en función de la Resistencia
38. Tabla de resistencia de RTD (R x 100) desde
    -200°C a 850°C según el documento europeo EN60751:2008*/
39. const unsigned int16 R[1051]={
40. 1852,1895, 1938, 1982, 2025, 2068, 2111, 2154, 2197, 2240, 2283,
41. 2325, 2368, 2411, 2454, 2497, 2539, 2582, 2624, 2667, 2710,
42. 2752, 2795, 2837, 2880, 2922, 2964, 3007, 3049, 3091, 3134,
43. 3176, 3218, 3260, 3302, 3344, 3386, 3428, 3470, 3512, 3554,
44. 3596, 3638, 3680, 3722, 3764, 3805, 3847, 3889, 3931, 3972,
45. 4014, 4056, 4097, 4139, 4180, 4222, 4263, 4305, 4346, 4388,
46. 4429, 4470, 4512, 4553, 4594, 4636, 4677, 4718, 4759, 4800,
47. 4842, 4883, 4924, 4965, 5006, 5047, 5088, 5129, 5170, 5211,
48. 5252, 5293, 5334, 5375, 5415, 5456, 5497, 5538, 5579, 5619,
49. 5660, 5701, 5741, 5782, 5823, 5863, 5904, 5944, 5985, 6026,
50. 6066, 6107, 6147, 6188, 6228, 6268, 6309, 6349, 6390, 6430,
51. 6470, 6511, 6551, 6591, 6631, 6672, 6712, 6752, 6792, 6833,
52. 6873, 6913, 6953, 6993, 7033, 7073, 7113, 7153, 7193, 7233,
53. 7273, 7313, 7353, 7393, 7433, 7473, 7513, 7553, 7593, 7633,
54. 7673, 7712, 7752, 7792, 7832, 7872, 7911, 7951, 7991, 8031,
55. 8070, 8110, 8150, 8189, 8229, 8269, 8308, 8348, 8387, 8427,
56. 8467, 8506, 8546, 8585, 8625, 8664, 8704, 8743, 8783, 8822,
57. 8862, 8901, 8940, 8980, 9019, 9059, 9098, 9137, 9177, 9216,
58. 9255, 9295, 9334, 9373, 9412, 9452, 9491, 9530, 9569, 9609,
59. 9648, 9687, 9726, 9765, 9804, 9844, 9883, 9922, 9961,10000,
60. 10039,10078,10117,10156,10195,10234,10273,10312,10351,10390,
61. 10429,10468,10507,10546,10585,10624,10663,10702,10740,10779,
62. 10818,10857,10896,10935,10973,11012,11051,11090,11129,11167,
63. 11206,11245,11283,11322,11361,11400,11438,11477,11515,11554,
```

64. 11593,11631,11670,11708,11747,11786,11824,11863,11901,11940,  
65. 11978,12017,12055,12094,12132,12171,12209,12247,12286,12324,  
66. 12363,12401,12439,12478,12516,12554,12593,12631,12669,12708,  
67. 12746,12784,12822,12861,12899,12937,12975,13013,13052,13090,  
68. 13128,13166,13204,13242,13280,13318,13357,13395,13433,13471,  
69. 13509,13547,13585,13623,13661,13699,13737,13775,13813,13851,  
70. 13888,13926,13964,14002,14040,14078,14116,14154,14191,14229,  
71. 14267,14305,14343,14380,14418,14456,14494,14531,14569,14607,  
72. 14644,14682,14720,14757,14795,14833,14870,14908,14946,14983,  
73. 15021,15058,15096,15133,15171,15208,15246,15283,15321,15358,  
74. 15396,15433,15471,15508,15546,15583,15620,15658,15695,15733,  
75. 15770,15807,15845,15882,15919,15956,15994,16031,16068,16105,  
76. 16143,16180,16217,16254,16291,16329,16366,16403,16440,16477,  
77. 16514,16551,16589,16626,16663,16700,16737,16774,16811,16848,  
78. 16885,16922,16959,16996,17033,17070,17107,17143,17180,17217,  
79. 17254,17291,17328,17365,17402,17438,17475,17512,17549,17586,  
80. 17622,17659,17696,17733,17769,17806,17843,17879,17916,17953,  
81. 17989,18026,18063,18099,18136,18172,18209,18246,18282,18319,  
82. 18355,18392,18428,18465,18501,18538,18574,18611,18647,18684,  
83. 18720,18756,18793,18829,18866,18902,18938,18975,19011,19047,  
84. 19084,19120,19156,19192,19229,19265,19301,19337,19374,19410,  
85. 19446,19482,19518,19555,19591,19627,19663,19699,19735,19771,  
86. 19807,19843,19879,19915,19951,19987,20023,20059,20095,20131,  
87. 20167,20203,20239,20275,20311,20347,20383,20419,20455,20490,  
88. 20526,20562,20598,20634,20670,20705,20741,20777,20813,20848,  
89. 20884,20920,20956,20991,21027,21063,21098,21134,21170,21205,  
90. 21241,21276,21312,21348,21383,21419,21454,21490,21525,21561,  
91. 21596,21632,21667,21703,21738,21774,21809,21844,21880,21915,  
92. 21951,21986,22021,22057,22092,22127,22163,22198,22233,22268,  
93. 22304,22339,22374,22409,22445,22480,22515,22550,22585,22621,  
94. 22656,22691,22726,22761,22796,22831,22866,22902,22937,22972,  
95. 23007,23042,23077,23112,23147,23182,23217,23252,23287,23321,  
96. 23356,23391,23426,23461,23496,23531,23566,23600,23635,23670,  
97. 23705,23740,23774,23809,23844,23879,23913,23948,23983,24018,  
98. 24052,24087,24122,24156,24191,24226,24260,24295,24329,24364,  
99. 24399,24433,24468,24502,24537,24571,24606,24640,24675,24709,  
100. 24744,24778,24813,24847,24881,24916,24950,24985,25019,25053,  
101. 25088,25122,25156,25191,25225,25259,25293,25328,25362,25396,  
102. 25430,25465,25499,25533,25567,25601,25635,25670,25704,25738,  
103. 25772,25806,25840,25874,25908,25942,25976,26010,26044,26078,  
104. 26112,26146,26180,26214,26248,26282,26316,26350,26384,26418,  
105. 26452,26486,26520,26553,26587,26621,26655,26689,26722,26756,  
106. 26790,26824,26857,26891,26925,26959,26992,27026,27060,27093,  
107. 27127,27161,27194,27228,27261,27295,27329,27362,27396,27429,  
108. 27463,27496,27530,27563,27597,27630,27664,27697,27731,27764,  
109. 27798,27831,27864,27898,27931,27964,27998,28031,28064,28098,  
110. 28131,28164,28198,28231,28264,28297,28331,28364,28397,28430,  
111. 28463,28497,28530,28563,28596,28629,28662,28695,28729,28762,  
112. 28795,28828,28861,28894,28927,28960,28993,29026,29059,29092,  
113. 29125,29158,29191,29224,29256,29289,29322,29355,29388,29421,  
114. 29454,29486,29519,29552,29585,29618,29650,29683,29716,29749,

```
115. 29781,29814,29847,29880,29912,29945,29978,30010,30043,30075,
116. 30108,30141,30173,30206,30238,30271,30303,30336,30369,30401,
117. 30434,30466,30498,30531,30563,30596,30628,30661,30693,30725,
118. 30758,30790,30823,30855,30887,30920,30952,30984,31016,31049,
119. 31081,31113,31145,31178,31210,31242,31274,31306,31339,31371,
120. 31403,31435,31467,31499,31531,31564,31596,31628,31660,31692,
121. 31724,31756,31788,31820,31852,31884,31916,31948,31980,32012,
122. 32043,32075,32107,32139,32171,32203,32235,32267,32298,32330,
123. 32362,32394,32426,32457,32489,32521,32553,32584,32616,32648,
124. 32679,32711,32743,32774,32806,32838,32869,32901,32932,32964,
125. 32996,33027,33059,33090,33122,33153,33185,33216,33248,33279,
126. 33311,33342,33374,33405,33436,33468,33499,33531,33562,33593,
127. 33625,33656,33687,33718,33750,33781,33812,33844,33875,33906,
128. 33937,33969,34000,34031,34062,34093,34124,34156,34187,34218,
129. 34249,34280,34311,34342,34373,34404,34435,34466,34497,34528,
130. 34559,34590,34621,34652,34683,34714,34745,34776,34807,34838,
131. 34869,34899,34930,34961,34992,35023,35054,35084,35115,35146,
132. 35177,35208,35238,35269,35300,35330,35361,35392,35422,35453,
133. 35484,35514,35545,35576,35606,35637,35667,35698,35728,35759,
134. 35790,35820,35851,35881,35912,35942,35972,36003,36033,36064,
135. 36094,36125,36155,36185,36216,36246,36276,36307,36337,36367,
136. 36398,36428,36458,36489,36519,36549,36579,36610,36640,36670,
137. 36700,36730,36760,36791,36821,36851,36881,36911,36941,36971,
138. 37001,37031,37061,37091,37121,37151,37181,37211,37241,37271,
139. 37301,37331,37361,37391,37421,37451,37481,37511,37541,37570,
140. 37600,37630,37660,37690,37719,37749,37779,37809,37839,37868,
141. 37898,37928,37957,37987,38017,38046,38076,38106,38135,38165,
142. 38195,38224,38254,38283,38313,38342,38372,38401,38431,38460,
143. 38490,38519,38549,38578,38608,38637,38667,38696,38725,38755,
144. 38784,38814,38843,38872,38902,38931,38960,38990,39019,39048
145. };
146. // -----
```

Lo siguiente que es necesario programar son las interrupciones empleadas por el microcontrolador. La primera es la que se encarga de recibir los datos provenientes de la conexión UART cuando hay un carácter pendiente de ser leído, la cual se puede ver programada en el siguiente extracto de código:

```
147. unsigned int8 trama[256];
148. unsigned int8 cnt;
149. // -----
150. #int_rda //Interrupción de recepción UART (rda)
151. void isr_rda(void)
152. {
153.     while(bit_test(U1STA,0))
154.     {
155.         trama[cnt] = U1RXREG;
156.         cnt++;
157.     }
158. }
159. // -----
```

La otra interrupción que es necesario definir es la que se activa cuando el ADS1247 toma un valor de voltaje. En la primera parte de esta interrupción se recibe el valor proveniente del convertidor ADC de 24 bits, como muestra el siguiente extracto de código:

```
160. #int_EXT1
161. void EXT1_isr(void)
162. {
163.     unsigned int8 s2, s1, s0;
164.     unsigned int16 i, aprx;
165.     int32 value=0;
166.     float valorV=0.0;
167.     float valorR=0.0;
168.     float valorT=0.0;
169.     float FSR, a, b, c;
170.     output_low(CS_ADS);
171.     delay_us(10);
172.     spi_write(0xFF);
173.     s2 = SPI1BUF;
174.     spi_write(0xFF);
175.     s1 = SPI1BUF;
176.     spi_write(0xFF);
177.     s0 = SPI1BUF;
178.     output_high(CS_ADS);
179.     delay_us(10);
180.     value= make32(s2,s1,s0,0); // Valor recibido
```

Una vez recibido se procede a la conversión de este valor a resistencia de la Pt100. Para ello es necesario en primer lugar obtener el voltaje al que corresponde este valor del ADC del ADS1247, empleando la operación que se puede observar en la línea 182 del programa, en el siguiente extracto de código:

```
181. FSR= 0.512; // FSR= Vref / Ganancia
182. valorV = (float)(value/256) *FSR/ 8388608.0; // Voltios en bornes RTD
```

El valor se almacena en la variable “value” la cual en primer lugar se divide entre 256 debido a que es una variable de 32 bits cuando en realidad el convertidor ADC del ADS1247 solo es de 24 bits, por lo que hay que quitarle los 8 bits vacíos no usados. Posteriormente se divide entre  $2^{23}$  (8388608) ya que solo es empleada la mitad positiva del convertidor ADC de 24 bits ya que el voltaje no puede ser negativo en ningún caso.

Por último, se multiplica por el FSR (el rango de la escala del amplificador de ganancia variable) que para cada configuración es distinto. Se pueden observar los valores correspondientes a cada caso en la Tabla 6, usándose los representados en la del cual solo se coge la parte positiva.

Tabla 6: Rango de la escala del amplificador de ganancia variable del ADS1247 para cada configuración

Configuración	V <sub>REF</sub>	Ganancia	FSR
2 hilos V <sub>REF</sub>	2,048 V	4	± 0.512 V
3 hilos V <sub>REF</sub>	2,048 V	8	± 0.256 V
3 hilos R <sub>REF</sub>	1,33 V	4	± 0.3325 V
3 hilos V <sub>REF</sub> R <sub>COMP</sub>	2,048 V	8	± 0.256 V
3 hilos R <sub>REF</sub> R <sub>COMP</sub>	1,33 V	4	± 0.3325 V
4 hilos V <sub>REF</sub>	2,048 V	4	± 0.512 V
4 hilos R <sub>REF</sub>	0,665 V	2	± 0.3325 V

Una vez obtenido el voltaje es necesario obtener la resistencia de la Pt100. Esto se consigue dividiéndolo entre la intensidad siguiendo la ley de Ohm ( $V = R \cdot I$ ). En todos los casos la intensidad es de 0.5 mA por lo que esta operación es la que se encuentra en la línea 183. Para poder usar el vector necesario para la conversión de esta resistencia a temperatura, es necesario previamente multiplicar el valor de la resistencia por 100 y almacenarlo en la variable “aprx”, como se ve a continuación:

```
183. valorR = valorV / 0.0005; // Resistencia con IDAC1 = 0.5 mA
184. aprx=(unsigned int16) (valorR*100.0);
```

El último paso es con un bucle de tipo “for” recorrer todo el vector (los 1051 valores) hasta encontrar con el valor de resistencia exacto o los dos valores entre los que se encuentra. La posición en el vector determina la temperatura a la que se encuentra si se le resta 200, ya que el vector empieza con la resistencia esperada si la Pt100 está a -200°C. En caso de que no se encuentre el valor exacto en el vector se realiza un promedio como se puede ver aquí:

```
185. for(i=0; i<1050; i++)
186. {
187.     if((R[i]<aprx)&&(aprx<R[i+1]))
188.     {
189.         a=(float)(-200+i);
190.         b=(float)( R[i+1] - R[i]);
191.         c=(float)( aprx - R[i]);
192.         valorT= a + (c/b); //Temperatura a partir de la tabla
193.         fprintf(TTL, "%02f\n\r",ValorT);
194.         i=1051;
195.     }
```

En caso de que sí se encuentre el valor exacto de la resistencia en el vector no será necesario realizar ningún promedio. En ambos casos se devolverá el dato por la pantalla del ordenador usando la función fprintf. A continuación se encuentra el código en caso de que el valor exacto de la resistencia se encuentre en el vector:

```
196. if(R[i]==aprx)
197.     {
198.         a=(float)(-200+i);
199.         valorT= a; //Temperatura a partir de la tabla
200.         fprintf(TTL, "%02f\n\r",ValorT);
201.         i=1051;
202.     }
203. }
204. }
```

### 4.1.3 Función principal

La última parte del código corresponde a la función principal del programa, en la que se inicializa y configura tanto el microcontrolador como el ADS1247. Para la calibración y configuración del ADS1247 se emplean funciones que se programarán en el otro archivo (ADS1247.h). Como se puede ver en este extracto de código, finaliza en un bucle infinito:

```
207. void main()
208. {
209.     output_low( STRT);
210.     output_high(CS_ADS);
211.     output_low( AD_RESET); //Reset del ADS1247
212.     delay_ms(10);
213.     output_high(AD_RESET);
214.     delay_ms(500);
215.     output_high( STRT );
216.     setup_spi ( SPI_MASTER | SPI_SS_DISABLED | SPI_L_TO_H | SPI_XMIT_H_TO_L |
                SPI_CLK_DIV_16); // Configuración del SPI
217.     ADS_cero(); // Preparación del ADS para la calibración de Offset
218.     ADS_Offset_calibration(); // Calibración del Offset
219.     ADS_full(); // Preparación del ADS para la calibración de la ganancia
220.     ADS_Gain_calibration(); // Calibración de la ganancia
221.     ADS_init(); // Configuración del ADS
222.     ADS_rd_register();
223.     ext_int_edge(1,H_TO_L); // Activación de la interrupción con cambio de alto a bajo
224.     clear_interrupt(INT_EXT1); // Limpiar interrupción del ADS
225.     enable_interrupts(INT_EXT1); // Interrupción para el ADS habilitada
226.     output_low( CS_ADS );
227.     delay_us(100);
228.     spi_write(RDATACT); // Modo de lectura continua de datos
229.     output_high(CS_ADS );
230.     delay_us(100);
231.     for(;;) // Superbucle infinito
232.     {
233.     }
234. }
```

## 4.2 Partes del código presente en el archivo ADS1247.h

### 4.2.2 Definiciones para el ADS1247

Al principio del código se sitúa una definición condicional en caso de que ya se haya incorporado al archivo principal el mismo archivo de tipo header previamente para que no se vuelva a incluir. Para ello se usa el siguiente código al principio del archivo y al final:

```
1. #ifndef INC_ADS1247
2.
3. #define INC_ADS1247 1
```

```
157. #endif
```

Una vez realizada esta comprobación se procede a definir un conjunto de direcciones de la memoria pertenecientes a los registros empleados en el ADS1247, con lo que se consigue una mayor facilidad a la hora de entender el código posterior:

```
4. // -----
5. //ADS1247 Registros
6. #define MUX0 0x00 //BCS[1:0] MUX_SP[2:0] MUX_SN[2:0]
7. #define VBIAS 0x01 //VBIAS[7:0]
8. #define MUX1 0x02 //CLKSTAT VREFCON[1:0] REFSELT[1:0] MUXCAL[2:0]
9. #define SYS0 0x03 //0 PGA[2:0] DR[3:0]
10. #define OFC0 0x04 //OFC[7:0]
11. #define OFC1 0x05 //OFC[15:8]
12. #define OFC2 0x06 //OFC[23:16]
13. #define FSC0 0x07 //FSC[7:0]
14. #define FSC1 0x08 //FSC[15:8]
15. #define FSC2 0x09 //FSC[23:16]
16. #define IDAC0 0x0A //ID[3:0] DRDY_MODE IMAG[2:0]
17. #define IDAC1 0x0B //I1DIR[3:0] I2DIR[3:0]
18. #define GPIOCFG 0x0C //IOCFG[7:0]
19. #define GPIODIR 0x0D //IODIR[7:0]
20. #define GPIODAT 0x0E //IODAT[7:0]
```

También se define una serie de comandos, por la misma razón que las definiciones anteriores:

```
21. // -----
22. //SPI ADS1247 Comandos
23. #define WAKEUP 0x00 //Salir del modo bajo consumo
24. #define SLP 0x02 //Entrar modo bajo consumo
25. #define SYNC 0x04 //Sincronizar conversiones del ADC
26. #define RESET 0x06 //Volver a los valores por defecto
27. #define NOP 0xFF //No operación
28. #define RDATA 0x12 //Leer datos una vez
29. #define RDATACT 0x14 //Modo de lectura continua de datos
```

```

30. #define SDATACT 0x16 //Parar de leer datos continuamente
31. #define RREG0 0x20 //Leer del registro rrrr 0010 rrrr
32. #define WREG0 0x40 //Escribir en el registro rrrr 0100 rrrr
33. #define SYSOCAL 0x60 //Calibración del Offset
34. #define SYSGCAL 0x61 //Calibración de la ganancia
35. #define SELFOCAL 0x62 //Calibración del Offset propia
    
```

#### 4.2.2 Función para la configuración del ADS1247 variable según el modo de funcionamiento

Después de las definiciones iniciales en este documento se definen 6 funciones: 5 comunes para todos los modos de funcionamiento del dispositivo y una que varía según el tipo de conexión y medición que se emplee. Esta función variable se denomina ADS\_init y se encarga de escribiendo en los registros correspondientes configurar las corrientes generadas, el amplificador de ganancia variable, la muestras por segundo y la referencia usada (sea la del voltaje interno o la resistencia de referencia). Los registros del ADS1247 editados por esta función son los siguientes:

- MUX0 – Registro de control del multiplexador 0. Dirección 00h.

Tabla 7: Bits del registro de control del multiplexador 0

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BCS [1]	BCS [0]	MUX_SP [2]	MUX_SP [1]	MUX_SP [0]	MUX_SN [2]	MUX_SN [1]	MUX_SN [0]

Los bits 7 y 6 controlan la intensidad de las corrientes “Burn-out” ofreciendo estas opciones:

- 00: 0  $\mu$ A
- 01: 0,5  $\mu$ A
- 10: 2  $\mu$ A
- 11: 10  $\mu$ A

Los bits 5, 4 y 3 controlan la entrada positiva del ADC entre las siguientes opciones:

- 000: AIN0 (por defecto)
- 001: AIN1
- 010: AIN2
- 011: AIN3

Los bits 5, 4 y 3 controlan la entrada positiva del ADC entre las siguientes opciones:

- 000: AIN0
- 001: AIN1 (por defecto)
- 010: AIN2
- 011: AIN3

- VBIAS – Registro de control de la tensión de polarización. Dirección 01h.

Tabla 8: Bits del registro de control de la tensión de polarización

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	-	-	-	VBIAS[3]	VBIAS[2]	VBIAS[1]	VBIAS[0]

Cada bit # empleado corresponde a la existencia de tensión de polarización en el pin AIN#:

- 0: No habilitado (por defecto)
- 1: Habilitado

- MUX1 – Registro de control del multiplexador 1. Dirección 02h.

Tabla 9: Bits del registro de control del multiplexador 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CLKSTAT	VREFCON [1]	VREFCON [0]	REFSELT [1]	REFSELT [0]	MUXCAL [2]	MUXCAL [1]	MUXCAL [0]

El bit 7 es solo de lectura e indica si se está empleando un reloj interno o externo:

- 0: Reloj interno
- 1: Reloj externo

Los bits 6 y 5 controlan cuando está encendido el voltaje interno de referencia:

- 00: Siempre apagado (por defecto)
- 01: Siempre encendido
- 10 o 11: Encendido durante las conversiones

Los bits 4 y 5 controlan la referencia usada por el ADC:

- 00: Referencia externa de los pines de entrada REFPO Y REFNO (por defecto)
- 01: --
- 10: Referencia interna
- 11: Referencia interna y conectada a los pines de entrada REFPO Y REFNO

Los bits 2, 1 y 0 permiten seleccionar distintos modos de funcionamiento:

- 000: Operación normal (por defecto)
- 001: Calibración de offset
- 010: Calibración de ganancia
- 011: Modo medición de temperatura
- 100: --
- 101: Monitorización de REF0
- 110: Monitorización de alimentación analógica
- 111: Monitorización de alimentación digital

- SYS0 – Registro de control del sistema 0. Dirección 03h.

Tabla 10: Bits del registro de control del sistema 0

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
-	PGA[2]	PGA[1]	PGA[0]	DR[3]	DR[2]	DR[1]	DR[0]

Los bits 6, 5 y 4 determinan la ganancia del amplificador programable (PGA):

- 000: PGA = 1 (por defecto)
- 001: PGA = 2
- 010: PGA = 4
- 011: PGA = 8
- 100: PGA = 16
- 101: PGA = 32
- 110: PGA = 64
- 111: PGA = 128

Los bits 3, 2, 1 y 0 controlan el ratio de muestras tomadas por el ADC por segundo (SPS):

- 0000: 5 SPS (por defecto)
- 0001: 10 SPS
- 0010: 20 SPS
- 0011: 40 SPS
- 0100: 80 SPS
- 0101: 160 SPS
- 0110: 320 SPS
- 0111: 640 SPS
- 1000: 1000 SPS
- 1001 HASTA 1111: 2000 SPS

- IDAC0 – Registro de control de IDAC 0. Dirección 0Ah.

Tabla 11: Bits del registro de control de IDAC 0

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID[3]	ID[2]	ID[1]	ID[0]	DRDY MODE	IMAG[2]	IMAG[1]	IMAG[0]

Los bits 7, 6, 5 y 4 son de lectura únicamente programados de fábrica y usados para revisar la identificación.

El bit 3 permite usar los pines DOUT para que también cumplan la función del pin DRDY:

- 0: Deshabilitado (por defecto)
- 1: Habilitado

Los bits 2, 1 y 0 permiten elegir la intensidad de las corrientes generadas por el ADS entre las siguientes opciones:

- 000: Apagadas (por defecto)
  - 001: 50  $\mu$ A
  - 010: 100  $\mu$ A
  - 011: 250  $\mu$ A
  - 100: 500  $\mu$ A
  - 101: 750  $\mu$ A
  - 110: 1000  $\mu$ A
  - 111: 1500  $\mu$ A
- IDAC – Registro de control de IDAC 1. Dirección 0Bh.

Tabla 12: Bits del registro de control de IDAC 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
I1DIR[3]	I1DIR[2]	I1DIR[1]	I1DIR[0]	I2DIR[3]	I2DIR[2]	I2DIR[1]	I2DIR[0]

Los bits 7, 6, 5 y 4 determinan el pin de salida de la primera corriente generada:

- 0000: AIN0
- 0001: AIN1
- 0010: AIN2
- 0011: AIN3
- 01xx: --
- 10xx: --
- 11xx: Desconectado (por defecto)

Los bits 3, 2, 1 y 0 determinan el pin de salida de la segunda corriente generada:

- 0000: AIN0
- 0001: AIN1
- 0010: AIN2
- 0011: AIN3
- 01xx y 10xx: --
- 11xx: Desconectado (por defecto)

Para escribir en estos registros es necesario usar el pin de selección de chip desde el microcontrolador, parar momentáneamente la lectura continua de datos y emplear la función `spi_write`. La función para cada una de las configuraciones en las que se puede usar el dispositivo son las siguientes:

- Función ADS\_init para la configuración 4 Y 2 hilos  $V_{REF}$

```
38. void ADS_init(void)
39. {
40.  output_low( CS_ADS);
41.  spi_write(SDATACT); // Parar de leer datos continuamente
42.  spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
43.  spi_write(3);
44.  spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
    / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
45.  spi_write(0b00000000); // Sin voltaje de polarización
46.  spi_write(0b01110000); // Oscilador interno / La referencia interna está activada con
    una conversión está en progreso / Usando referencia interna Vref / Operación normal
47.  spi_write(0b00100000); // PGA 4 / 5 SPS
48.  spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
49.  spi_write(1);
50.  spi_write(0b00000100); // Corrientes de 500ma
51.  spi_write(0b00001111); // IDAC 1= AIN0 IDAC 2= ninguno
52.  delay_us(100);
53.  output_high(CS_ADS);
54.  delay_ms(2);
55. }
56.
57. // -----
58.
```

- Función ADS\_init para la configuración 4 hilos  $R_{REF}$

```
38. void ADS_init(void)
39. {
40.  output_low( CS_ADS);
41.  spi_write(SDATACT); // Parar de leer datos continuamente
42.  spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
43.  spi_write(3);
44.  spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
    / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
45.  spi_write(0b00000000); // Sin voltaje de polarización
46.  spi_write(0b01100000); // Oscilador interno / La referencia interna está activada con
    una conversión está en progreso / Usando referencia externa Rref / Operación normal
47.  spi_write(0b00010000); // PGA 2 / 5 SPS
48.  spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
49.  spi_write(1);
50.  spi_write(0b00000100); // Corrientes de 500ma
51.  spi_write(0b00001111); // IDAC 1= AIN0 IDAC 2= ninguno
52.  delay_us(100);
53.  output_high(CS_ADS);
54.  delay_ms(2);
55. }
```

- Función ADS\_init para la configuración 3 hilos V<sub>REF</sub> (con y sin R<sub>COMP</sub>)

```
38. void ADS_init(void)
39. {
40.   output_low( CS_ADS);
41.   spi_write(SDATACT); // Parar de leer datos continuamente
42.   spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
43.   spi_write(3);
44.   spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
   / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
45.   spi_write(0b00000000); // Sin voltaje de polarización
46.   spi_write(0b01110000); // Oscilador interno / La referencia interna está activada con
   una conversión está en progreso / Usando referencia interna Vref / Operación normal
47.   spi_write(0b00110000); // PGA 8 / 5 SPS
48.   spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
49.   spi_write(1);
50.   spi_write(0b00000100); // Corrientes de 500ma
51.   spi_write(0b00000011); // IDAC 1= AIN0 IDAC 2= AIN3
52.   delay_us(100);
53.   output_high(CS_ADS);
54.   delay_ms(2);
55. }
```

- Función ADS\_init para la configuración 3 hilos R<sub>REF</sub> (con y sin R<sub>COMP</sub>)

```
38. void ADS_init(void)
39. {
40.   output_low( CS_ADS);
41.   spi_write(SDATACT); // Parar de leer datos continuamente
42.   spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
43.   spi_write(3);
44.   spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
   / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
45.   spi_write(0b00000000); // Sin voltaje de polarización
46.   spi_write(0b01100000); // Oscilador interno / La referencia interna está activada con
   una conversión está en progreso / Usando referencia interna Rref / Operación normal
47.   spi_write(0b00100000); // PGA 4 / 5 SPS
48.   spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
49.   spi_write(1);
50.   spi_write(0b00000100); // Corrientes de 500ma
51.   spi_write(0b00000011); // IDAC 1= AIN0 IDAC 2= AIN3
52.   delay_us(100);
53.   output_high(CS_ADS);
54.   delay_ms(2);
55. }
```

#### 4.2.2 Funciones para la configuración del ADS1247 no dependientes del modo de funcionamiento

En primer lugar, la función ADS\_cero prepara el ADS1247 para que el microcontrolador ejecute una calibración del offset para mejorar los valores medidos:

```
59. void ADS_cero(void)
60. {
61.  output_low( CS_ADS);
62.  spi_write(SDATACT); // Parar de leer datos continuamente
63.  spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
64.  spi_write(2);
65.  spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación /
        ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
66.  spi_write(0b00000000); // Sin voltaje de polarización
67.  spi_write(0b00110001); // Oscilador interno / La referencia interna está activa /
        Usando referencia interna Vref / Calibración del Offset
68.  delay_us(100);
69.  output_high(CS_ADS);
70.  delay_ms(2);
71. }
72.
73. // -----
```

En segundo lugar, la función ADS\_full prepara el ADS1247 para que el microcontrolador ejecute una calibración de la ganancia que como en el caso anterior ayuda a la reducción de errores en la medida:

```
75. void ADS_full(void)
76. {
77.  output_low( CS_ADS);
78.  spi_write(SDATACT); // Parar de leer datos continuamente
79.  spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
80.  spi_write(2);
81.  spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación /
        ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
82.  spi_write(0b00000000); // Sin voltaje de polarización
83.  spi_write(0b00110010); // Oscilador interno / La referencia interna está activa /
        Usando referencia interna Vref / Calibración de la ganancia
84.  delay_us(100);
85.  output_high(CS_ADS);
86.  delay_ms(2);
87. }
88.
89. // -----
```

En tercer lugar, la función ADS\_rd\_register permite leer los primeros 15 registros (empezando desde la dirección 00h) del ADS1247:

```
91. void ADS_rd_register(void)
92. {
93.   unsigned int8 B0,B1,B2;
94.   output_low( CS_ADS);
95.   spi_write(SDATACT);
96.   spi_write(RREG0);
97.   spi_write(14);
98.   spi_write(0xFF);
99.   B0=SPI1BUF;
100.  spi_write(0xFF);
101.  B0=SPI1BUF;
102.  spi_write(0xFF);
103.  B0=SPI1BUF;
104.  spi_write(0xFF);
105.  B0=SPI1BUF;
106.  spi_write(0xFF);
107.  B0=SPI1BUF;
108.  spi_write(0xFF);
109.  B1=SPI1BUF;
110.  spi_write(0xFF);
111.  B2=SPI1BUF;
112.  spi_write(0xFF);
113.  B0=SPI1BUF;
114.  spi_write(0xFF);
115.  B1=SPI1BUF;
116.  spi_write(0xFF);
117.  B2=SPI1BUF;
118.  spi_write(0xFF);
119.  B0=SPI1BUF;
120.  spi_write(0xFF);
121.  B0=SPI1BUF;
122.  spi_write(0xFF);
123.  B0=SPI1BUF;
124.  spi_write(0xFF);
125.  B0=SPI1BUF;
126.  spi_write(0xFF);
127.  B0=SPI1BUF;
128.  delay_us(10);
129.  output_high(CS_ADS);
130.  delay_ms(2);
131. }
132. // -----
```

Esta función puede ser empleada para, usando la función fprintf, devolver por pantalla el valor almacenado en estos registros.

Por último, están las funciones `ADS_Offset_calibration` y `ADS_Gain_calibration` las cuales ejecutan los pasos necesarios desde el microcontrolador para respectivamente realizar una calibración del offset y una calibración de la ganancia.

```
133. void ADS_Offset_calibration(void)
134. {
135.   output_low( CS_ADS);
136.   spi_write(SDATACT);
137.   delay_us(10);
138.   spi_write(SYSOCAL);
139.   delay_us(10);
140.   delay_ms(3500);
141.   output_high(CS_ADS);
142.   delay_ms(2);
143. }
144. // -----
145. void ADS_Gain_calibration(void)
146. {
147.   output_low( CS_ADS);
148.   spi_write(SDATACT);
149.   delay_us(10);
150.   spi_write(SYSGCAL);
151.   delay_us(10);
152.   delay_ms(3500);
153.   output_high(CS_ADS);
154.   delay_ms(2);
155. }
```

Este es el último fragmento de código del archivo, el cual finalmente concluye con la directiva `#endif` que se había comentado al principio.

### 4.3 Entorno de trabajo

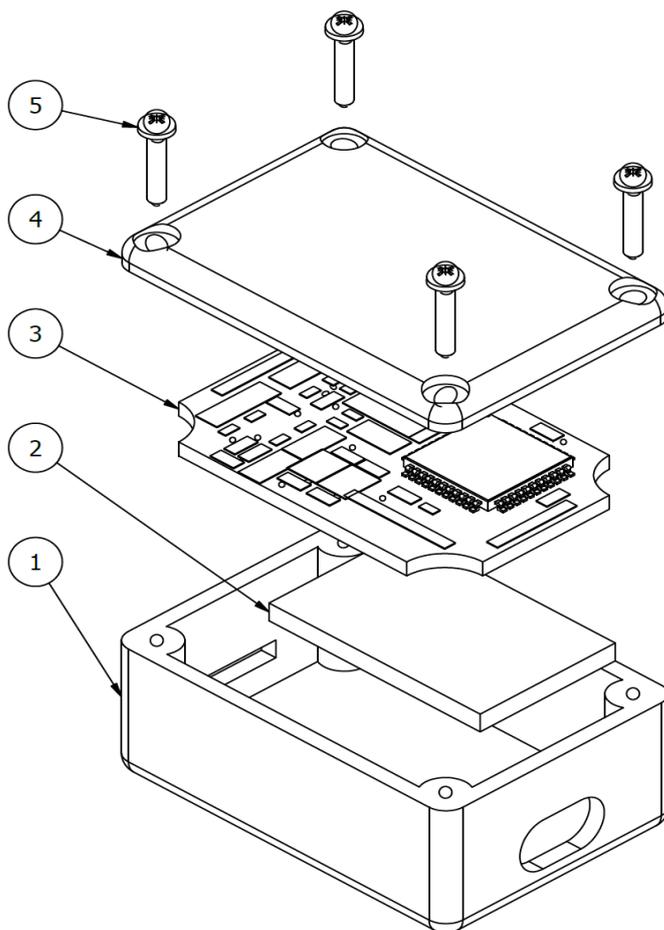
Para realizar el programa y programar el microcontrolador se ha empleado el MPLAB IDE v8.92 con el compilador PCWHD de la empresa CCS (debido a su optimización adicional respecto al incorporado en MPLAB) y el programador PICKit 3. En el entorno de desarrollo de MPLAB ya está incluido el archivo de tipo header necesario para utilizar el microcontrolador PIC24FJ64GA004 llamado "24FJ64GA004.h".

### 4.4 Visualización de los datos desde el ordenador

Para la visualización de los datos se ha empleado el programa Serial Input/Output Monitor incorporado en la suite del compilador PCWHD que es compatible con la función `fprintf` usada. Este programa permite tanto visualizar la temperatura enviada al ordenador en tiempo real como almacenar un conjunto de temperaturas y exportarlas en un documento de texto que luego puede ser procesado por Matlab para generar gráficas y realizar operaciones matemáticas. Para configurarlo simplemente basta con seleccionar el puerto de comunicaciones donde esté conectado el dispositivo, seleccionar el ratio de baudios 9600, 8 bits de datos y sin Paridad para comprobar error en la transmisión.

## **CAPÍTULO 5. DISEÑO DEL PROTOTIPO Y MONTAJE**

Una vez diseñada la placa de circuito impreso y conocido su tamaño es el momento de escoger una caja donde poder introducirla junto con el convertidor de TTL a micro USB necesario para conectarlo al ordenador. Tras mirar una gran cantidad de ellas, finalmente se escogió la caja CAMDENBOSS RX2006/S-FIVES de color negro la cual tiene una tapa atornillada para proteger el interior y tiene las medidas más pequeñas que permiten introducir toda la electrónica en su interior sin problemas. Dado que no tiene ningún agujero, se deberá de hacer los dos agujeros necesarios (uno para el conector micro USB y el otro para la conexión con la Pt100). No se realizará un agujero para la conexión que permite programar el microcontrolador, por lo tanto, para cambiar el programa será necesario abrir la tapa. En la Figura 14 se puede observar como en la parte superior se sitúa la placa de circuito impreso y en la inferior el convertidor con salida Micro USB, situándose a cada lado de la caja el agujero correspondiente a cada una de las interfaces presentes.



Lista de piezas:

1. Caja
2. Convertidor TTL a Micro USB
3. Placa de circuito impreso
4. Tapa
5. Tornillos

Figura 14: Plano de conjunto del prototipo (extracto del plano 5)

Para conectar ambos componentes es necesario emplear 4 cables que saldrán de la parte inferior de la placa de circuito impreso por la parte de la derecha y se conectarán a la parte izquierda del convertidor, el cual estará en el centro de la caja tocando la pared donde se encuentra la salida del micro USB. Los dos agujeros (tanto el redondeado como el rectangular) necesarios en la caja se realizarán con un pequeño taladro con una broca indicada para plástico ABS con un posterior lijado para tener un buen acabado. Las medidas exactas de los agujeros que hay que realizar en la caja se encuentran en el plano 5 junto con una representación del conjunto. Para conocer todas las medidas de la caja se debe consultar el plano facilitado por el fabricante RS [9]. Para fijar las piezas se empleará en el convertidor y en los conectores al exterior pegamento termofusible, que se aplicará en el interior de la caja.

La forma del agujero del micro USB es debida a que se ha empleado un cable especial micro USB a USB-A en el cual el terminal micro USB se conecta a una base donde se adhiere un imán que está conectado al resto del cable, como se puede ver en la Figura 15.



Figura 15: Terminación del cable micro USB magnetizado Fonken de un metro

Por último, en la Figura 16, Figura 17 y Figura 18 se puede observar el prototipo montado:



Figura 16: Imagen del prototipo con la tapa por el lado que se conecta el micro USB

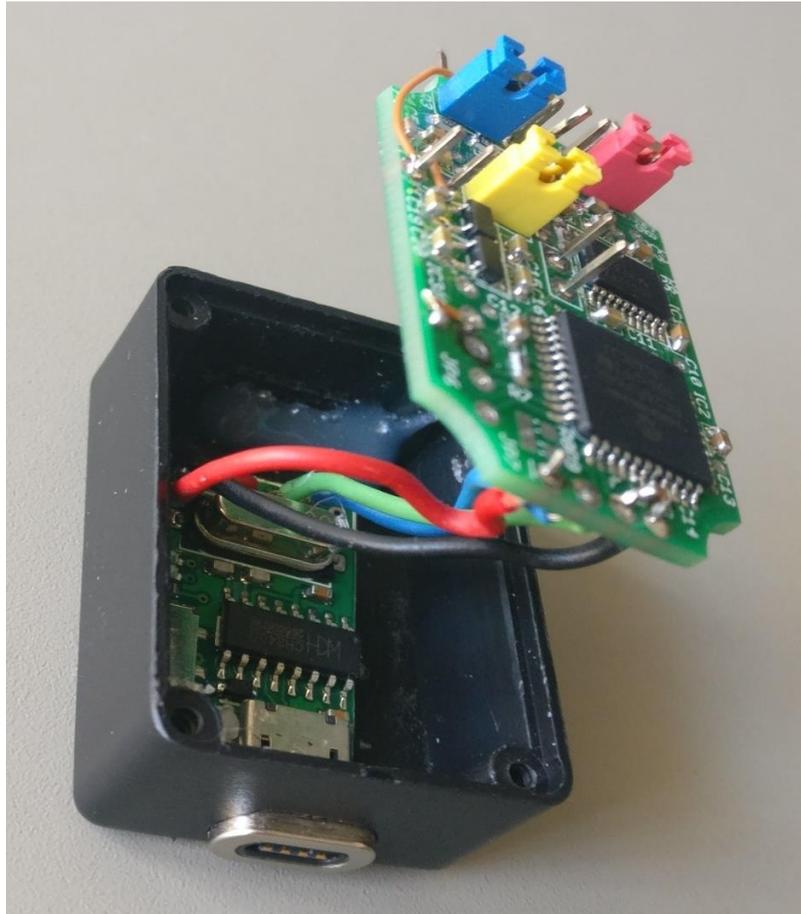


Figura 17: Imagen del prototipo ya montado sin la tapa

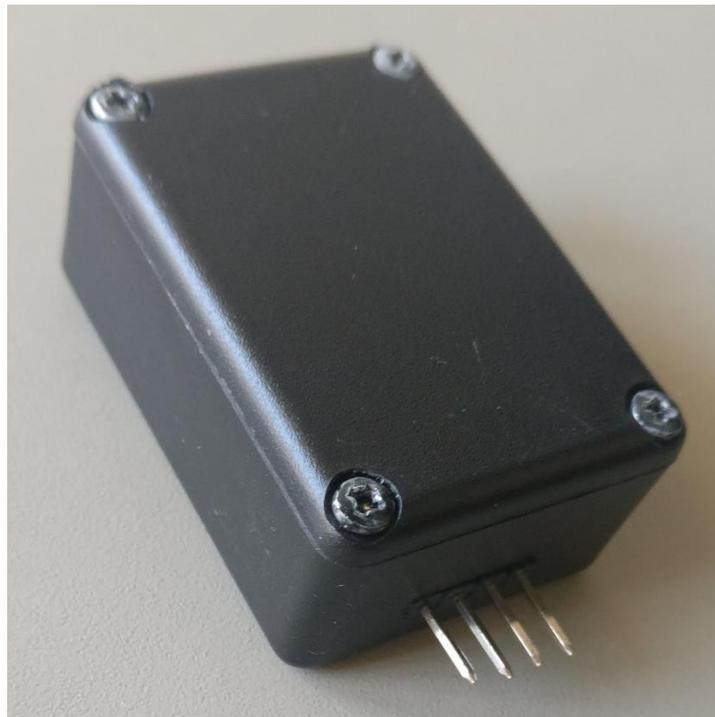


Figura 18: Imagen del prototipo con la tapa por el lado que se conecta la Pt100

## **CAPÍTULO 6. ENSAYOS Y AJUSTES**

Una vez construido el prototipo se llevan a cabo una serie de ensayos (Figura 19) para comprobar su error de medición, comparando a su vez dos Pt100 de gamas y precios distintos. A su vez se procede a emplear las diversas formas de tomar las mediciones del dispositivo sobre el mejor sensor usado. Para ello será necesario tener un ordenador cercano al dispositivo y será recomendable mantener en todo momento el programador PICKit 3 conectado a la placa de circuito impreso. Esta recomendación es debida a que una vez el sensor esté a la temperatura deseada el cambio entre los distintas configuraciones deberá de ser rápido para que los resultados no se vean perjudicados.

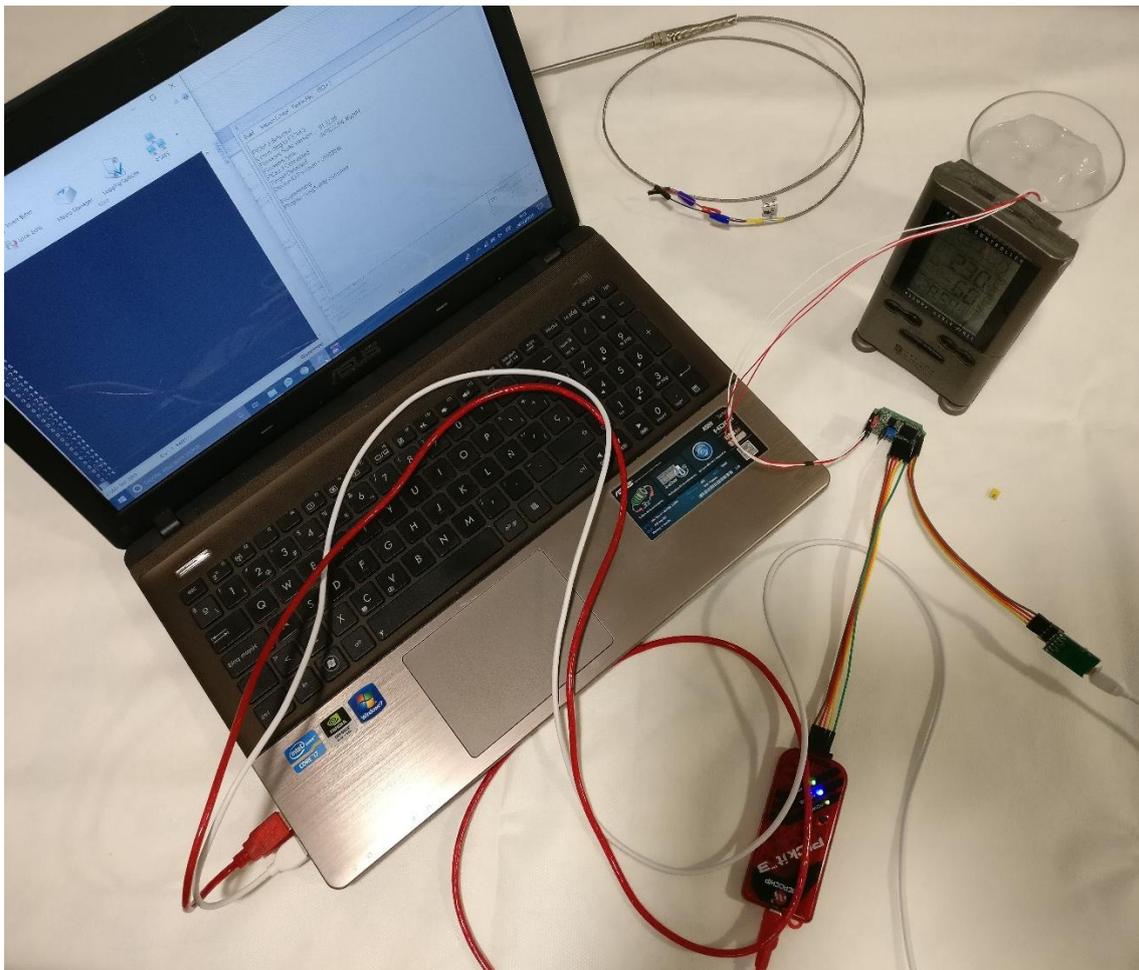


Figura 19: Preparación del ensayo de la Pt100 de 4 hilos con agua a 0°C

Para los ensayos se emplean los siguientes sensores de temperatura:

- Pt100 de 4 hilos (Figura 20) clase B con una tolerancia de  $\pm 0.3^{\circ}\text{C}$  a  $0^{\circ}\text{C}$  y de  $\pm 0,8$  a  $100^{\circ}\text{C}$  (valores basados en la Tabla 3 del documento europeo EN 60751:2008 [6]) con unos cables de conexión de medio metro de longitud.

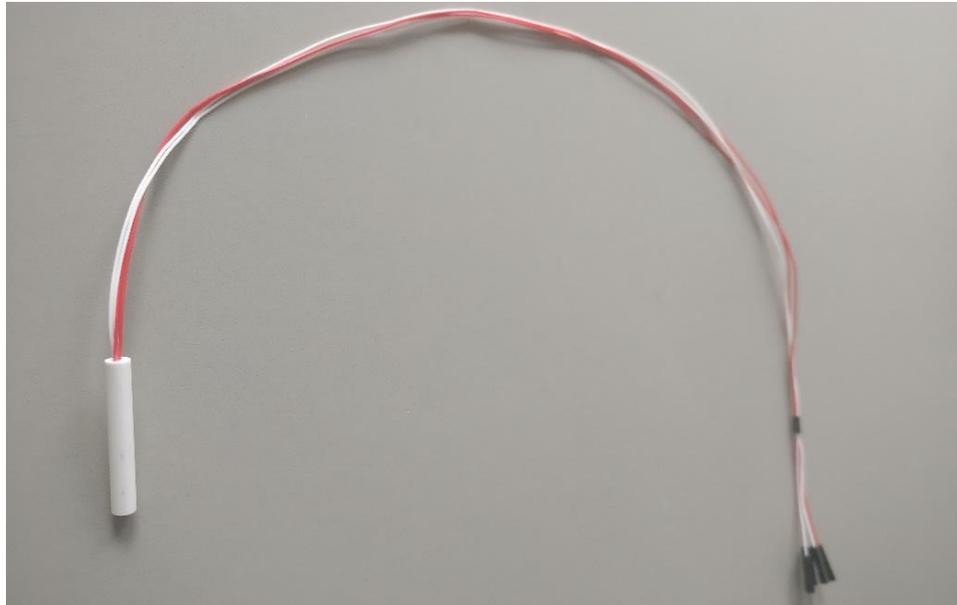


Figura 20: Sensor de temperatura Pt100 4 hilos RS PRO de clase B

- Pt100 de 3 hilos (Figura 21) con bajo coste comprada en china con tolerancia desconocida con unos cables de conexión de un metro de longitud.



Figura 21: Sensor de temperatura Pt100 de 3 hilos

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

---

Los ensayos se realizarán midiendo agua en estado líquido a dos temperaturas, a 0°C (Tabla 1Figura 22) y a 100°C (Figura 23). Se escogen estas temperaturas debido a que son las temperaturas de cambio de estado del agua en estado líquido a sólido (solidificación) y a gaseoso (ebullición). Gracias a producirse un cambio de estado la temperatura permanece constante en un valor conocido y podemos valorar con mayor exactitud el error que se da en la medida.



Figura 22: Ensayo con agua a 0°C de la Pt100 de 3 hilos



Figura 23: Preparación del ensayo de la Pt100 de 4 hilos con agua a 100°C

Los datos de los ensayos realizados se encuentran en la Tabla 1 (Tabla 13) y en la Tabla 14.

Tabla 13: Ensayo del dispositivo midiendo agua a 0°C

Configuración	Pt100 4 hilos	Pt100 3 hilos
2 hilos $V_{REF}$	1,12°C	--
3 hilos $V_{REF}$	0,6°C	10,64°C
3 hilos $R_{REF}$	-0,89°C	8,8°C
3 hilos $V_{REF} R_{COMP}$	0,14°C	--
3 hilos $R_{REF} R_{COMP}$	-1,1°C	--
4 hilos $V_{REF}$	0,58°C	--
4 hilos $R_{REF}$	-0,91°C	--

Tabla 14: Ensayo del dispositivo midiendo agua a 100°C

Configuración	Pt100 4 hilos	Pt100 3 hilos
2 hilos $V_{REF}$	101,03°C	--
3 hilos $V_{REF}$	100,7°C	101.34°C
3 hilos $R_{REF}$	98,92°C	99.41°C
3 hilos $V_{REF} R_{COMP}$	100,15°C	--
3 hilos $R_{REF} R_{COMP}$	98,6°C	--
4 hilos $V_{REF}$	100,5°C	--
4 hilos $R_{REF}$	99°C	--

En primer lugar, la Pt100 de 3 hilos barata ofrece un rendimiento insuficiente a 0°C de temperatura teniendo un error de hasta 10°C. El hecho de que a 100°C ofrezca un rendimiento mucho mejor, estando cercana a la Pt100 de clase B nos indica la escasa linealidad que posee y que está diseñada para comportarse a más altas temperaturas. A partir de esta observación se puede extraer como conclusión que es importante adquirir sensores probados con tolerancias homogéneas y conocidas. Por tanto, por mucho que se utilice un dispositivo muy preciso a la hora de leer el voltaje del sensor de temperatura, si este sensor presenta en sí mismo un gran error no se podrá corregir y la medida será imprecisa.

En segundo lugar, la resistencia de compensación es contraproducente ya que no es necesaria para facilitar la conversión del voltaje a temperatura (debido a que se hace todo mediante software) y añade un error adicional debida a su propia tolerancia y a variabilidad que puede tener la corriente generada por el ADS1247. Por lo tanto, las dos configuraciones en las que se emplea no deberían de ser consideradas para la ejecución de un producto definitivo posterior a la fase de prototipo actual.

En lo que respecta a los distintos tipos de conexión, se puede observar que el incremento de error al usar 2 hilos es apreciable respecto al uso de 3 o 4 hilos. A su vez, debido a la distancia tan pequeña a la que se sitúa el sensor (inferior a 5 metros) no se aprecian diferencias relevantes entre la configuración de 3 o 4 hilos. En consecuencia, el sensor Pt100 de 4 hilos usado podría ser usado en un dispositivo que solo soportase Pt100 de 3 hilos sin apreciar un aumento significativo en cuanto a error en la medida.

Por último, se aprecian diferencias notables en cuanto al error adicional introducido debido a la resistencia de referencia respecto al voltaje interno de bajo ruido proporcionado por el ADS1247. Para obtener una referencia mejor que el voltaje de referencia interno se necesitaría una resistencia de menor tolerancia, siendo recomendable buscar una de 0,01% ya que como se ha visto en este ensayo una de 0,1% de tolerancia no ha sido suficiente.

Tras analizar los ensayos se puede observar cierta correlación entre el error producido a 0°C y el observable a 100°C. Una forma de mejorar estas mediciones sería aplicar una corrección en la fórmula que obtiene la temperatura basada en la resistencia de la Pt100. En este caso, restar -0,4°C si el rango de operación es entre 0 y 100°C parece razonable y mejoraría el error casi eliminándolo. Debido al error en la linealidad que las Pt100 tienen, en caso de hacer este tipo de corrección habría que tomar la mayor cantidad de datos posibles para hacer una corrección con seguridad, siendo recomendable incluso emplear otros termómetros calibrados externamente para la comparación.

## **CAPÍTULO 7. CONCLUSIONES**

En conclusión, un equipo de medida de temperatura basado en una Pt100 es capaz de cumplir todos los objetivos que se habían marcado: preciso, fiable, de reducido tamaño, flexible y con un consumo de energía reducido. Gracias al diseño de la placa de circuito impreso se ha podido lograr el menor tamaño posible y poder emplear cualquier tipo de Pt100, incluso empleando distintos métodos de obtención de los datos al configurar el ADS1247. Debido a la necesidad de tener que escoger componente a componente se puede comprender mejor el funcionamiento y las limitaciones de cada una de las partes del sistema. Una mayor complejidad a la hora de programar este tipo de sistemas se ve compensada con la gran cantidad de información presente en los documentos del fabricante tanto del analog front-end ADS1247 como del microcontrolador usado.

Se ha probado de gran utilidad emplear herramientas del propio fabricante del microcontrolador como el entorno desarrollo MPLAB y el programador PICkit3 debido a su elevada compatibilidad y baja tasa de fallos. Aun así, la optimización adicional otorgada por el compilador PCWHD de CCS frente al compilador incorporado en MPLAB hace que compense su precio, sobre todo en dispositivos muy sensibles a la velocidad de respuesta y optimización de recursos.

Los sensores de temperatura Pt100 otorgan al sistema una gran adaptabilidad a la situación en la que tenga que ser usado tanto en tamaños como en rangos de temperatura. Como se ha podido observar en los ensayos, del sensor que se conecte al dispositivo dependerá en gran medida la precisión a la hora de tomar la medida al ser el dispositivo capaz de aprovechar hasta las Pt100 más precisas. Sin embargo, cabe destacar que a no ser que el sensor esté a mucha distancia no se aprecia gran diferencia entre usar configuraciones de 3 o 4 hilos, pero sí que se nota la tolerancia determinada por la clase a la que pertenece.

Finalmente, gracias a la elaboración de este proyecto se ha podido profundizar en el funcionamiento de sistemas electrónicos embebidos desde su concepción hasta su implementación. Por lo tanto, un análisis de este tipo sobre un dispositivo de medición de temperatura ha sido muy enriquecedor y esto otorga una experiencia aplicable a la hora de afrontar otros proyectos.

## **REFERENCIAS**

[1] Imagen creada por SRC:

<https://srcsl.com/termopares-la-ciencia-de-la-termometria/> (17/05/2019).

[2] Imagen creada por "Alex7":

<https://alex7tutoriales.com/elementos-primarios-de-temperatura/> (17/05/2019).

[3] Imagen de un termistor fabricado por DACHS:

<https://www.dachs.es/proteccion-circuitos/termistores> (17/05/2019).

[4] Imagen de un ADS1247 de la página oficial de Texas Instruments:

<http://www.ti.com/product/ADS1247#> (17/05/2019).

[5] Imagen de un microcontrolador PIC24FJ64GA004 de la página oficial de Microchip:

<https://www.microchip.com/wwwproducts/en/PIC24FJ64GA004> (17/05/2019).

[6] Documento Europeo EN 60751:2008. Termómetros industriales de resistencia de platino y sensores de temperatura de platino.

[7] ADS124x 24-Bit, 2-kSPS, Analog-To-Digital Converters With Programmable Gain Amplifier (PGA) For Sensor Measurement datasheet (Rev. H). 31/03/2016.

Enlace: <http://www.ti.com/lit/ds/symlink/ads1247.pdf> (17/05/2019).

[8] Datasheet de la familia de microcontroladores de 16-bit PIC24FJ64GA004.

Enlace: <http://ww1.microchip.com/downloads/en/DeviceDoc/39881e.pdf> (17/05/2019).

[9] Plano de la Caja de encapsulado CAMDENBOSS RX2006/S-FIVES de color Negro.

Enlace: <https://docs-emea.rs-online.com/webdocs/15e0/0900766b815e0575.pdf>(17/05/2019).

[10] Tema 4: Presupuestos, cuadros de precios y pliego de condiciones. Diapositivas asignatura Proyectos, Escuela Técnica Superior de Ingenieros Industriales de la Universitat Politècnica de València.

[11] Jesús Bausà Aragonés, Carlos García Gómez, Benito Zaragozaí Zaragozaí, Alex Gil Martínez, Daniel Moreno Campos y Antonio Galiana Llinares (2003). Sensores de temperatura.

Enlace: [http://server-die.alc.upv.es/asignaturas/lased/2003-04/0.sens\\_temp/archivos/sensoresde temperatura.pdf](http://server-die.alc.upv.es/asignaturas/lased/2003-04/0.sens_temp/archivos/sensoresde temperatura.pdf) (17/05/2019).

[12] Circuitos microelectrónicos. A.S. Sedra y K.C. Smith. Editorial McGraw-Hill. 2006.

[13] Electrónica: teoría de circuitos y dispositivos electrónicos. R.L. Boylestad y L.Nashelsky. Editorial Prentice Hall. 2009.

[14] Canal de Youtube Microchips Technology. ADC Aspects-Episode 3-Offset and Gain Errors.

Enlace: [https://www.youtube.com/watch?v=ZfUz1s\\_jWmQ](https://www.youtube.com/watch?v=ZfUz1s_jWmQ) (17/05/2019).

[15] Pt 100 RS Pro Datasheet.

Enlace: <https://docs-emea.rs-online.com/webdocs/15bb/0900766b815bb532.pdf> (18/05/2019).

[16] Microchip. 16-Bit MCU and DSC Programmer's Reference Manual.

Enlace: <http://ww1.microchip.com/downloads/en/DeviceDoc/70000157g.pdf> (17/05/2019).

[17] Microchip. MPLAB IDE User's guide.

Enlace: <http://ww1.microchip.com/downloads/en/devicedoc/51519a.pdf> (17/05/2019).

[18] Instrumentación Electrónica. Miguel A. Pérez García, et al. Editorial Thomson Editores Spain Paraninfo S.A. 2004.

[19] Electrónica. A.R. Hambley. Editorial Prentice HALL. 2008.

---

DOCUMENTO N°2:

# **PRESUPUESTO**

---

## PRESUPUESTO

En este documento se desarrolla el presupuesto correspondiente a este proyecto. El presupuesto está dividido en una serie de tablas gracias a las cuales se ha podido, a partir de los costes de los elementos por separado, plantear el coste total del desarrollo de este proyecto.

### 1 Cuadros de precios

#### 1.1 Materiales

Para la realización del proyecto se debe de tener en cuenta que el tamaño de componentes es de 0603 [Métrica 1608]. También es importante destacar que los precios son de la tienda Farnell.

Tabla 1: Cuadro de precios de los materiales

Descripción	Coste (€/unidad)
10 PCB de dos capas verde hecho en ITEAD	28,52 €
Caja negra CAMDENBOSS RX2006/S-FIVES con medidas de 40 x 28 x 18mm (Referencia Farnell: 1871051)	13,74 €
Texas Instruments ADS1247IPW (Referencia Farnell: 1735556)	10,37 €
MICROCHIP PIC24FJ64GA004-I/PT (Referencia Farnell: 1332299)	2,87 €
Regulador de tensión de 4,3V-12V a 3,3V NCP551SN33T1G (R. F.: 1652368)	0,47 €
Condensador GRM188R60J106ME47D 10 µF (Referencia Farnell: 1735528)	0,20 €
Condensador 0603YD105KAT2A 1 µF (Referencia Farnell: 1327680)	0,16 €
Condensador C0603X5R1A104K030BC 100 nF (Referencia Farnell: 2524852)	0,13 €
Condensador MC0603B682K500CT 6,8 nF (Referencia Farnell: 1759100)	0,02 €
Resistencia MCSR06X5101FTL 5.1 kΩ, tolerancia 1% (Referencia Farnell:2074275)	0,01 €
Resistencia ACPP06034K7BE 4,7 kΩ, tolerancia 0,1% (Referencia Farnell: 2504957)	0,25 €
Resistencia WINT0603LF111000A 100 Ω, 0.05% (Referencia Farnell: 2576324)	0,97 €
Resistencia CPF0603B1K33E1 1.33 kΩ, 0,1% (Referencia Farnell: 2331003)	0,36 €
40 Cables hembra-hembra para GPIO de 10cm	4,23 €
Jumper M7581-05 con punto de prueba (Referencia Farnell: 2396303)	0,08 €
Convertidor TTL a micro USB	7,99 €
Sensor Pt100 RS PRO (+250°C, diámetro 5mm y Clase B)	30,13 €
Sensor Pt100 de bajo coste de 1 metro con rosca de 8mm acabado metálico	4,86 €
Cable micro USB magnetizado Fonken de un metro	3,06 €

## 1.2 Mano de obra

Se considera que el proyecto ha sido realizado por un ingeniero en Tecnologías Industriales recién titulado.

Tabla 2: Cuadro de precios de la mano de obra

Descripción	Coste (€/h)
Ingeniero en Tecnologías Industriales recién titulado	20,00 €

## 1.3 Equipos amortizados linealmente

Para el estudio pormenorizado de la amortización lineal de los equipos empleados se han seleccionado solo aquellos que se utilizan al menos 2 días.

Tabla 3: Cuadro de precios de los equipos de uso habitual

	Precio de compra	Precio residual	Vida útil (meses)	Coste (€/mes)
Compilador CCS PCWHD	600 €	0 €	36	16,70 €
PICKit 3	16,80 €	0 €	24	0,70 €

## 2 Cuadro de precios descompuestos

El presupuesto total se ha dividido en dos capítulos: el diseño del sistema y la realización de los ensayos, donde está incluida la construcción del prototipo. En la tienda Farnell (donde se ha consultado el precio de los componentes electrónicos) algunos componentes electrónicos no pueden ser pedidos individualmente si no que es obligatorio pedirlos en bloques de 10. Los componentes que sobren serán guardados y usados en caso de avería u otros imprevistos.

Tabla 4: Cuadro de precios descompuestos

Código UO	Ud	Descripción	Medición	Precio unitario	Importe
<b>1</b>		<b>Diseño del sistema</b>			
<b>1.1</b>		<b>Diseño del circuito electrónico y la placa de circuito impreso</b>			
	h	Ingeniero en Tecnologías Industriales recién titulado	100	20	2.000 €
				Coste total:	2.000 €

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

<b>1.2</b>		<b>Realización de los programas</b>			
	mes	PCWHD (compilador CCS) usado durante 3 años	2	16,7	33,40 €
	mes	PICkit 3 (programador y depurador de microcontroladores PIC)	2	0,7	1,40 €
	h	Ingeniero en Tecnologías Industriales recién titulado	120	20	2.400 €
				Coste total:	2.434,80 €
<b>1.3</b>	Ud	Diseño del encapsulado final			
	h	Ingeniero en Tecnologías Industriales recién titulado	10	20	200 €
				Coste total:	200 €
<b>2</b>		<b>Realización de ensayos</b>			
<b>2.1</b>		<b>Construcción del prototipo</b>			
	Ud	10 PCB de dos capas verde hecho en ITEAD	1	28,52	28,52 €
	Ud	Caja negra CAMDENBOSS RX2006/S-FIVES con medidas de 40 x 28 x 18mm (Referencia Farnell: 1871051)	1	13,74	13,74 €
	Ud	Texas Instruments ADS1247IPW (Referencia Farnell: 1735556)	1	10,37	10,37 €
	Ud	MICROCHIP PIC24FJ64GA004-I/PT (Referencia Farnell: 1332299)	1	2,87	2,87 €
	Ud	Regulador de tensión de 4,3V-12V a 3,3V NCP551SN33T1G (R. F.: 1652368)	2	0,471	0,94 €
	Ud	Condensador GRM188R60J106ME47D 10 $\mu$ F (Referencia Farnell: 1735528)	10	0,197	1,97 €
	Ud	Condensador 0603YD105KAT2A 1 $\mu$ F (Referencia Farnell: 1327680)	10	0,16	1,60 €
	Ud	Condensador C0603X5R1A104K030BC 100 nF (Referencia Farnell: 2524852)	10	0,131	1,31 €
	Ud	Condensador MC0603B682K500CT 6,8 nF (Referencia Farnell: 1759100)	10	0,0199	0,20 €
	Ud	Resistencia MCSR06X5101FTL 5.1 k $\Omega$ , tolerancia 1% (Referencia Farnell:2074275)	10	0,0087	0,09 €
	Ud	Resistencia ACPP06034K7BE 4,7 k $\Omega$ , tolerancia 0,1% (Referencia Farnell: 2504957)	4	0,246	0,98 €
	Ud	Resistencia WINT0603LF111000A 100 $\Omega$ , 0.05% (Referencia Farnell: 2576324)	2	0,972	1,94 €

Diseño e implementación de un equipo de medida de temperatura de alta resolución basado en el analog front-end ADS1247.

	Ud	Resistencia CPF0603B1K33E1 1.33 kΩ, 0,1% (Referencia Farnell: 2331003)	1	0,362	0,36 €
	Ud	40 Cables hembra-hembra para GPIO de 10cm	1	4,23	4,23 €
	Ud	Jumper M7581-05 con punto de prueba (Referencia Farnell: 2396303)	10	0,077	0,77 €
	Ud	Convertidor TTL a micro USB	1	7,99	7,99 €
	Ud	Sensor Pt100 RS PRO (+250°C, diámetro 5mm y Clase B)	1	30,13	30,13 €
	Ud	Sensor Pt100 de bajo coste de 1 metro con rosca de 8mm acabado metálico	1	4,86	4,86 €
	Ud	Cable micro USB magnetizado Fonken de un metro	1	3,06	3,06 €
	min	Aplicación de pegamento termofusible con pistola	15	0,15	2,25 €
	min	Taladro eléctrico con broca para perforar plástico	10	0,6	6,00 €
	min	Lija con mango para agujeros de plástico	10	0,2	2,00 €
	h	Ingeniero en Tecnologías Industriales recién titulado	20	20	400 €
				Coste total:	526,19 €
<b>2.2</b>		<b>Toma de medidas y análisis</b>			
	h	Un litro de agua mantenida a 0º gracias al uso de hielo	2	0,4	0,80 €
	h	Un litro de agua mantenida a 100ºC	2	0,2	0,40 €
	h	Ingeniero en Tecnologías Industriales recién titulado	50	20	1.000 €
				Coste total:	1.001,20 €

### 3 Cuadro de precios unitarios

En la siguiente tabla se presenta el precio de cada unidad de obra.

Tabla 5: Cuadro de precios unitarios

Código	Descripción de la unidad de obra	Importe
1.1	Diseño del circuito electrónico y la placa de circuito impreso	2.000,00 €
1.2	Realización de los programas	2.434,80 €
1.3	Diseño del encapsulado final	200,00 €
2.1	Construcción del prototipo	526,19 €
2.2	Toma de medidas y análisis	1.001,20 €

#### 4 Cuadro de presupuesto base de licitación

Tabla 6: Cuadro de presupuesto base de licitación

Nº Capítulo	Nombre Capítulo	Importe
1	Diseño del sistema	4.634,80 €
2	Realización de ensayos	1.527,39 €
Presupuesto de Ejecución Material:		6.162,19 €
Gastos generales (13%):		801,08 €
Beneficio industrial (6%):		369,73 €
Presupuesto de ejecución por contrata:		7.333,00 €
IVA (21%):		1.539,93 €
Presupuesto de Licitación:		8.872,93 €

El presupuesto base de licitación asciende a OCHO MIL OCHOCIENTOS SETENTA Y DOS EUROS y NOVENTA Y TRES CÉNTIMOS.

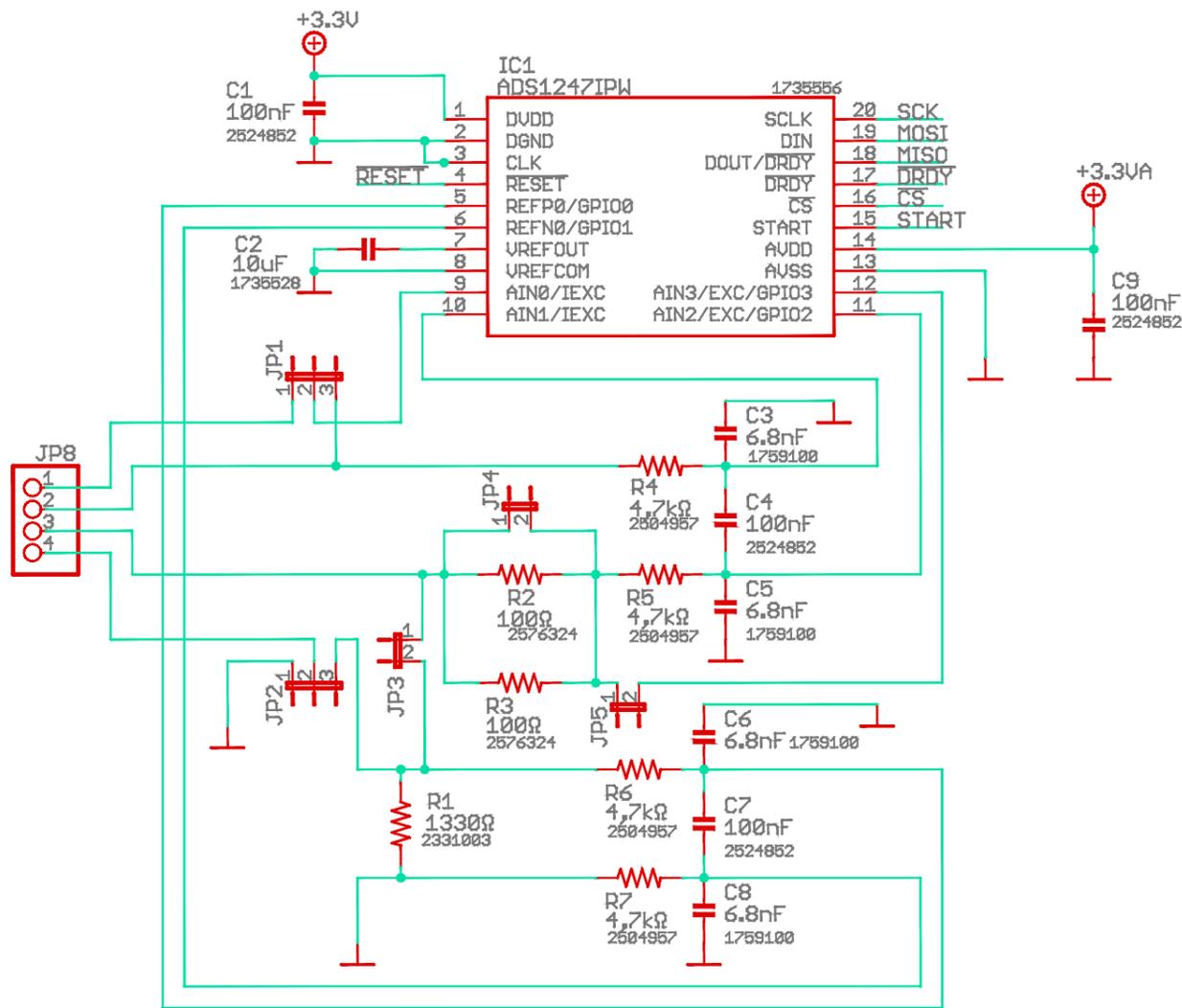
---

DOCUMENTO N°3:

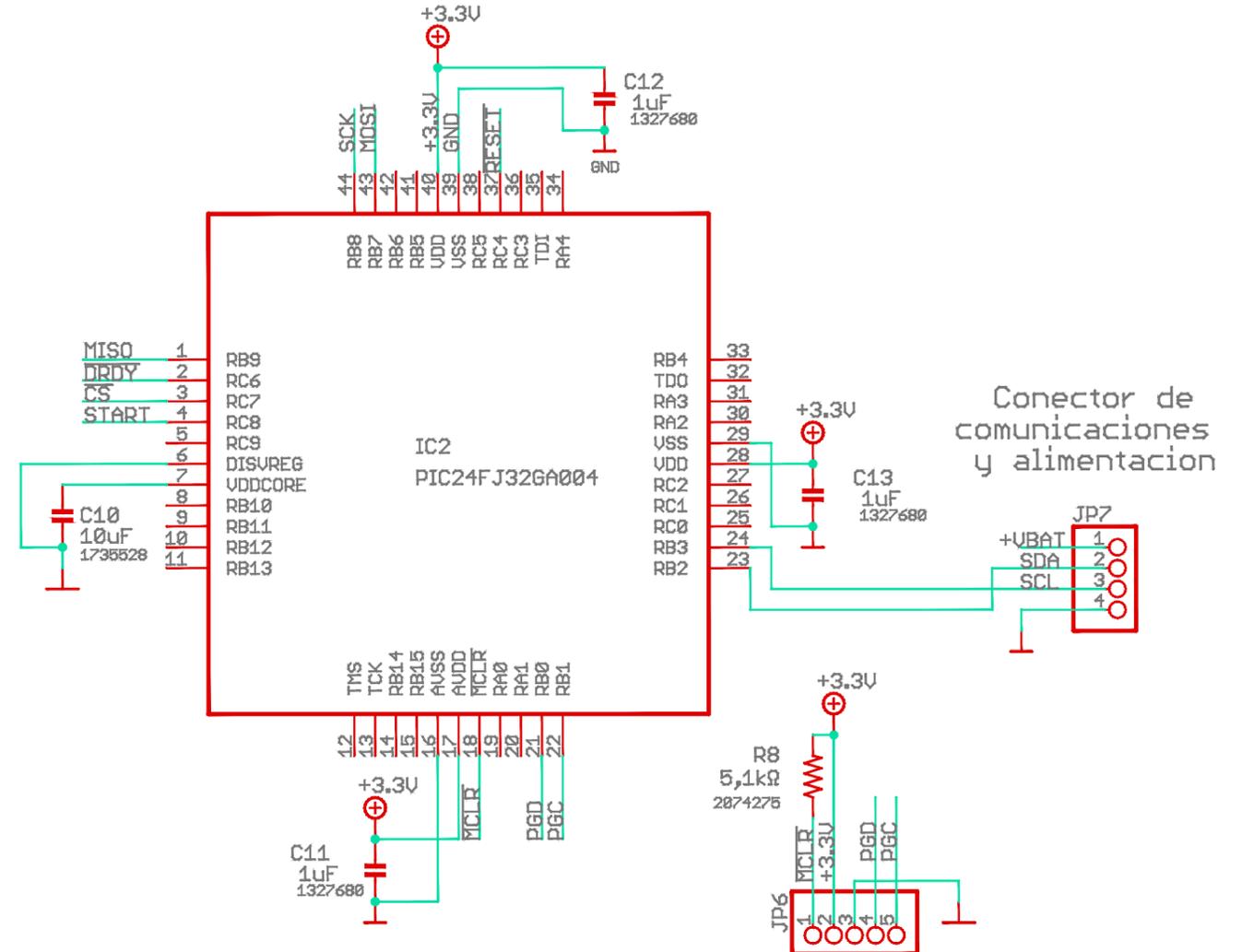
# **PLANOS**

---

## Analog Front-End para medida de temperatura con Pt100



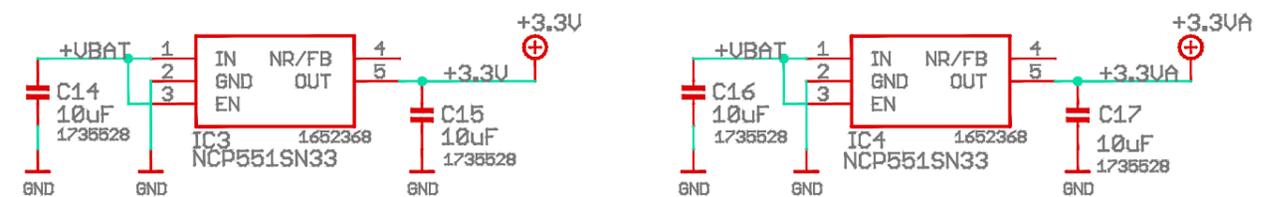
## Microcontrolador

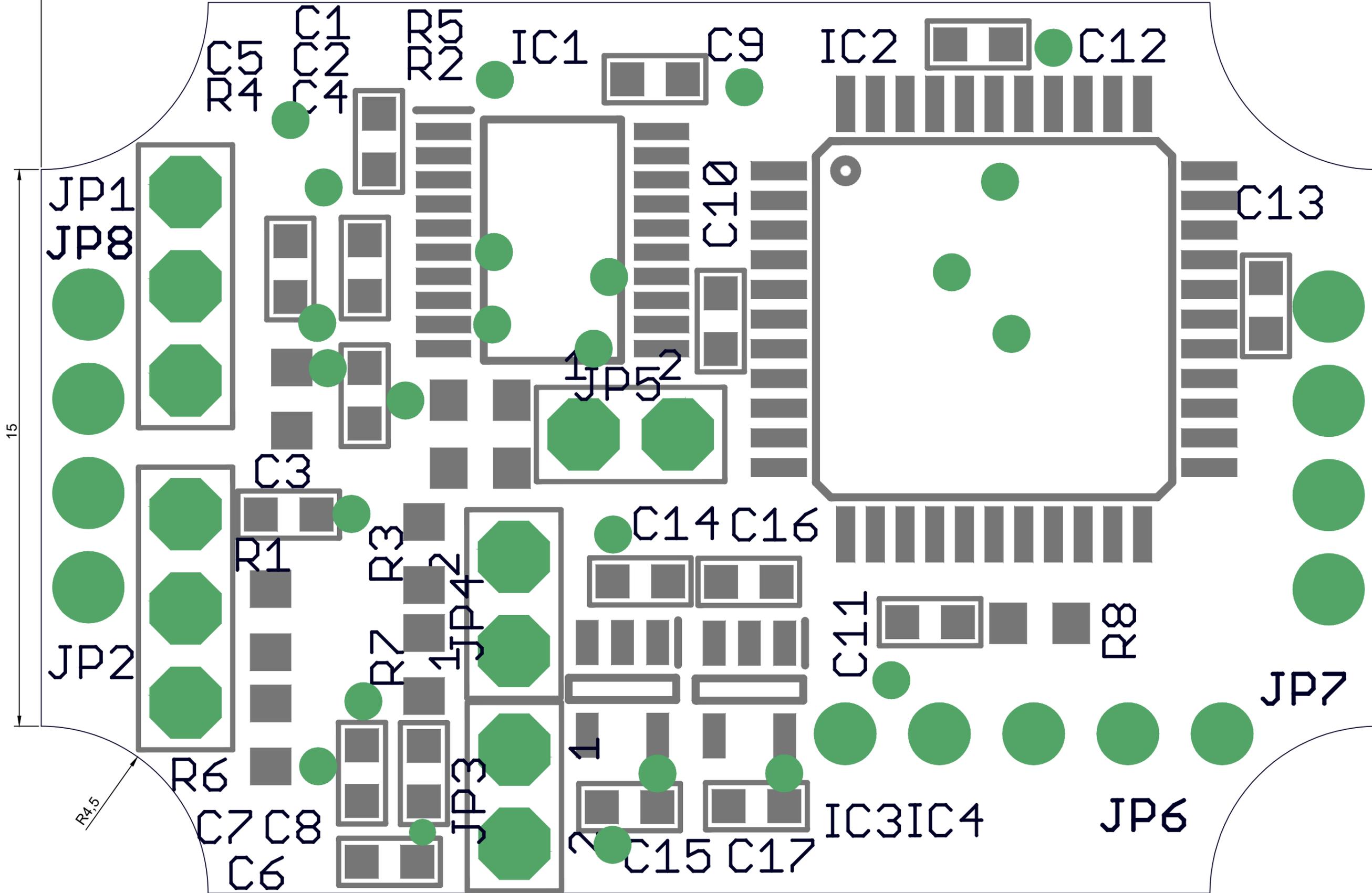


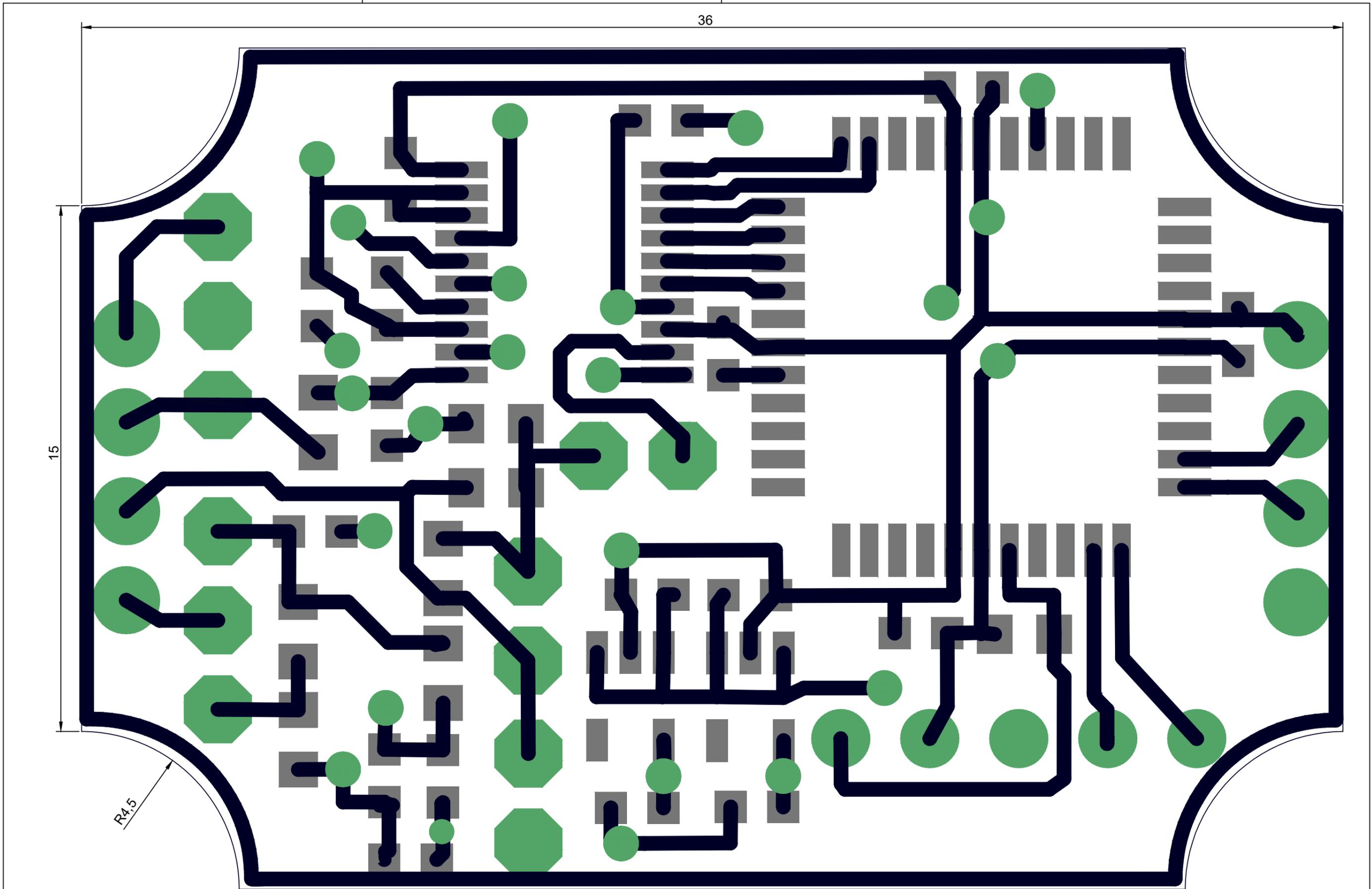
Configuracion	JP1	JP2	JP3	JP4	JP5
2-hilos	23	23	c	c	nc
3-hilos	23	23	nc	c	c
3-hilos+Rref	23	23	nc	c	c
3-hilos+Rcomp	23	23	nc	nc	c
3-hilos+Rref+Rcomp	23	23	nc	nc	c
4-hilos	12	23	nc	c	nc
4-hilos+Rref	12	23	nc	c	nc

nc = no conectado  
c = conectado

## Reguladores de tensión (digital: +3.3V analógico: +3.3V)







36

15

R4.5

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES



UNIVERSITAT POLITÈCNICA DE VALÈNCIA



ESCUELA TÉCNICA SUPERIOR INGENIEROS INDUSTRIALES VALENCIA

Proyecto: EQUIPO DE MEDIDA DE TEMPERATURA DE ALTA RESOLUCIÓN BASADO EN EL ANALOG FRONT-END ADS1247

Plano: Pistas en la cara superior de la placa de circuito impreso

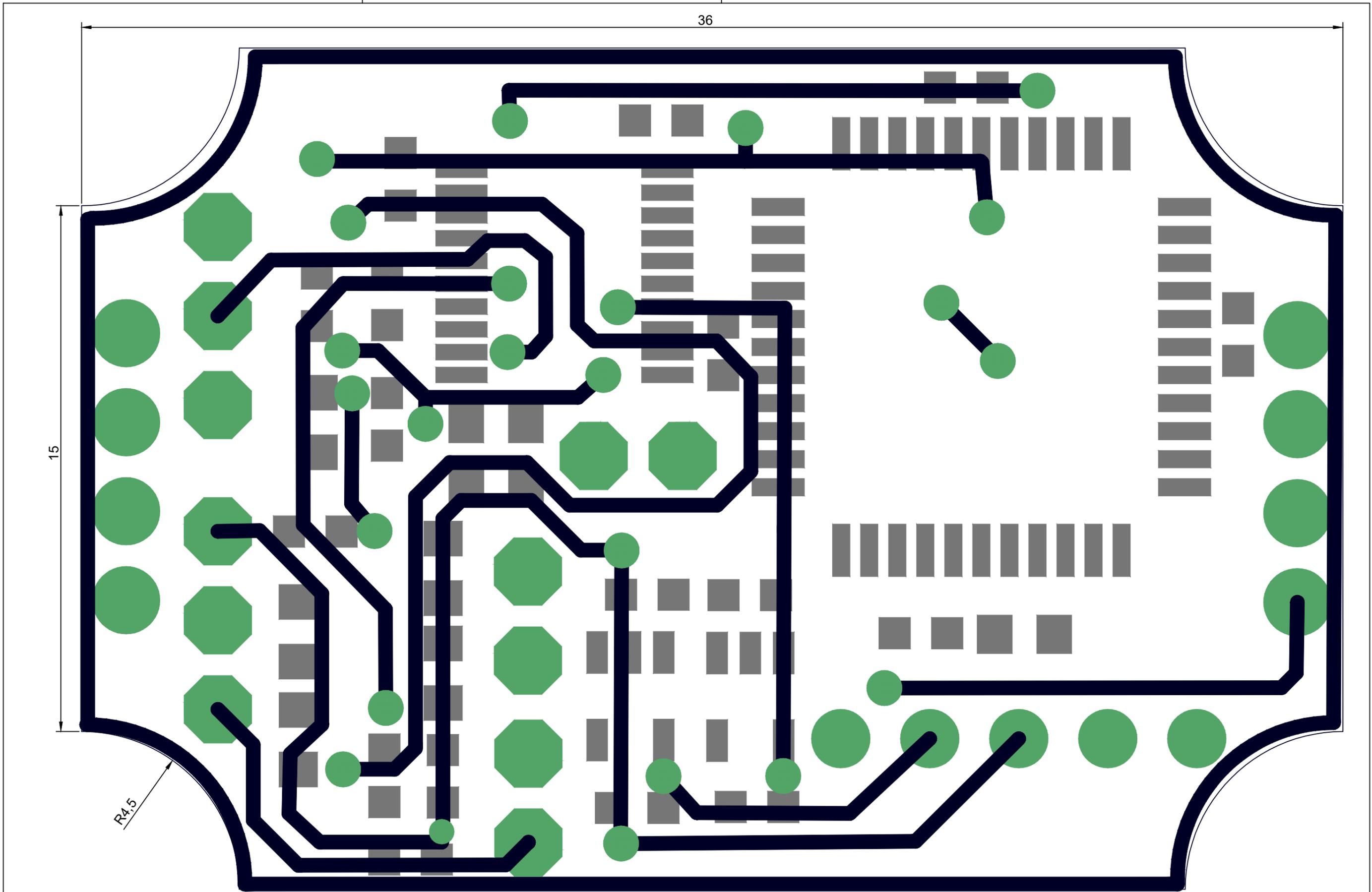
Autor: Víctor Serrano Aladrén

Fecha: Mayo 2019

Escala: 10:1

Nº Plano:

3



36

15

R4.5

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES



Proyecto: EQUIPO DE MEDIDA DE TEMPERATURA DE ALTA RESOLUCIÓN BASADO EN EL ANALOG FRONT-END ADS1247

Plano: Pistas en la cara inferior de la placa de circuito impreso

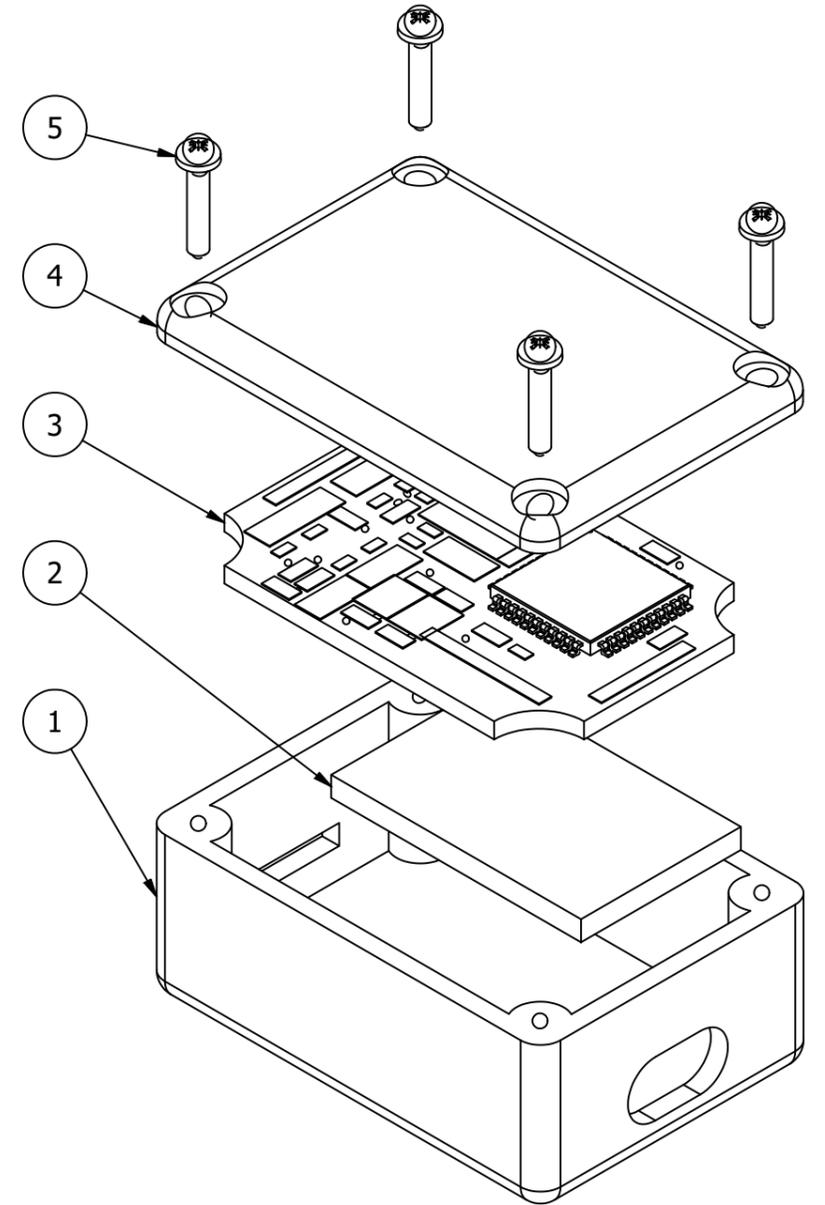
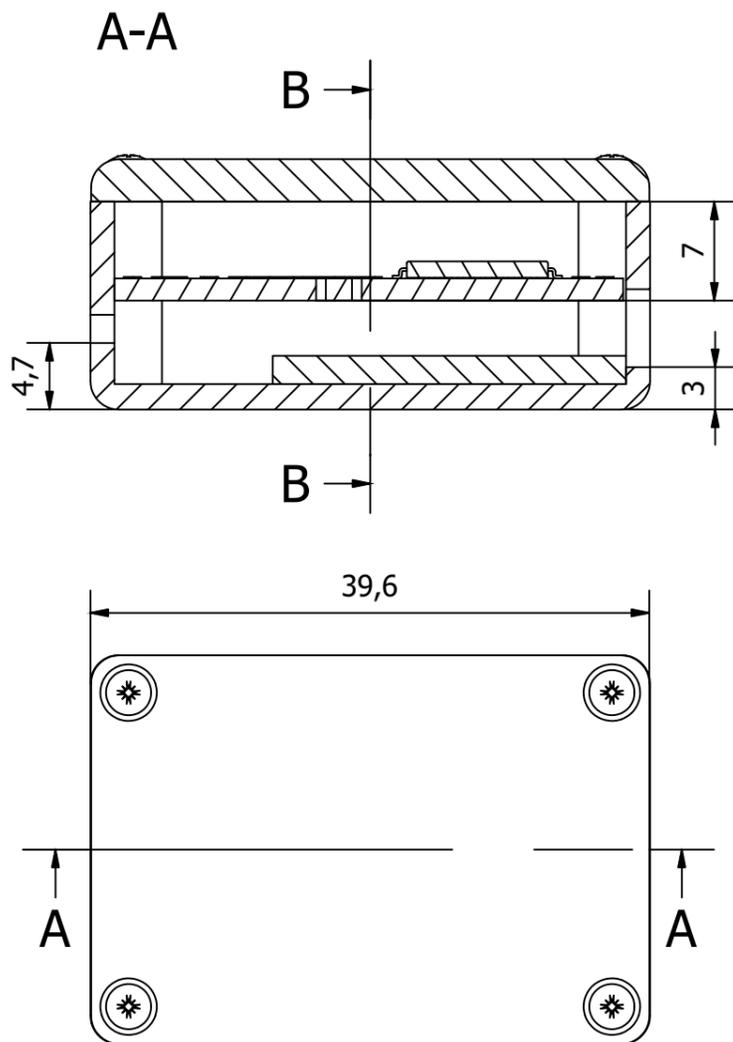
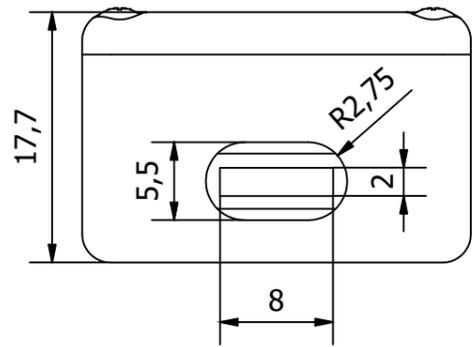
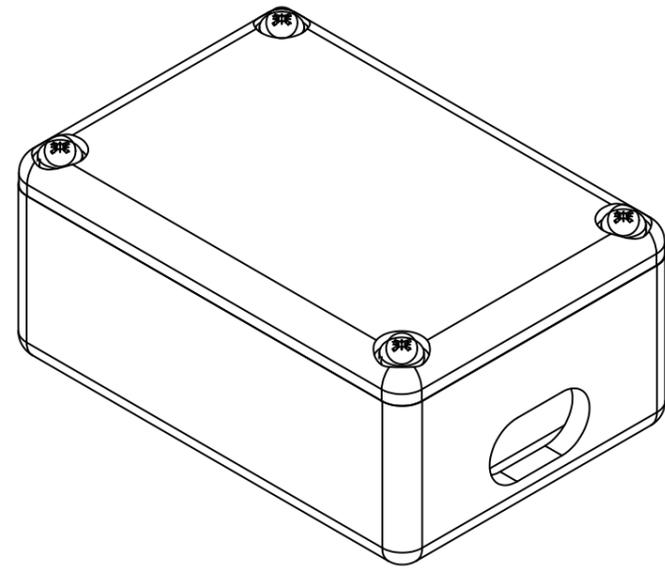
Autor: Víctor Serrano Aladrén

Fecha: Mayo 2019

Escala: 10:1

Nº Plano:

4



5	4	Tornillo de cabeza cilíndrica redondeada
4	1	Tapa
3	1	Placa de circuito impreso
2	1	Convertidor TTL a Micro USB
1	1	Caja CAMDENBOSS RX2006/S-FIVES
MARCA	CANTIDAD	DESCRIPCIÓN

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES



Proyecto: EQUIPO DE MEDIDA DE TEMPERATURA DE ALTA RESOLUCIÓN BASADO EN EL ANALOG FRONT-END ADS1247

Plano: Ensamblaje de las piezas del prototipo

Fecha: Mayo 2019

Nº Plano:

Autor: Víctor Serrano Aladrén

Escala: 2:1

5

---

DOCUMENTO N°4:

**ANEXO**

---

## ANEXO 1: Programas

En el presente anexo se va a exponer el código íntegro de todos los programas empleados para cada una de las configuraciones que pueden emplearse en el dispositivo. El programa base sobre el cual luego se realizarán una serie de ajustes para cada uno de los casos consta de los siguientes archivos con funciones diferenciadas:

- Pt100.c: Configuración del microcontrolador, interrupciones de envío y recepción de datos, conversión de voltaje a temperatura y función principal donde se inicializa el ADS1247.
- ADS1247.h: Configuración del ADS1247 mediante la definición de las funciones empleadas en la función principal

Para realizar el programa y programar el microcontrolador se ha empleado el MPLAB IDE v8.92 con el compilador PCWHD de la empresa CCS y el programador PICKit 3. En este entorno ya está incluido el archivo de tipo header necesario para utilizar el microcontrolador PIC24FJ64GA004 llamado "24FJ64GA004.h".

Como base se va a utilizar el programa empleado en el caso de una Pt100 de 4 hilos medida con el voltaje de referencia interno del ADS1247, el cual es el mismo programa también para el caso de 2 hilos medido con voltaje de referencia interno.

### 1 Pt100.c

#### 1.1 Configuración 4 hilos $V_{REF}$

```
1. #include <24FJ64GA004.h>
2. #fuses NOJTAG // El puerto JTAG está deshabilitado
3. #fuses NOPROTECT // La protección de código está deshabilitada
4. #fuses NOWRT // Escrituras a la memoria del programa permitidas
5. #fuses ICSP1 // Las funciones del emulador son compartidas con PGEC1/PGED1
6. #fuses NOWDT // Watch Dog Timer está deshabilitado; SWDTEN bit está deshabilitado
7. #fuses WDT128 // WDT Prescaler Ratio de 1:128
8. #fuses WPOSTS16 // Watchdog Timer Postscaler Select bits-> 1:32,768 (No usado)
9. #fuses NOIESO // Modo IESO está deshabilitado
10. #fuses FRC_PS // RC Oscillator rápido con Postscaler (FRCDIV)
11. #fuses CKSNOFSM // Clock switching habilitado, Fail -Safe Clock Monitor deshabilitado
12. #fuses OSCIO // OSCO/CLKO/RC15 funciona como puerto I/O (RC15)
13. #fuses NOPR // Modo Primary Oscillator está deshabilitado
```

```
14. //-----
15. #use delay(clock=4000000)
16. //-----
17. #define ON( pin)  output_high(pin)
18. #define OFF(pin)  output_low( pin)
19. //-----
20. #pin_select SDI1 =  PIN_B9  // MISO (Master Input Slave Output)   ( 1)
21. #pin_select INT1 =  PIN_C6  // DRDY: External Interrupt 1      ( 2)
22. #define CS_ADS    PIN_C7  //                               ( 3)
23. #define STRT      PIN_C8  //                               ( 4)
24. #pin_select U1TX =  PIN_B2  // RB2 TX                      (23)
25. #pin_select U1RX =  PIN_B3  // RB3 RX                      (24)
26. #define AD_RESET  PIN_C4  //                               (37)
27. #pin_select SDO1 =  PIN_B7  // MOSI (Master Output Slave Input) (43)
28. #pin_select SCK1OUT= PIN_B8 // SCLK (Clock)                (44)
29. //-----
30. #byte U1STA = 0x0222
31. #byte U1RXREG = 0x0226
32. #use rs232(baud=9600,UART1,stream=TTL) /* Envíos a 9600 baudios en el UART1
    (Universal Asynchronous Receiver-Transmitter) */
33. //-----
34. #byte SPI1BUF = 0x0248
35. #include "ADS1247.h"
36. //-----
37. /*Matriz para obtener la Temperatura en función de la Resistencia
38. Tabla de resistencia de RTD (R x 100) desde
    -200°C a 850°C según el documento europeo EN60751:2008*/
39. const unsigned int16 R[1051]={
40. 1852,1895, 1938, 1982, 2025, 2068, 2111, 2154, 2197, 2240, 2283,
41. 2325, 2368, 2411, 2454, 2497, 2539, 2582, 2624, 2667, 2710,
42. 2752, 2795, 2837, 2880, 2922, 2964, 3007, 3049, 3091, 3134,
43. 3176, 3218, 3260, 3302, 3344, 3386, 3428, 3470, 3512, 3554,
44. 3596, 3638, 3680, 3722, 3764, 3805, 3847, 3889, 3931, 3972,
45. 4014, 4056, 4097, 4139, 4180, 4222, 4263, 4305, 4346, 4388,
46. 4429, 4470, 4512, 4553, 4594, 4636, 4677, 4718, 4759, 4800,
47. 4842, 4883, 4924, 4965, 5006, 5047, 5088, 5129, 5170, 5211,
48. 5252, 5293, 5334, 5375, 5415, 5456, 5497, 5538, 5579, 5619,
49. 5660, 5701, 5741, 5782, 5823, 5863, 5904, 5944, 5985, 6026,
50. 6066, 6107, 6147, 6188, 6228, 6268, 6309, 6349, 6390, 6430,
51. 6470, 6511, 6551, 6591, 6631, 6672, 6712, 6752, 6792, 6833,
52. 6873, 6913, 6953, 6993, 7033, 7073, 7113, 7153, 7193, 7233,
53. 7273, 7313, 7353, 7393, 7433, 7473, 7513, 7553, 7593, 7633,
54. 7673, 7712, 7752, 7792, 7832, 7872, 7911, 7951, 7991, 8031,
55. 8070, 8110, 8150, 8189, 8229, 8269, 8308, 8348, 8387, 8427,
56. 8467, 8506, 8546, 8585, 8625, 8664, 8704, 8743, 8783, 8822,
57. 8862, 8901, 8940, 8980, 9019, 9059, 9098, 9137, 9177, 9216,
58. 9255, 9295, 9334, 9373, 9412, 9452, 9491, 9530, 9569, 9609,
59. 9648, 9687, 9726, 9765, 9804, 9844, 9883, 9922, 9961,10000,
60. 10039,10078,10117,10156,10195,10234,10273,10312,10351,10390,
61. 10429,10468,10507,10546,10585,10624,10663,10702,10740,10779,
62. 10818,10857,10896,10935,10973,11012,11051,11090,11129,11167,
```

63. 11206,11245,11283,11322,11361,11400,11438,11477,11515,11554,  
64. 11593,11631,11670,11708,11747,11786,11824,11863,11901,11940,  
65. 11978,12017,12055,12094,12132,12171,12209,12247,12286,12324,  
66. 12363,12401,12439,12478,12516,12554,12593,12631,12669,12708,  
67. 12746,12784,12822,12861,12899,12937,12975,13013,13052,13090,  
68. 13128,13166,13204,13242,13280,13318,13357,13395,13433,13471,  
69. 13509,13547,13585,13623,13661,13699,13737,13775,13813,13851,  
70. 13888,13926,13964,14002,14040,14078,14116,14154,14191,14229,  
71. 14267,14305,14343,14380,14418,14456,14494,14531,14569,14607,  
72. 14644,14682,14720,14757,14795,14833,14870,14908,14946,14983,  
73. 15021,15058,15096,15133,15171,15208,15246,15283,15321,15358,  
74. 15396,15433,15471,15508,15546,15583,15620,15658,15695,15733,  
75. 15770,15807,15845,15882,15919,15956,15994,16031,16068,16105,  
76. 16143,16180,16217,16254,16291,16329,16366,16403,16440,16477,  
77. 16514,16551,16589,16626,16663,16700,16737,16774,16811,16848,  
78. 16885,16922,16959,16996,17033,17070,17107,17143,17180,17217,  
79. 17254,17291,17328,17365,17402,17438,17475,17512,17549,17586,  
80. 17622,17659,17696,17733,17769,17806,17843,17879,17916,17953,  
81. 17989,18026,18063,18099,18136,18172,18209,18246,18282,18319,  
82. 18355,18392,18428,18465,18501,18538,18574,18611,18647,18684,  
83. 18720,18756,18793,18829,18866,18902,18938,18975,19011,19047,  
84. 19084,19120,19156,19192,19229,19265,19301,19337,19374,19410,  
85. 19446,19482,19518,19555,19591,19627,19663,19699,19735,19771,  
86. 19807,19843,19879,19915,19951,19987,20023,20059,20095,20131,  
87. 20167,20203,20239,20275,20311,20347,20383,20419,20455,20490,  
88. 20526,20562,20598,20634,20670,20705,20741,20777,20813,20848,  
89. 20884,20920,20956,20991,21027,21063,21098,21134,21170,21205,  
90. 21241,21276,21312,21348,21383,21419,21454,21490,21525,21561,  
91. 21596,21632,21667,21703,21738,21774,21809,21844,21880,21915,  
92. 21951,21986,22021,22057,22092,22127,22163,22198,22233,22268,  
93. 22304,22339,22374,22409,22445,22480,22515,22550,22585,22621,  
94. 22656,22691,22726,22761,22796,22831,22866,22902,22937,22972,  
95. 23007,23042,23077,23112,23147,23182,23217,23252,23287,23321,  
96. 23356,23391,23426,23461,23496,23531,23566,23600,23635,23670,  
97. 23705,23740,23774,23809,23844,23879,23913,23948,23983,24018,  
98. 24052,24087,24122,24156,24191,24226,24260,24295,24329,24364,  
99. 24399,24433,24468,24502,24537,24571,24606,24640,24675,24709,  
100. 24744,24778,24813,24847,24881,24916,24950,24985,25019,25053,  
101. 25088,25122,25156,25191,25225,25259,25293,25328,25362,25396,  
102. 25430,25465,25499,25533,25567,25601,25635,25670,25704,25738,  
103. 25772,25806,25840,25874,25908,25942,25976,26010,26044,26078,  
104. 26112,26146,26180,26214,26248,26282,26316,26350,26384,26418,  
105. 26452,26486,26520,26553,26587,26621,26655,26689,26722,26756,  
106. 26790,26824,26857,26891,26925,26959,26992,27026,27060,27093,  
107. 27127,27161,27194,27228,27261,27295,27329,27362,27396,27429,  
108. 27463,27496,27530,27563,27597,27630,27664,27697,27731,27764,  
109. 27798,27831,27864,27898,27931,27964,27998,28031,28064,28098,  
110. 28131,28164,28198,28231,28264,28297,28331,28364,28397,28430,  
111. 28463,28497,28530,28563,28596,28629,28662,28695,28729,28762,  
112. 28795,28828,28861,28894,28927,28960,28993,29026,29059,29092,  
113. 29125,29158,29191,29224,29256,29289,29322,29355,29388,29421,

```
114. 29454,29486,29519,29552,29585,29618,29650,29683,29716,29749,
115. 29781,29814,29847,29880,29912,29945,29978,30010,30043,30075,
116. 30108,30141,30173,30206,30238,30271,30303,30336,30369,30401,
117. 30434,30466,30498,30531,30563,30596,30628,30661,30693,30725,
118. 30758,30790,30823,30855,30887,30920,30952,30984,31016,31049,
119. 31081,31113,31145,31178,31210,31242,31274,31306,31339,31371,
120. 31403,31435,31467,31499,31531,31564,31596,31628,31660,31692,
121. 31724,31756,31788,31820,31852,31884,31916,31948,31980,32012,
122. 32043,32075,32107,32139,32171,32203,32235,32267,32298,32330,
123. 32362,32394,32426,32457,32489,32521,32553,32584,32616,32648,
124. 32679,32711,32743,32774,32806,32838,32869,32901,32932,32964,
125. 32996,33027,33059,33090,33122,33153,33185,33216,33248,33279,
126. 33311,33342,33374,33405,33436,33468,33499,33531,33562,33593,
127. 33625,33656,33687,33718,33750,33781,33812,33844,33875,33906,
128. 33937,33969,34000,34031,34062,34093,34124,34156,34187,34218,
129. 34249,34280,34311,34342,34373,34404,34435,34466,34497,34528,
130. 34559,34590,34621,34652,34683,34714,34745,34776,34807,34838,
131. 34869,34899,34930,34961,34992,35023,35054,35084,35115,35146,
132. 35177,35208,35238,35269,35300,35330,35361,35392,35422,35453,
133. 35484,35514,35545,35576,35606,35637,35667,35698,35728,35759,
134. 35790,35820,35851,35881,35912,35942,35972,36003,36033,36064,
135. 36094,36125,36155,36185,36216,36246,36276,36307,36337,36367,
136. 36398,36428,36458,36489,36519,36549,36579,36610,36640,36670,
137. 36700,36730,36760,36791,36821,36851,36881,36911,36941,36971,
138. 37001,37031,37061,37091,37121,37151,37181,37211,37241,37271,
139. 37301,37331,37361,37391,37421,37451,37481,37511,37541,37570,
140. 37600,37630,37660,37690,37719,37749,37779,37809,37839,37868,
141. 37898,37928,37957,37987,38017,38046,38076,38106,38135,38165,
142. 38195,38224,38254,38283,38313,38342,38372,38401,38431,38460,
143. 38490,38519,38549,38578,38608,38637,38667,38696,38725,38755,
144. 38784,38814,38843,38872,38902,38931,38960,38990,39019,39048
145. };
146. // -----
147. unsigned int8 trama[256];
148. unsigned int8 cnt;
149. // -----
150. #int_rda //Interrupción de recepción UART (rda)
151. void isr_rda(void)
152. {
153.     while(bit_test(U1STA,0))
154.     {
155.         trama[cnt] = U1RXREG;
156.         cnt++;
157.     }
158. }
159. // -----
160. #int_EXT1
161. void EXT1_isr(void)
162. {
163.     unsigned int8 s2, s1, s0;
164.     unsigned int16 i, aprx;
```

```
165. int32 value=0;
166. float valorV=0.0;
167. float valorR=0.0;
168. float valorT=0.0;
169. float FSR, a, b, c;
170. output_low(CS_ADS);
171. delay_us(10);
172. spi_write(0xFF);
173. s2 = SPI1BUF;
174. spi_write(0xFF);
175. s1 = SPI1BUF;
176. spi_write(0xFF);
177. s0 = SPI1BUF;
178. output_high(CS_ADS);
179. delay_us(10);
180. value= make32(s2,s1,s0,0); // Valor recibido
181. FSR = 0.512; // FSR= Vref / Ganancia
182. valorV = (float)(value/256) *FSR/ 8388608.0; // Voltios en bornes RTD
183. valorR = valorV / 0.0005; // Resistencia con IDAC1 = 0.5 mA
184. aprx=(unsigned int16) (valorR*100.0);
185. for(i=0; i<1050; i++)
186. {
187.     if((R[i]<aprx)&&(aprx<R[i+1]))
188.     {
189.         a=(float)(-200+i);
190.         b=(float)( R[i+1] - R[i]);
191.         c=(float)( aprx - R[i]);
192.         valorT= a + (c/b); //Temperatura a partir de la tabla
193.         fprintf(TTL, "%.02f\n\r",ValorT);
194.         i=1051;
195.     }
196.     if(R[i]==aprx)
197.     {
198.         a=(float)(-200+i);
199.         valorT= a; //Temperatura a partir de la tabla
200.         fprintf(TTL, "%.02f\n\r",ValorT);
201.         i=1051;
202.     }
203. }
204. }
205. // -----
206.
207. void main()
208. {
209.     output_low( STRT);
210.     output_high(CS_ADS);
211.     output_low( AD_RESET); //Reset del ADS1247
212.     delay_ms(10);
213.     output_high(AD_RESET);
214.     delay_ms(500);
215.     output_high( STRT );
```

```

216.  setup_spi ( SPI_MASTER | SPI_SS_DISABLED | SPI_L_TO_H | SPI_XMIT_H_TO_L |
        SPI_CLK_DIV_16);           // Configuración del SPI
217.  ADS_cero();                     // Preparación del ADS para la calibración de Offset
218.  ADS_Offset_calibration();       // Calibración del Offset
219.  ADS_full();                     // Preparación del ADS para la calibración de la ganancia
220.  ADS_Gain_calibration();         // Calibración de la ganancia
221.  ADS_init();                     // Configuración del ADS
222.  ADS_rd_register();
223.  ext_int_edge(1,H_TO_L);        // Activación de la interrupción con cambio de alto a bajo
224.  clear_interrupt(INT_EXT1);      // Limpiar interrupción del ADS
225.  enable_interrupts(INT_EXT1);    // Interrupción para el ADS habilitada
226.  output_low( CS_ADS );
227.  delay_us(100);
228.  spi_write(RDATACT);             // Modo de lectura continua de datos
229.  output_high(CS_ADS );
230.  delay_us(100);
231.  for(;;)                          // Superbucle infinito
232.  {
233.  }
234.  }
235.
236.  // -----

```

## 1.2 Modificaciones para el resto de las configuraciones

En primer lugar, el valor de la variable FSR (definida en la línea 181) empleado en la conversión del dato recibido por el ADS1247 varía según la configuración empleada al ser el voltaje de referencia dividido entre la ganancia del amplificador programable. En la siguiente tabla se puede ver el valor concreto para cada caso, siempre empleándose con el valor positivo:

Configuración	V <sub>REF</sub>	Ganancia	FSR
2 hilos V <sub>REF</sub>	2,048 V	4	± 0.512 V
3 hilos V <sub>REF</sub>	2,048 V	8	± 0.256 V
3 hilos R <sub>REF</sub>	1,33 V	4	± 0.3325 V
3 hilos V <sub>REF</sub> R <sub>COMP</sub>	2,048 V	8	± 0.256 V
3 hilos R <sub>REF</sub> R <sub>COMP</sub>	1,33 V	4	± 0.3325 V
4 hilos V <sub>REF</sub>	2,048 V	4	± 0.512 V
4 hilos R <sub>REF</sub>	0,665 V	2	± 0.3325 V

En segundo lugar, para los casos en los que se usa resistencia de compensación es necesario cambiar la ecuación de la línea 183 por la siguiente:

$$\text{valorR} = (\text{valorV} / 0.0005) + 50;$$

## 2 ADS1247.h

### 2.1 Configuración 4 hilos V<sub>REF</sub>

```
156. #ifndef INC_ADS1247
157.
158. #define INC_ADS1247 1
159. //-----
160. //ADS1247 Registros
161. #define MUX0    0x00    //BCS[1:0] MUX_SP[2:0] MUX_SN[2:0]
162. #define VBIAS  0x01    //VBIAS[7:0]
163. #define MUX1   0x02    //CLKSTAT VREFCON[1:0] REFSELT[1:0] MUXCAL[2:0]
164. #define SYS0   0x03    //0 PGA[2:0] DR[3:0]
165. #define OFC0   0x04    //OFC[7:0]
166. #define OFC1   0x05    //OFC[15:8]
167. #define OFC2   0x06    //OFC[23:16]
168. #define FSC0   0x07    //FSC[7:0]
169. #define FSC1   0x08    //FSC[15:8]
170. #define FSC2   0x09    //FSC[23:16]
171. #define IDAC0  0x0A    //ID[3:0] DRDY_MODE IMAG[2:0]
172. #define IDAC1  0x0B    //I1DIR[3:0] I2DIR[3:0]
173. #define GPIOCFG 0x0C    //IOCFG[7:0]
174. #define GPIODIR 0x0D    //IODIR[7:0]
175. #define GPIODAT 0x0E    //IODAT[7:0]
176. //-----
177. //SPI ADS1247 Comandos
178. #define WAKEUP 0x00    //Salir del modo bajo consumo
179. #define SLP    0x02    //Entrar modo bajo consumo
180. #define SYNC   0x04    //Sincronizar conversiones del ADC
181. #define RESET  0x06    //Volver a los valores por defecto
182. #define NOP    0xFF    //No operación
183. #define RDATA  0x12    //Leer datos una vez
184. #define RDATACT 0x14    //Modo de lectura continua de datos
185. #define SDATACT 0x16    //Parar de leer datos continuamente
186. #define RREG0  0x20    //Leer del registro rrrr 0010 rrrr
187. #define WREG0  0x40    //Escribir en el registro rrrr 0100 rrrr
188. #define SYSOCAL 0x60    //Calibración del Offset
189. #define SYSGCAL 0x61    //Calibración de la ganancia
190. #define SELFOCAL 0x62    //Calibración del Offset propia
191. //-----
192.
193. void ADS_init(void)
194. {
195.     output_low( CS_ADS);
196.     spi_write(SDATACT); // Parar de leer datos continuamente
197.     spi_write(WREG0);   // Escribir en 3 registros empezando en la dirección 00h
198.     spi_write(3);
199.     spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
    // ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
200.     spi_write(0b00000000); // Sin voltaje de polarización
```

```
201. spi_write(0b01110000); // Oscilador interno / La referencia interna está activada con
    una conversión está en progreso / Usando referencia interna Vref / Operación normal
202. spi_write(0b00100000); // PGA 4 / 5 SPS
203. spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
204. spi_write(1);
205. spi_write(0b00000100); // Corrientes de 500ma
206. spi_write(0b00001111); // IDAC 1= AIN0 IDAC 2= ninguno
207. delay_us(100);
208. output_high(CS_ADS);
209. delay_ms(2);
210. }
211.
212. // -----
213.
214. void ADS_cero(void)
215. {
216. output_low( CS_ADS);
217. spi_write(SDATACT); // Parar de leer datos continuamente
218. spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
219. spi_write(2);
220. spi_write(0b00001010); // Apagado de las corrientes "Burn-
    out" de comprobación /
    ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
221. spi_write(0b00000000); // Sin voltaje de polarización
222. spi_write(0b00110001); // Oscilador interno / La referencia interna está activa /
    Usando referencia interna Vref / Calibración del Offset
223. delay_us(100);
224. output_high(CS_ADS);
225. delay_ms(2);
226. }
227.
228. // -----
229.
230. void ADS_full(void)
231. {
232. output_low( CS_ADS);
233. spi_write(SDATACT); // Parar de leer datos continuamente
234. spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
235. spi_write(2);
236. spi_write(0b00001010); // Apagado de las corrientes "Burn-
    out" de comprobación /
    ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
237. spi_write(0b00000000); // Sin voltaje de polarización
238. spi_write(0b00110010); // Oscilador interno / La referencia interna está activa /
    Usando referencia interna Vref / Calibración de la ganancia
239. delay_us(100);
240. output_high(CS_ADS);
241. delay_ms(2);
242. }
243.
244. // -----
```

```
245.
246. void ADS_rd_register(void)
247. {
248.     unsigned int8 B0,B1,B2;
249.     output_low( CS_ADS);
250.     spi_write(SDATACT);
251.     spi_write(RREG0);
252.     spi_write(14);
253.     spi_write(0xFF);
254.     B0=SPI1BUF;
255.     spi_write(0xFF);
256.     B0=SPI1BUF;
257.     spi_write(0xFF);
258.     B0=SPI1BUF;
259.     spi_write(0xFF);
260.     B0=SPI1BUF;
261.     spi_write(0xFF);
262.     B0=SPI1BUF;
263.     spi_write(0xFF);
264.     B1=SPI1BUF;
265.     spi_write(0xFF);
266.     B2=SPI1BUF;
267.     spi_write(0xFF);
268.     B0=SPI1BUF;
269.     spi_write(0xFF);
270.     B1=SPI1BUF;
271.     spi_write(0xFF);
272.     B2=SPI1BUF;
273.     spi_write(0xFF);
274.     B0=SPI1BUF;
275.     spi_write(0xFF);
276.     B0=SPI1BUF;
277.     spi_write(0xFF);
278.     B0=SPI1BUF;
279.     spi_write(0xFF);
280.     B0=SPI1BUF;
281.     spi_write(0xFF);
282.     B0=SPI1BUF;
283.     delay_us(10);
284.     output_high(CS_ADS);
285.     delay_ms(2);
286. }
287. // -----
288. void ADS_Offset_calibration(void)
289. {
290.     output_low( CS_ADS);
291.     spi_write(SDATACT);
292.     delay_us(10);
293.     spi_write(SYSOCAL);
294.     delay_us(10);
295.     delay_ms(3500);
```

```
296. output_high(CS_ADS);
297. delay_ms(2);
298. }
299. //-----
300. void ADS_Gain_calibration(void)
301. {
302. output_low( CS_ADS);
303. spi_write(SDATACT);
304. delay_us(10);
305. spi_write(SYSGCAL);
306. delay_us(10);
307. delay_ms(3500);
308. output_high(CS_ADS);
309. delay_ms(2);
310. }
311. //-----
312. #endif
```

## 2.2 Modificaciones para el resto de las configuraciones

Lo único que varía entre las distintas configuraciones es la función que configura el ADS1247 llamada ADS\_init. A continuación, están todas las distintas funciones según la configuración que se emplee. La configuración de 2 hilos V<sub>REF</sub> comparte el código con la configuración 4 hilos V<sub>REF</sub>. A su vez cada una de las versiones de la configuración de 3 hilos comparten también la función ADS\_init tanto si se usa resistencia de compensación como si no se usa.

### 2.2.1 Función ADS\_init para la configuración 4 hilos R<sub>REF</sub>

```
56. void ADS_init(void)
57. {
58. output_low( CS_ADS);
59. spi_write(SDATACT); // Parar de leer datos continuamente
60. spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
61. spi_write(3);
62. spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
// ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
63. spi_write(0b00000000); // Sin voltaje de polarización
64. spi_write(0b01100000); // Oscilador interno / La referencia interna está activada con
una conversión está en progreso / Usando referencia externa Rref / Operación normal
65. spi_write(0b00010000); // PGA 2 / 5 SPS
66. spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
67. spi_write(1);
68. spi_write(0b00000100); // Corrientes de 500ma
69. spi_write(0b00001111); // IDAC 1= AINO IDAC 2= ninguno
70. delay_us(100);
71. output_high(CS_ADS);
72. delay_ms(2);
73. }
```

### 2.2.1 Función ADS\_init para la configuración 3 hilos V<sub>REF</sub> (con y sin R<sub>COMP</sub>)

```
56. void ADS_init(void)
57. {
58.   output_low( CS_ADS);
59.   spi_write(SDATACT); // Parar de leer datos continuamente
60.   spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
61.   spi_write(3);
62.   spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
   / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
63.   spi_write(0b00000000); // Sin voltaje de polarización
64.   spi_write(0b01110000); // Oscilador interno / La referencia interna está activada con
   una conversión está en progreso / Usando referencia interna Vref / Operación normal
65.   spi_write(0b00110000); // PGA 8 / 5 SPS
66.   spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
67.   spi_write(1);
68.   spi_write(0b00000100); // Corrientes de 500ma
69.   spi_write(0b00000011); // IDAC 1= AIN0 IDAC 2= AIN3
70.   delay_us(100);
71.   output_high(CS_ADS);
72.   delay_ms(2);
73. }
```

### 2.2.1 Función ADS\_init para la configuración 3 hilos R<sub>REF</sub> (con y sin R<sub>COMP</sub>)

```
56. void ADS_init(void)
57. {
58.   output_low( CS_ADS);
59.   spi_write(SDATACT); // Parar de leer datos continuamente
60.   spi_write(WREG0); // Escribir en 3 registros empezando en la dirección 00h
61.   spi_write(3);
62.   spi_write(0b00001010); // Apagado de las corrientes "Burn-out" de comprobación
   / ADC Entrada positiva = AIN1 / ADC Entrada negativa = AIN2
63.   spi_write(0b00000000); // Sin voltaje de polarización
64.   spi_write(0b01110000); // Oscilador interno / La referencia interna está activada con
   una conversión está en progreso / Usando referencia interna Rref / Operación normal
65.   spi_write(0b00100000); // PGA 4 / 5 SPS
66.   spi_write(WREG0+0x0A); // Escribir en 2 registros empezando en la dirección 0Ah
67.   spi_write(1);
68.   spi_write(0b00000100); // Corrientes de 500ma
69.   spi_write(0b00000011); // IDAC 1= AIN0 IDAC 2= AIN3
70.   delay_us(100);
71.   output_high(CS_ADS);
72.   delay_ms(2);
73. }
```