



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

**GRADO EN INGENIERÍA ELECTRÓNICA
INDUSTRIAL Y AUTOMÁTICA**

TRABAJO FIN DE GRADO

**Control de fuerza con el brazo
robot colaborativo UR3**

Autor: Jose Antonio Ferriz Beneito

Tutor: Dr. Ranko Zotovic Stanisic

ETSID (UPV), Valencia, Julio del 2018

Contenido

-AGRADECIMIENTOS	6
-RESUMEN	7
-ABSTRACT.....	7
-INTRODUCCIÓN.....	8
Marco del proyecto.....	8
Objetivos del proyecto	8
Estructura del documento	8
-ESTADO DEL ARTE	9
Brazos robot colaborativos	9
Sensores de Fuerza/Par.....	11
Control de fuerza del robot.....	13
-MATERIALES Y DISPOSITIVOS.....	14
Robot colaborativo.....	14
Caja de control	15
Consola de programación	16
Sensor fuerza/par.....	17
Adaptador del sensor al robot	20
Herramienta de contacto	21
-MÉTODOS DE PROGRAMACIÓN.....	22
Comunicación sensor-robot directa.....	22
Envío de la lectura del sensor desde el PC.....	22
Control completo del robot desde el PC.....	23
Selección de opción.....	23
-DESARROLLO DE LA PROGRAMACIÓN	24
Lectura del sensor	24
Cálculo de las fuerzas y pares.....	25
Programación en LabVIEW	26
Control mediante Polyscope	29
-1 Aprender el uso de Polyscope.....	30
-2 Control de fuerza con el comando force()	30
- 3 Cliente para comunicación TCP/IP	30
-4 Desarrollo del programa principal	31
Pruebas.....	35
-CONCLUSIONES	38
-PRESUPUESTO	38

-REFERENCIAS BIBLIOGRÁFICAS	39
-PLANOS	39
-Plano del transductor ATI Delta sin adaptador.....	40
-Plano del acople del transductor al robot UR3.....	41
-Plano del acople de la herramienta al sensor Delta	42

-ÍNDICE DE FIGURAS

Figura 1- Robot colaborativo en una fábrica de coches.....	9
Figura 2- Dibujo representativo del área de trabajo de robots y cobots.....	10
Figura 3- Sensores de Fuerza y Torsión multi ejes de la compañía ATI.	12
Figura 4- Esquema del control por bucle Proporcional-Derivativo.....	14
Figura 5- Robot UR3 utilizado en el proyecto	14
Figura 6- Caja de control con la consola colgada	15
Figura 7 – Consola de programación del UR3	16
Figura 8- Sensor ATI Delta	17
Figura 9- Dibujo del transductor del sensor ATI Delta	18
Figura 10- Sensor ATI Delta usado en el proyecto	18
Figura 11- Dibujo del cable del transductor	19
Figura 12- Dibujo de la caja de alimentación	19
Figura 13- Dibujo del cable de la fuente de alimentación	20
Figura 14- Dibujo de la tarjeta de adquisición de datos	20
Figura 15- Adaptador del sensor al UR3 mecanizado	21
Figura 16- Herramienta de contacto mecanizada.....	21
Figura 17- Esquema de la comunicación sensor-robot directa.....	24
Figura 18- Esquema del envío de la lectura desde el PC.....	24
Figura 19- Esquema del control completo desde PC	24
Figura 20- Dibujo representativo de los ejes del transductor.....	24
Figura 21- Cálculos para la conversión de voltaje a fuerzas del sensor	26
Figura 22- Diagrama de bloques de LabVIEW para la lectura del sensor	26
Figura 23- Diagrama de bloques de LabVIEW para servidor TCP/IP	27
Figura 24- Comb. en LabVIEW del programa de lectura del sensor y del servidor TCP/IP	27
Figura 25- Modificación del programa de LabVIEW para sobrescribir el buffer	28
Figura 26- Diagrama de bloques de LabVIEW utilizado en el proyecto	28
Figura 27- Imagen mostrada de LabVIEW en funcionamiento	29
Figura 28- Programa para el control del UR3 en la pantalla de programación	34
Figura 29- Tabla de resultados con fuerza de referencia de 10 Newtons	35
Figura 30- Gráfica de lecturas para $K_p= 0.002$ y $K_d = 0.3$	36
Figura 31- Gráfica de lecturas para $K_p= 0.0025$ y $K_d = 0.4$	36
Figura 32- Gráfica de lecturas para $K_p= 0.003$ y $K_d = 0.5$	36
Figura 33- Tabla con los resultados para fuerzas de referencia de 5 y 15 Newtons.....	37
Figura 34- Tabla del presupuesto en materiales.....	38

-AGRADECIMIENTOS

En primer lugar, agradecer todo el apoyo recibido de mi familia y amigos, sobre todo a mis padres, gracias a ellos he podido llegar a donde estoy ahora. A mi novia Verónica por aguantar mis frustraciones y enfados y aun así ayudarme a seguir adelante.

Como no, agradecer a mi tutor Ranko y mis compañeros Rodrigo y Carlos por estar conmigo durante la realización de este proyecto y su ayuda en la resolución de los problemas que han ido surgiendo.

También agradezco a los alumnos, profesores y técnicos su compañía haciendo más agradables las horas trabajando en el laboratorio y echando una mano cuando fuese necesario.

Por último, agradecer a la Universidad Politécnica de Valencia y la Escuela Técnica Superior de Ingeniería del Diseño por permitirme estudiar en su centro y adquirir los conocimientos y capacidades que he desarrollado durante estos años.

-RESUMEN

Este documento contiene la memoria del Trabajo Fin de Grado titulado “Control de fuerza con el brazo robot colaborativo UR3” escrito por Jose Antonio Ferriz Beneito.

Los objetivos a lograr durante la realización de este proyecto son:

- Comprender la mecánica del robot colaborativo y su programación para conseguir un conocimiento avanzado de su funcionamiento.
- Entender el funcionamiento de los sensores de fuerza/par y sus distintas aplicaciones.
- Desarrollar una comunicación entre un ordenador y un robot.
- Crear las bases para desarrollar un control por fuerza de un robot colaborativo.

-ABSTRACT

This document contains the memory of the Final Degree Project entitled “Control de fuerza con el brazo robot colaborativo UR3” written by Jose Antonio Ferriz Beneito.

The objectives to achieve during the realization of this project are:

- Understand the mechanics of the collaborative robot and its programming to get an advanced knowledge of its operation.
- Develop a communication between a computer and a robot.
- Create the bases to develop a force control of a collaborative robot.

-INTRODUCCIÓN

Marco del proyecto

Este Trabajo Fin de Grado viene dado de que el robot colaborativo UR3 no dispone de un control por fuerza preciso ya que éste calcula las fuerzas generadas a partir de unos cálculos realizados con los datos obtenidos en los ejes de los motores. Gracias a la incorporación de un sensor de Fuerza y Par en la herramienta permitirá que este robot colaborativo reciba unos datos más precisos y permita un control de fuerza con mayor precisión.

Objetivos del proyecto

La idea principal del proyecto es la incorporación de un sensor de fuerza en un robot colaborativo, la creación de una comunicación entre ambos y el uso del sensor para permitir al robot una lectura más precisa de las fuerzas que genera ampliando así las aplicaciones útiles de dicho robot. Todo esto permitirá realizar un control de fuerza del robot mejor que el implementado por el fabricante.

Estructura del documento

Para la redacción de este documento se ha seguido una organización típica de una memoria de TFG:

Primero se ha realizado un resumen del Trabajo Fin de Grado, los objetivos a cumplir y su desarrollo.

Después se han descrito los elementos utilizados y su uso en este proyecto.

A continuación, se han propuesto diferentes métodos para su realización y la elección del que se va a utilizar.

Más tarde se procede al desarrollo del trabajo.

Y por último se realiza una conclusión seguida de un presupuesto, la bibliografía, los anexos, pliego de condiciones y planos.

-ESTADO DEL ARTE

Para comprender mejor el trabajo realizado es importante conocer el funcionamiento de los brazos robot colaborativos y los sensores de fuerza y par ya que se trata de los dispositivos utilizados en este proyecto.

También conocer el control proporcional y derivativo que se aplicará para el control por fuerza.

Brazos robot colaborativos

Los robots colaborativos también llamados cobots son una herramienta muy útil para la pequeña y mediana industria, fáciles de programar y libres de complicados sistemas de seguridad que permiten la interacción entre los robots y las personas.



Figura 1- Robot colaborativo en una fábrica de coches

El uso de estos robots colaborativos reduce los riesgos para los operarios y los costes de implantar sistemas de seguridad a su alrededor optimizando el espacio en las líneas de montaje y aumentando la producción.

Estos nuevos robots permiten que la automatización robótica de las fábricas sea más accesible, permitiendo una mayor flexibilidad en la automatización de diversas tareas que se desarrollan en la industria.

Los robots colaborativos están preparados para ser programados de una manera sencilla, por personal no cualificado y sin necesidad de conocimiento de programación. Más que programación del robot se suele hablar de enseñar o entrenar al robot ya que los movimientos se pueden programar moviendo manualmente el robot por su muñeca a las posiciones deseadas y registrando estas posiciones para que luego las repita de forma automática.

Esto permite que los propios empleados de la fábrica sean capaces de recibir, instalar, programar y poner en marcha una instalación robótica. Esta facilidad de instalación y programación permite que el robot pueda ser utilizado en diferentes tareas dentro de la línea de producción de una manera flexible.

Los robots disponen de sensores de fuerza y consumo pudiendo detectar colisiones con su entorno permitiéndoles la posibilidad de desconectar sus sistemas evitando dañar a los operarios en caso de contacto.

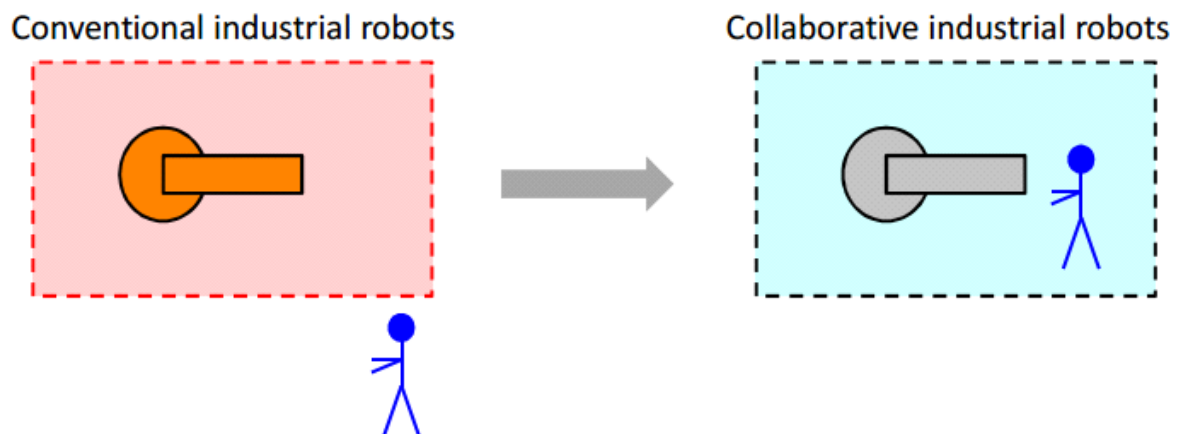


Figura 2- Dibujo representativo del área de trabajo de robots y cobots

Para que los robots colaborativos trabajen cerca de las personas tienen ciertas limitaciones como su limitación de carga a manipular y la velocidad de los movimientos, pero a cambio tienen muchas otras ventajas que los hacen ser una opción muy interesante y que cada vez se utilizan en más industrias introduciéndolos en sus líneas de montaje.

Sensores de Fuerza/Par

Un sensor es un dispositivo formado por células sensibles que es capaz de detectar variaciones en una magnitud física o química de su entorno, convirtiendo esos estímulos o acciones en señales útiles para un sistema de medida o control, es decir, en señales transducibles.

Generalmente los sensores no producen señales eléctricas y por ello se asocian con transductores. Los transductores son dispositivos que reciben señales físicas y generalmente las convierten en señales eléctricas. Las señales eléctricas nos permiten tratar la información del sensor de una manera más simple.

Por lo general transductor y sensor se usan como sinónimos aun no siendo exactamente lo mismo.

Un método utilizado en la medición de fuerzas consiste en medir el efecto que provoca la fuerza sobre un elemento elástico, denominado célula de carga, cuyas dimensiones se ven alteradas. Si la célula presenta una forma apropiada las variaciones de dimensión pueden ser proporcionales al efecto que las provoca.

Dentro de los diferentes tipos de sensores de fuerza se destacan algunos a continuación:

- Capacitivos. Se basan en las variaciones de la capacitancia debidas a un estímulo externo.
- A Reluctancia. Se basan en una modificación de la reluctancia magnética, es decir, la resistencia al paso de flujo magnético de un material al incidir sobre él un campo magnético.
- Galgas extensiométricas. Basados en materiales que cambian su resistencia eléctrica al sufrir un esfuerzo.
- Piezoeléctricos. Generan una diferencia de potencial en su superficie al aplicarles tensiones mecánicas.
- Elemento vibrante. Se basan en la vibración de la frecuencia de oscilación producida al aplicar una fuerza.

Centrándonos en los actuales sensores de fuerza-par desarrollados comentaremos algunos de estos sensores comerciales.

La empresa ATI ha desarrollado un sensor de fuerza-par de 6 ejes, en otras palabras, la información que nos proporciona son las coordenadas en los 3 ejes cartesianos de la fuerza y del par que se ejercen sobre el sensor. Su principal utilidad es conocer la carga que soporta el efector final de un brazo robótico. Las señales de lectura pasan a un transductor que utiliza calibradores de tensión para detectar las fuerzas,

las cuales también proporcionan alta inmunidad al ruido y permiten alta protección sobre la carga. El controlador que va conjunto a este sensor permite una comunicación serie de datos con un ordenador u otro dispositivo. Otras características que posee este dispositivo es que nos permite modificar el sistema de referencia que toma para las medidas, como también nos permite aislar la carga que se proporcionaría por el peso de la herramienta. ATI también permite la personalización de su sensor.



Figura 3- Sensores de Fuerza y Torsión multi ejes de la compañía ATI.

Este tipo de dispositivo se usa principalmente en robótica en aplicaciones de manipulación de objetos, operaciones de mecanizado y también se usa en el campo de la medicina. Nos permite conocer los esfuerzos que sufre un efector final en todo momento, pudiendo llevar a cabo una estrategia de realimentación que consiga reducir esos esfuerzos antes de que exista un daño en la estructura del sistema.

Control de fuerza del robot

Existen diferentes métodos para realizar el control de fuerza en un robot. Como la fuerza aplicada debe ser constante, para este proyecto se utiliza el control por bucle interior/exterior.

Para aplicar una fuerza se ordena al robot una velocidad objetivo a alcanzar, cuanto mayor sea esta velocidad mayor será la fuerza aplicada por el robot.

Este tipo de control permite ajustar la fuerza ejercida respecto a un valor de referencia mediante la realimentación de la fuerza de salida. Al comparar la fuerza de referencia con la actual genera el error actual que se multiplica por una constante proporcional para crear una respuesta que reduzca este error.

Esta respuesta obtenida es la velocidad que se desea que el robot alcance, pero para evitar oscilaciones, en la señal de salida se dispone de un bucle interno de la velocidad. En este bucle se coge la velocidad del instante anterior y se multiplica por una constante derivativa lo que permite que las posibles oscilaciones en el sistema se vean amortiguados.

En la siguiente imagen se puede observar de manera esquematizada el funcionamiento de este sistema de control.

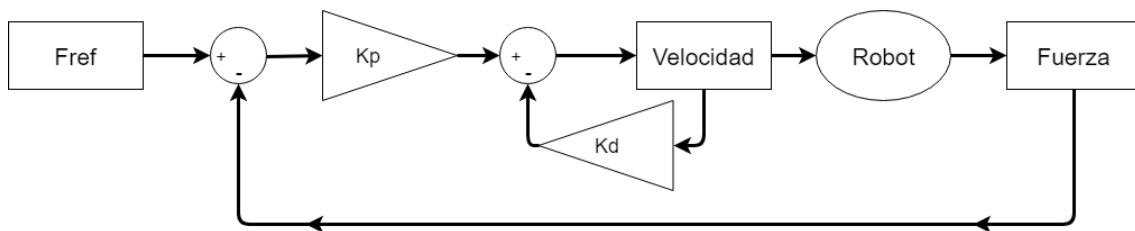


Figura 4- Esquema del control por bucle Proporcional-Derivativo

-MATERIALES Y DISPOSITIVOS

En este apartado se realizará una breve descripción de las principales características de los elementos utilizados en el proyecto.

Robot colaborativo

Primero se comentarán las características del brazo robótico colaborativo usado, su caja de control y la consola de programación:

El modelo que se ha utilizado es el brazo robot UR3 perteneciente a la empresa Universal Robots y se encuentra en el edificio del DISA en el laboratorio de robótica que se puede observar en la siguiente imagen.

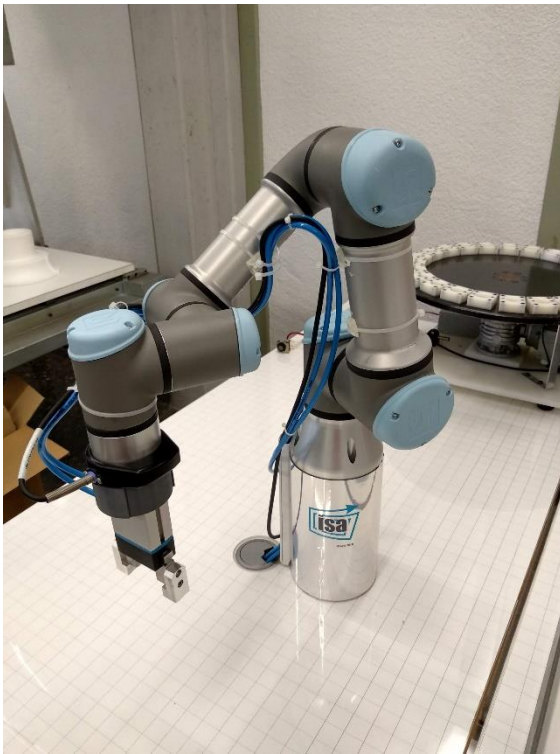


Figura 5- Robot UR3 utilizado en el proyecto

El robot tiene un peso de unos 11 Kilogramos fabricado en aluminio y plásticos de PP. Puede trabajar con una carga útil de hasta 3 Kilogramos con un alcance de 500 milímetros.

Dispone de 6 articulaciones giratorias permitiendo 6 grados de libertad:

- La base con 360º de acción y una velocidad máxima de 180º/s.
- El hombro con 360º de acción y una velocidad máxima de 180º/s.
- El codo con 360º de acción y una velocidad máxima de 180º/s.

- La muñeca 1 con 360º de acción y una velocidad máxima de 360º/s.
- La muñeca 2 con 360º de acción y una velocidad máxima de 360º/s.
- La muñeca 3 con radio de acción infinito y una velocidad máxima de 360º/s.

Estas características permiten una velocidad máxima en la herramienta de 1m/s.

Caja de control

La caja de control contiene los puertos e interfaces de comunicación de los que dispone el robot permitiendo su programación y comunicación con dispositivos externos.



Figura 6- Caja de control con la consola colgada

Tiene un peso de 15 Kg fabricado en acero con dimensiones 475 x 423 x 268 mm. Está alimentado por una fuente de 100-240 V por corriente alterna a 50-60 Hz.

Los puertos disponibles son los siguientes:

16 entradas digitales

16 salidas digitales

2 entradas analógicas

2 salidas analógicas

Fuente de alimentación para E/S de 24 Voltios 2 Amperios

También dispone de comunicación mediante:

TCP/IP 100 Mbit

Modbus TCP

Profinet

EthernetIP

Consola de programación

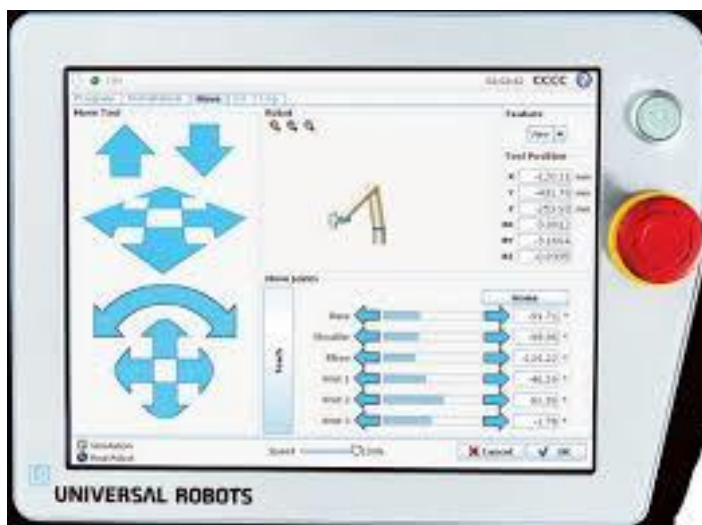


Figura 7 – Consola de programación del UR3

Se trata de una pantalla táctil de 12” portátil con la interfaz gráfica del usuario PolyScope. Esta interfaz permite realizar una programación sencilla y rápida del robot de una manera intuitiva y cómoda muy sencilla de aprender a utilizar. El funcionamiento de esta interfaz está explicado más adelante.

Sensor fuerza/par

Para la utilización de un control por fuerza en el robot se empleará el sensor F/T: Delta, fabricado por ATI Industrial Automation. Ha sido proporcionado por el IRF.



Figura 8- Sensor ATI Delta

Este transductor de 6 ejes permite la medida de las fuerzas y pares aplicados por la herramienta.

El sensor está formado por el transductor F/T, el cable que conecta el transductor con la caja de alimentación, la caja de alimentación, el cable de alimentación que conecta con la DAQ y una tarjeta de adquisición de datos (DAQ) para enviar la información a un ordenador.

El transductor es una estructura robusta y compacta que realiza la conversión de las fuerzas y pares aplicados en señales analógicas de medida de esfuerzo que son enviadas al controlador. El transductor se usa montado entre un robot y la herramienta de dicho robot. El modelo Delta utilizado en este proyecto dispone de una interfaz electrónica integrada que requiere ser conectada a una fuente de alimentación. Las partes que lo conforman están mecanizadas a partir de una pieza de metal sólida, consiguiendo de esta manera un decrecimiento de la histéresis y aumentando la dureza y repetibilidad de la estructura. Suelen ubicarse en la muñeca del robot unidos a un efector, para poder medir los esfuerzos que soporta el actuador. Presenta protecciones contra sobrecarga para evitar posibles daños accidentales. Para realizar las medidas de fuerza y

par, el transductor dispone de 6 galgas extensiométricas que detectan las pequeñas deformaciones elásticas del metal que se traducen en variaciones de voltaje. Estas señales son enviadas a través del cable del transductor.

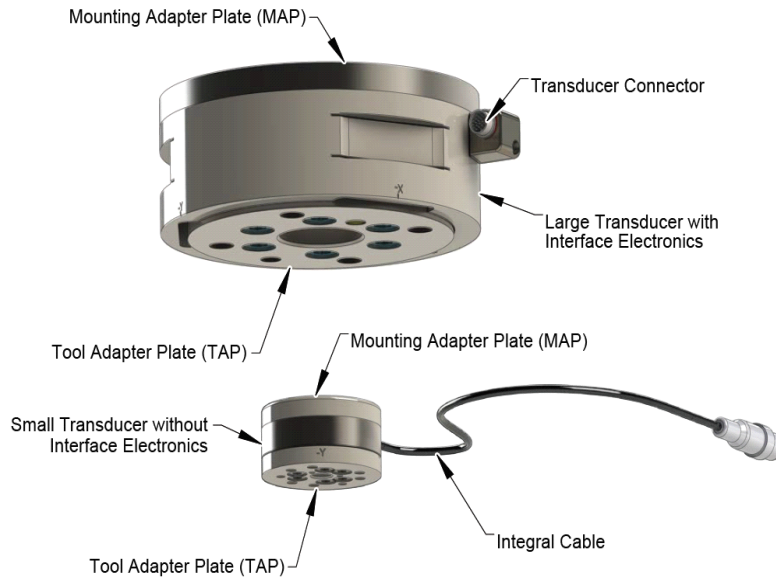


Figura 9- Dibujo del transductor del sensor ATI Delta



Figura 10- Sensor ATI Delta usado en el proyecto

El cable del transductor con alta flexibilidad está protegido eléctricamente para evitar interferencias en la transmisión desde el transductor a la caja de suministro de alimentación.



Figura 11- Dibujo del cable del transductor

La caja de alimentación (PS) es necesaria para alimentar los componentes electrónicos de la interfaz incorporada en el transductor. Para conectar con el transductor, la fuente de alimentación está equipada con un conector hembra de 20 pines para el cable del transductor. Para proporcionar energía la caja de alimentación se puede conectar a una fuente de alimentación de 12 voltios montada en la pared o directamente desde una fuente de 5 voltios a través del cable de alimentación del dispositivo DAQ. Un conector macho de 26 pines proporciona la interfaz del dispositivo DAQ. Si ambas fuentes están conectadas, la caja PS usará la fuente de 12 voltios ignorando la de 5 voltios.

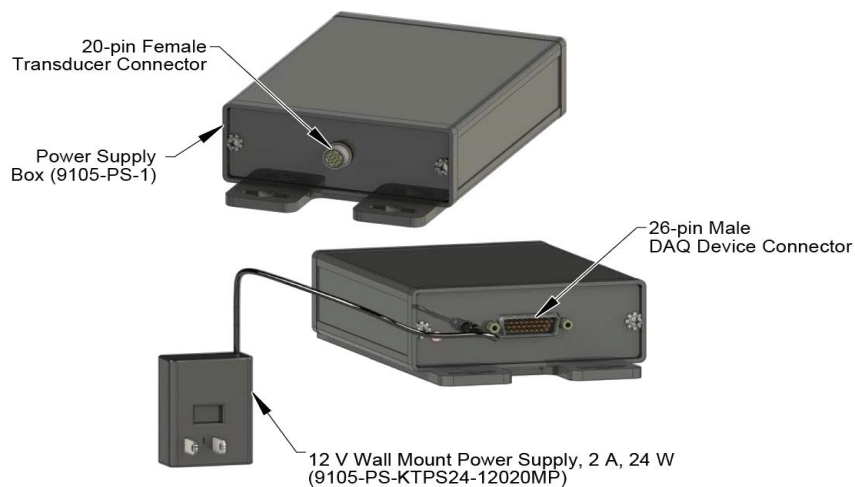


Figura 12- Dibujo de la caja de alimentación

El cable de la fuente de alimentación conecta la caja de la fuente de alimentación con la tarjeta de adquisición de datos DAQ. Este cable generalmente tiene un conector en el extremo para conectar con la DAQ pero también está disponible sin terminar.

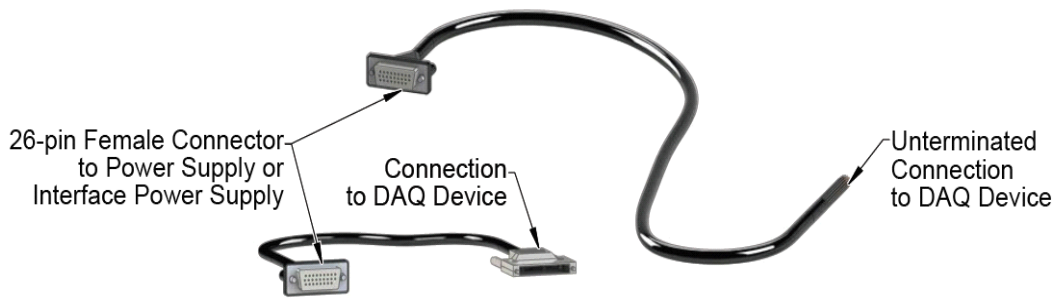


Figura 13- Dibujo del cable de la fuente de alimentación

La tarjeta de adquisición de datos DAQ convierte las señales del transductor de voltajes analógicos a datos que un ordenador pueda procesar. Usando el software ATI, los datos del transductor se convierten en valores de fuerza y par. El sistema de adquisición de datos también proporciona energía al sistema del transductor. La DAQ que se utiliza en este proyecto está instalada en el ordenador.

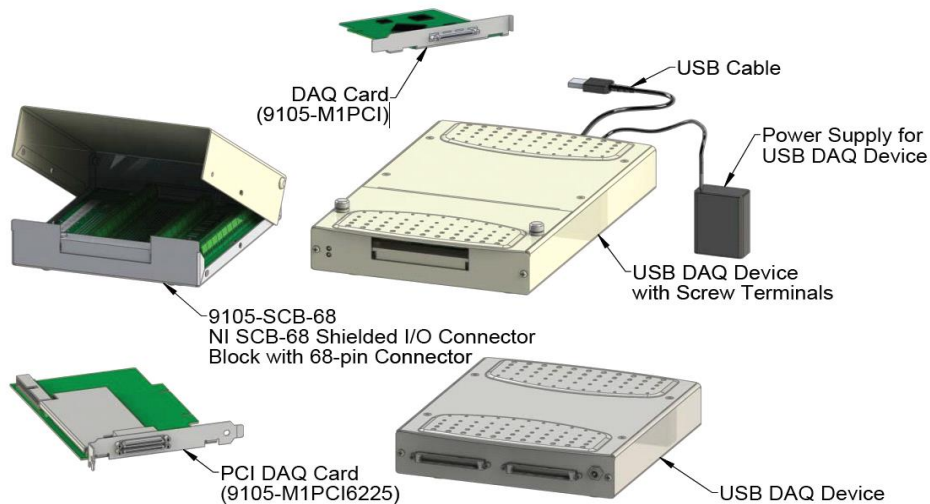


Figura 14- Dibujo de la tarjeta de adquisición de datos

Adaptador del sensor al robot

El transductor viene con un adaptador de montaje estándar para unir mecánicamente el transductor al brazo robot, pero dado que este adaptador no sirve para el UR3 se procedió a mecanizar uno nuevo utilizando las medidas del sensor y del acople del brazo robot. (Plano disponible al final de la memoria)



Figura 15- Adaptador del sensor al UR3 mecanizado

Herramienta de contacto

Para realizar un correcto contacto entre el sensor y la superficie donde se aplique la fuerza se ha mecanizado una herramienta con dicho fin. Esta herramienta viene en dos partes que se pueden observar en la siguiente imagen. (Plano disponible al final de la memoria)



Figura 16- Herramienta de contacto mecanizada

-MÉTODOS DE PROGRAMACIÓN

Para realizar un control por fuerza del brazo robot UR3 mediante el sensor de fuerza se han estudiado diferentes formas de programación. Las opciones estudiadas son las siguientes:

Comunicación sensor-robot directa

Con esta opción se evita el uso de un ordenador externo. Para ello se procesa la información recibida del sensor mediante convertidores analógico-digitales y un multiplexor, enviando la información directamente al robot.

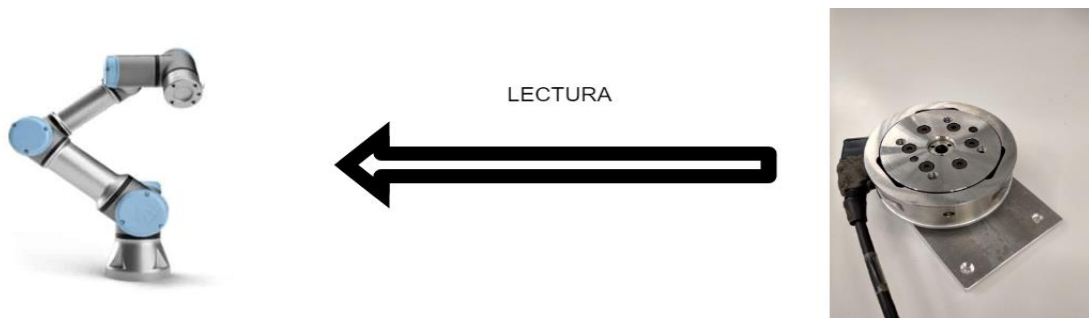


Figura 17- Esquema de la comunicación sensor-robot directa

Envío de la lectura del sensor desde el PC

En esta opción el ordenador recibe los valores de las fuerzas y pares a través de la tarjeta de adquisición de datos, los procesa y envía al brazo robot mediante comunicación TCP/IP. El robot contiene la programación para procesar los datos recibidos y ejecutar el control de fuerza.



Figura 18- Esquema del envío de la lectura desde el PC

Control completo del robot desde el PC

En esta opción la programación se encuentra en el ordenador que recibe la información desde el sensor y se encarga de enviar las ordenes al robot mediante la comunicación TCP/IP.



Figura 19- Esquema del control completo desde PC

Selección de opción

Dependiendo de nuestras prioridades se elegirá una de las opciones. Comparándolas vemos que la opción de comunicación sensor-robot directa ahorra el tener que disponer de un ordenador externo para reenviar la información del sensor al robot, pero debido a que la caja de control no dispone de entradas suficientes se utiliza otra opción. La opción de envío de la lectura del sensor desde el PC permite procesar toda la información en el propio robot siendo una programación más sencilla y por consiguiente con menor posibilidad de fallar. Por último, el control completo del robot desde el PC permite disponer de una velocidad de procesamiento que dependerá de las características del ordenador, pero se deben enviar los comandos de uno en uno lo que no permite realizar un control a tiempo real.

Vistas estas opciones se opta por realizar la lectura del sensor desde un ordenador, procesar esta información y enviarla al UR3 para que este se encargue del control utilizando la información recibida.

-DESARROLLO DE LA PROGRAMACIÓN

Aquí se describen los procedimientos que se han ido llevando a cabo durante la realización del Trabajo Fin de Grado:

Lectura del sensor

En este apartado se explica el funcionamiento del sensor de fuerza/par ATI Delta y como se procesan las señales generadas:

El transductor responde a las fuerzas y pares aplicados de acuerdo con la tercera ley de Newton que establece que: Para cada acción siempre hay una reacción opuesta o igual; o, la acción mutua de dos cuerpos entre ellos siempre es igual y de dirección contraria.

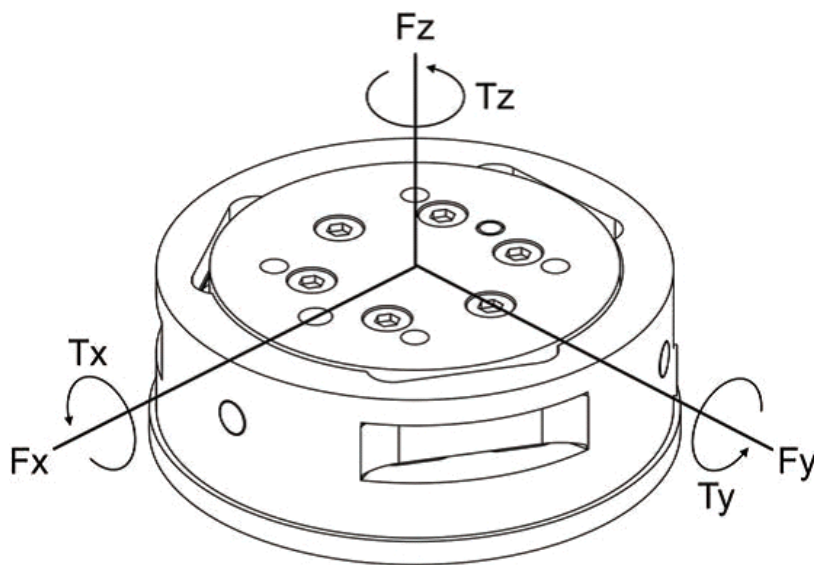


Figura 20- Dibujo representativo de los ejes del transductor

La fuerza aplicada al transductor flexiona tres haces colocados simétricamente usando la ley de Hooke: $s = E * e$

s = El estrés aplicado a la viga (proporcional a la fuerza)

E = Módulo de elasticidad de la viga

e = tensión aplicada a la viga

Las galgas extensiométricas son semiconductores unidos a las vigas y actúan como resistencias sensibles a las tensiones. La resistencia del medidor cambia en función de la tensión aplicada de la siguiente manera:

$$\Delta R = S_a * R_o * e$$

ΔR = Cambio en la resistencia de las galgas

S_a = Factor de medición de las galgas

R_0 = Resistencia de las galgas sin esfuerzos

E = Tensión aplicada a las galgas

Los datos de las galgas extensiométricas representan voltajes amplificados del transductor que se convierten en datos digitales por el sistema de adquisición de datos. Usando el programa de demostración para monitorear los datos de deformación durante la instalación se puede utilizar para evitar errores de saturación que podrían dañar el transductor.

Cálculo de las fuerzas y pares

Los cálculos deben realizarse para derivar las cargas detectadas en el transductor. El transductor envía las cargas como valores compuestos que requieren ser convertidos a valores correspondientes a los seis ejes cartesianos.

Para realizar esta conversión se ha de aplicar primero una corrección de la desviación. La corrección de la desviación es un vector de polarización que elimina los datos de fuerza y par compensando el peso de la herramienta o la variación de la temperatura ambiente. Si no se aplica esta corrección el peso de la herramienta se verá como datos de fuerza en el transductor lo que nos proporcionaría una medición de las fuerzas errónea.

Estos datos siguen siendo valores en voltios por lo que se ha de aplicar la matriz de calibración. La matriz de calibración del transductor es proporcionada por la propia empresa ATI. Esta matriz estándar es multiplicada por los datos corregidos de las galgas extensiométricas proporcionando los datos de fuerzas y pares que se pueden usar para la aplicación del control por fuerza.

ATI suministra el software necesario para realizar estos cálculos. Tanto al software como los valores de calibración se pueden acceder desde el CD que viene junto al transductor.

La siguiente figura muestra los cálculos requeridos para convertir los datos provenientes de las galgas en datos de fuerza y par. Los cálculos son proporcionados por ATI. Este método se puede aplicar a través de múltiples softwares que permitan el cálculo matricial como labVIEW.

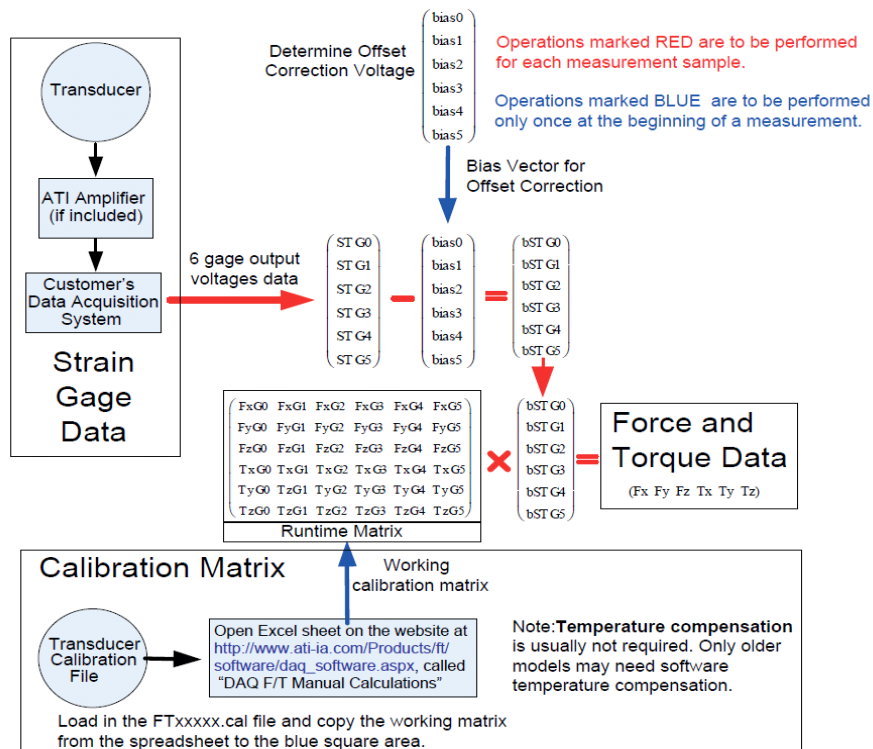


Figura 21- Cálculos para la conversión de voltaje a fuerzas del sensor

Programación en LabVIEW

La programación en LabVIEW proporcionada por ATI se observa en la siguiente imagen:

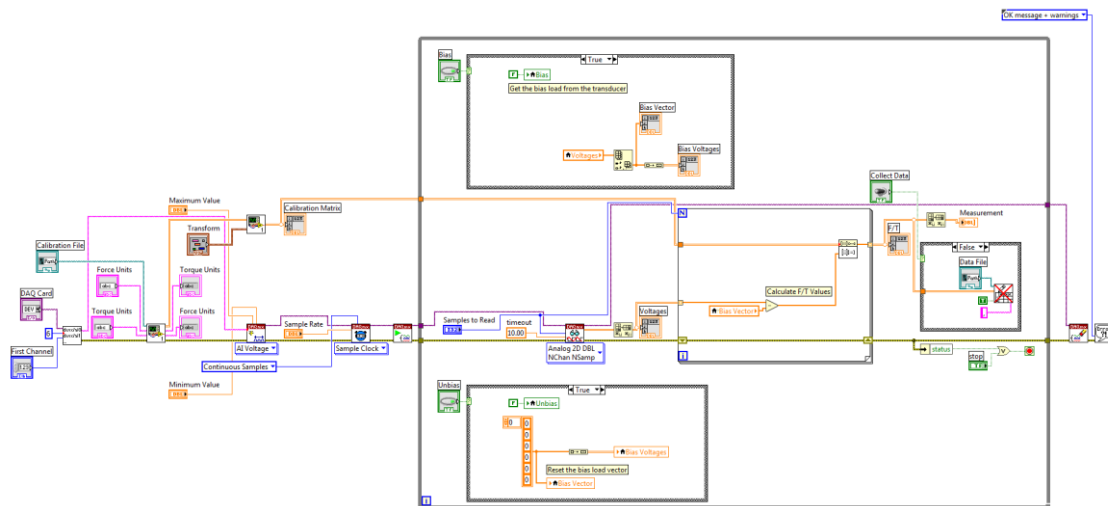


Figura 22- Diagrama de bloques de LabVIEW para la lectura del sensor

El programa en este estado permite visualizar en pantalla las lecturas del sensor a tiempo real pero no permite su envío por lo que se deben aplicar ciertas modificaciones a este programa:

1º Se crea un servidor TCP/IP para realizar la comunicación con el robot y poder enviar la información.

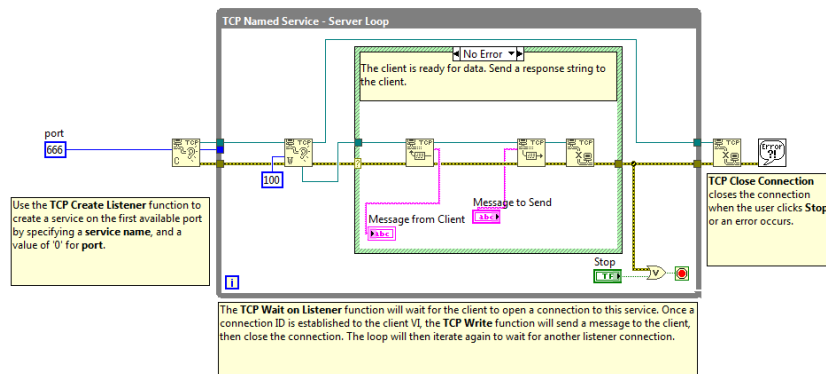


Figura 23- Diagrama de bloques de LabVIEW para servidor TCP/IP

2º Se incorpora el servidor TCP/IP al programa principal y se selecciona el valor que se quiere enviar. Dado que el programa genera una matriz de datos con todas las fuerzas y pares del sensor se extrae el valor para la fuerza en el eje Z ya que es el que nos interesa en este momento, en posteriores trabajos estos datos se podrían ampliar para enviar las 3 fuerzas y 3 pares leídos; como la fuerza en el eje Z se trata de un valor numérico y para enviarlo ha de ser un string entre paréntesis, estos datos se modifican por lo que el programa pasa a estar como se observa en la siguiente imagen.

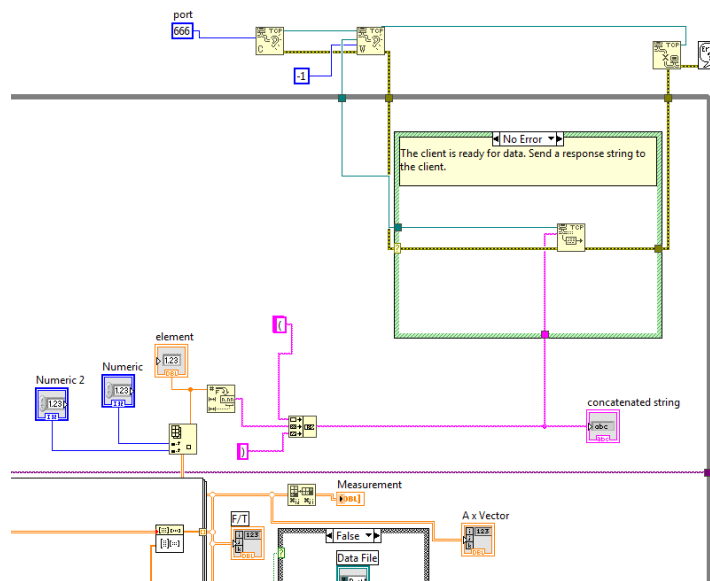


Figura 24- Comb. en LabVIEW del programa de lectura del sensor y del servidor TCP/IP

3º Cuando se ejecuta el programa se observa que salta un error de “buffer overflow” indicando que no es capaz de procesar todos los datos que recibe ya que la lectura es muy rápida. Como no afecta la pérdida de datos porque solo es necesaria la lectura más actual se ha de aplicar una sobrescritura del buffer y realizar la lectura únicamente del dato más reciente modificando el programa como se observa a continuación.

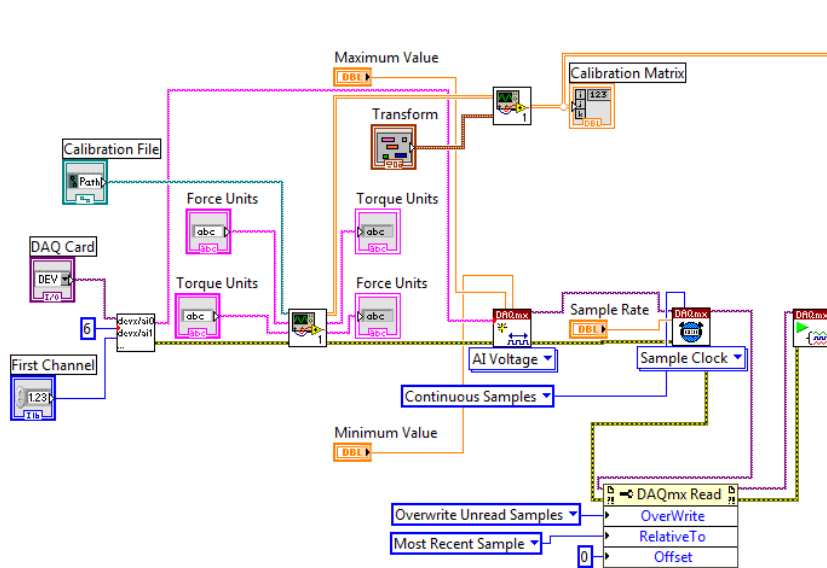


Figura 25- Modificación del programa de LabVIEW para sobrescribir el buffer

4º Finalmente el programa queda así:

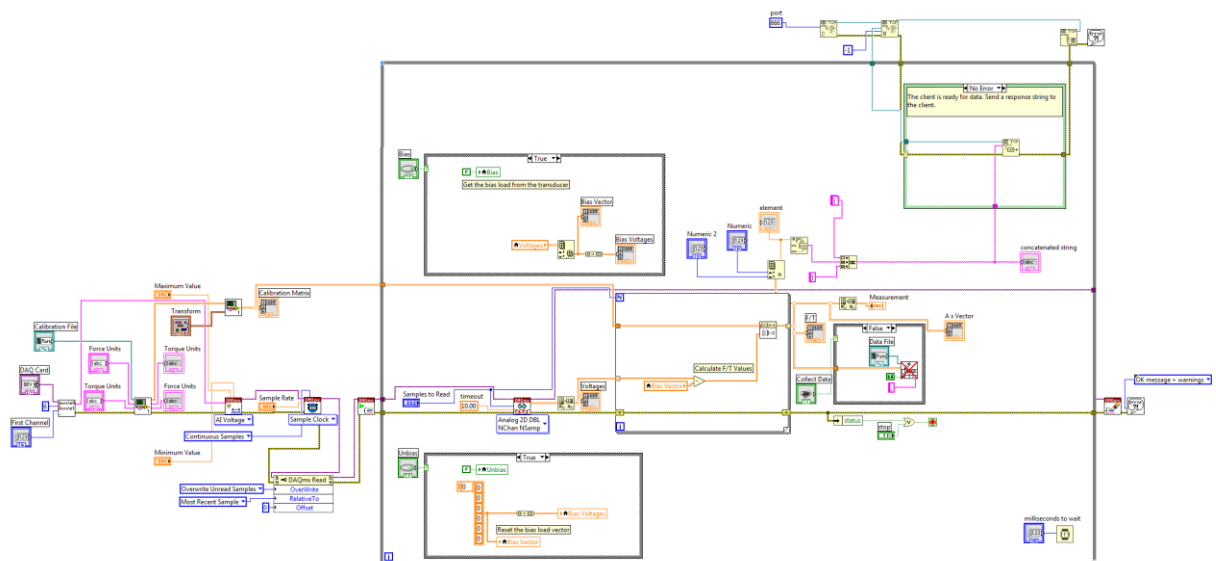


Figura 26- Diagrama de bloques de LabVIEW utilizado en el proyecto

Y cuando el programa está en funcionamiento se observa en la pantalla una imagen como la siguiente:

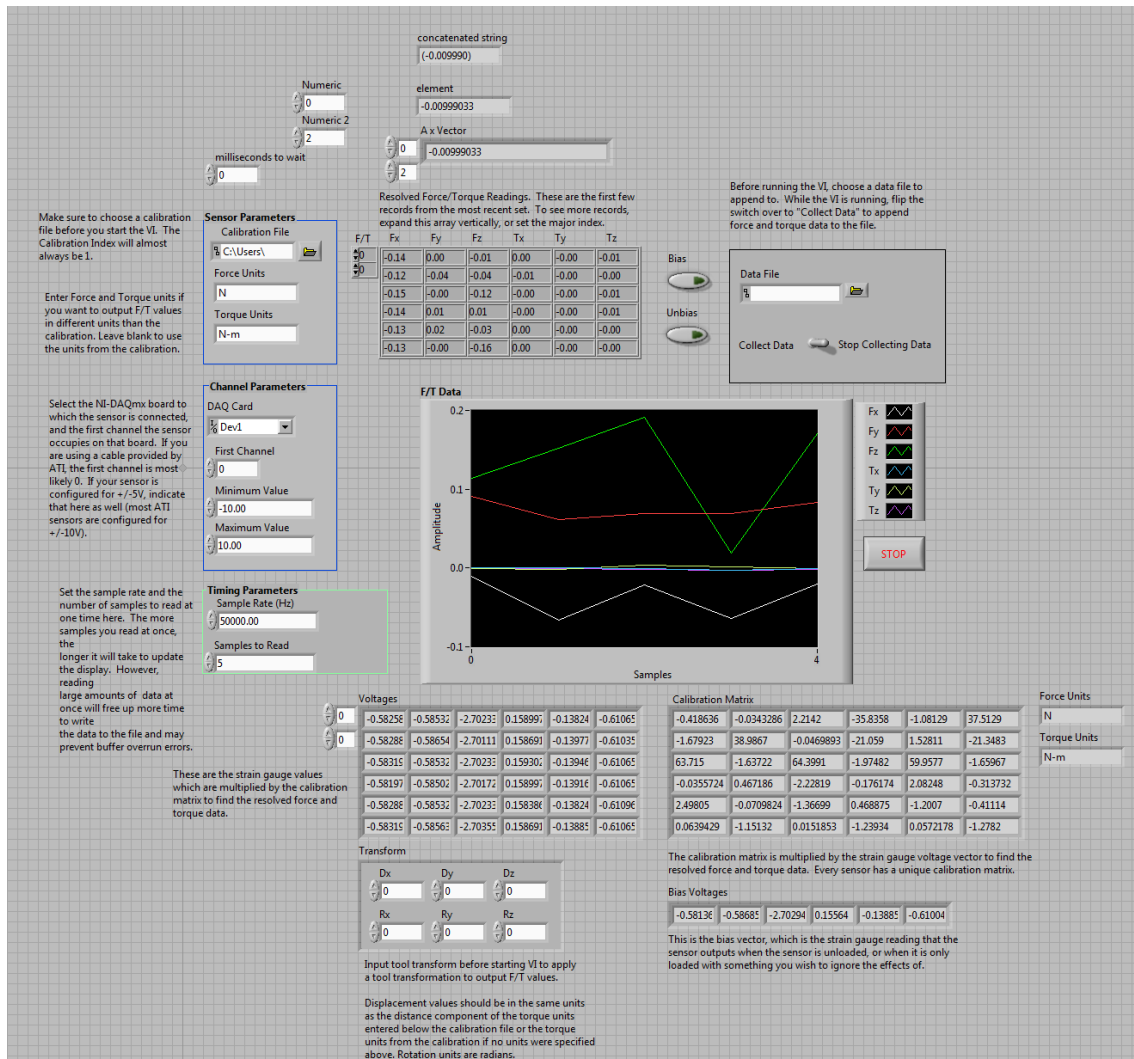


Figura 27- Imagen mostrada de LabVIEW en funcionamiento

Control mediante Polyscope

Para comenzar a trabajar con el robot colaborativo UR3 lo primero es comprender su funcionamiento, para ello está disponible en la página web de Universal Robots el manual de funcionamiento del robot donde se puede conocer todo lo necesario sobre el robot. Después de esto se procede a realizar contacto con el robot mediante la interfaz de programación PolyScope. Universal Robots proporciona este sistema de programación en sus robots permitiendo el poder configurarlos de una manera sencilla, intuitiva y fácil de aprender. Esta interfaz se encuentra en una pantalla táctil portable permitiéndonos movernos alrededor del robot de forma cómoda mientras se programa. También dispone de un puerto usb para poder guardar y cargar los programas desde un pen drive.

A continuación, se explica el procedimiento llevado a cabo en Polyscope:

-1 Aprender el uso de Polyscope

-Al comenzar el proyecto el sensor no se encontraba disponible por lo que ese tiempo se aprovechó para aprender sobre el funcionamiento del UR3 y la programación con Polyscope. Para ello se estudió el manual y se procedió a la realización de algunos tutoriales sobre programación en URScript, el lenguaje que se utiliza en los robots de Universal Robots.

-2 Control de fuerza con el comando force()

-Después de conocer los comandos principales y la configuración el robot se realizaron pruebas con la función “force()” de la lista de comandos. Esta función sirve para realizar un control de fuerza del robot utilizando los valores que éste estima a partir de unos algoritmos que tienen en cuenta las variaciones en sus ejes. Según la información proporcionada por Universal Robots este calculo de la fuerza tiene una precisión de ± 10 Newtons.

-3 Cliente para comunicación TCP/IP

-Para poder trabajar con los valores de fuerza medidos por el sensor estos datos tienen que ser enviados desde un ordenador hasta el robot, para ello se decidió utilizar una comunicación servidor-cliente TCP/IP utilizando el UR3 como cliente, por lo que se procede a desarrollar el programa encargado de realizar esta función.

Para ello se debe tener en cuenta que se quiere que la comunicación con el ordenador para la recepción de datos no se vea afectada por la ejecución de los comandos del programa principal por lo que se creará un subproceso en el programa para este fin. Permitiendo al programa actualizar el valor de la fuerza de forma independiente al funcionamiento del programa.

Para crear un cliente lo primero es abrir una conexión entre el ordenador y el robot llamado “socket”. El “socket” necesita una IP y un puerto. La IP debe ser la del servidor de la comunicación, en este caso el ordenador y el puerto puede ser cualquier número siempre que este puerto no esté siendo usado por otro dispositivo. Una vez conocida la IP y seleccionado un puerto se procede a escribir esta parte del código en el apartado “Before Start” ya que ha de realizarse la conexión antes de que el programa se inicie. Polyscope dispone en su programación de un

apartado llamado “Before Start” antes del programa principal que permite la escritura de una serie de comandos iniciales que se ejecutarán una única vez ya que el programa principal se ejecutará de forma cíclica mientras no se detenga.

Para abrir esta conexión se ha de crear un bucle que se encargue de intentar abrir esta conexión de forma repetida hasta que la comunicación se realice, de la siguiente forma:

Bucle open = False

└ *Open = socket_open(“158.42.206.20”,666)*

Una vez abierta la conexión con el ordenador el bucle terminará y el programa comenzará a funcionar.

Mientras que el programa esté en marcha se ejecutará de forma paralela un subproceso que se encargará de leer la información recibida por la conexión y guardarla en una variable. Esta información se trata de la fuerza leída por el sensor en el eje Z y se guardará en la variable: *fzsensor* que será utilizada por el programa principal. El subproceso tiene la siguiente estructura:

Subproceso

└ *receiveFromServ = socket_read_ascii_float()*

└ *fzsensor = receiveFromServ<[1]*

-4 Desarrollo del programa principal

Una vez comprobado que el envío de las lecturas funciona correctamente se procede a desarrollar el programa de control del robot.

Primero, en el apartado “Before Start” se definen las variables que vayamos a utilizar de la siguiente manera:

Before Start

└ *fzsensor = 0.0*

└ *Fref = -10.0*

└ *fuerzamax = 0*

└ *velomax = 0*

└ *velocidad = 0.005*

La variable “fzsensor” se encargará de guardar la lectura del sensor recibida a través de la comunicación TCP/IP, en la variable “Fref” escribimos el valor de la fuerza de referencia que queremos aplicar en valor negativo ya que el sensor entiende así la fuerza de compresión, en la variable “fuerzamax” se guarda la mayor lectura de fuerza aplicada para tener un control de los picos de fuerza que se apliquen y en “velomax” la velocidad con el mismo objetivo, También se le da un valor inicial a “velocidad” para que no hayan problemas al iniciar el programa.

Una vez definidas las variables se procede al desarrollo del programa principal.

La idea es que la herramienta del robot siempre se mantenga orientada de forma vertical, en el eje z. Inicialmente se desplace a una posición elevada sobre el punto donde se quiere aplicar la fuerza, después se aproxime a un centímetro del punto de contacto de forma lineal y luego lentamente vaya bajando verticalmente hasta que realice el contacto, momento en el cual se comenzará a realizar un control proporcional y derivativo de la velocidad mediante la fuerza para alcanzar la fuerza de referencia anteriormente definida. Utilizando los comandos de Polyscope el programa queda escrito de la siguiente manera:

Programa del robot

```
L  MoveJ
    L  Punto_de_paso_1
L  MoveL
    L  Punto_de_paso_2
L  Bucle fzsensor > -3
    L  velocidad = (Fref - fzsensor) * 0.0001
    L  speedl ([0,0, velocidad,0,0,0],10,0)
L  Bucle
    L  veloz = velocidad
    L  velocidad = (Fref - fzsensor) * 0.002
    L  velocidad = velocidad - (veloz * 0.3)
```


- └ *speedl*([0,0, *velocidad*,0,0,0],10,0)
- └ *If* -*fzsensor* > *fuerzamax*
 - └ *fuerzamax* = -*fzsensor*
- └ *If* *veloz* > *velomax*
 - └ *velomax* = *veloz*

A continuación, se pasa a explicar los comandos utilizados en el programa:

-*MoveJ*, el robot se desplaza hasta la posición “Punto_de_paso_1” haciendo que todos sus ejes giren al mismo tiempo permitiendo alcanzar la posición lo más rápido posible.

-*MoveL*, el robot se desplaza hasta la posición “Punto_de_paso_2” de forma lineal desde su última posición permitiendo un movimiento seguro y previsible hasta la zona de contacto.

-*Bucle fzsensor* > -3, este bucle se mantendrá mientras la lectura del sensor no supere un valor de 3 Newtons, esto permitirá controlar la bajada de la herramienta hasta que se realice el contacto.

-*velocidad* = (*Fref* – *fzsensor*) * 0.0001, aquí se ajusta la velocidad deseada aplicando un control proporcional a la fuerza, la constante proporcional es muy baja para que la aproximación sea muy lenta y evitar un impacto peligroso contra la superficie.

-*speedL*([0,0, *velocidad*,0,0,0],10,0), esta función hace que la herramienta del robot se desplace a la velocidad elegida, requiere 6 valores (3 para la velocidad en los ejes y 3 para velocidades angulares) por lo que como solo se permitirá el desplazamiento de la herramienta en el eje Z, este valor dependerá de la variable “velocidad” y se ajustarán el resto de valores como 0. Después necesita la aceleración, a la que se le da un valor de 10 m/s² para que la velocidad se alcance lo más rápido posible. Y, por último, el tiempo de ejecución se indicará como 0 para que la función termine lo antes posible.

-*Bucle*, este es el bucle de control por lo que se ejecutará de forma continuada durante el funcionamiento del programa.

-*veloz* = *velocidad*, en la variable “*veloz*” se guardará la última velocidad utilizada para poder aplicar un control derivativo.

-*velocidad* = $(Fref - fzsensor) * 0.002$, aquí se ajusta la velocidad deseada aplicando un control proporcional a la fuerza, la constante proporcional se modificará durante las pruebas ejecutadas más adelante.

- $velocidad = velocidad - (veloz * 0.3)$, en este comando se aplica el control derivativo, para ello se utiliza la variable “*veloz*” que guarda la velocidad del ciclo anterior, esta se multiplica por la constante derivativa que se modificará durante las pruebas ejecutadas más adelante.

- *If -fzsensor > fuerzamax*, el comando if permite que se ejecute la siguiente función si se cumple que la fuerza medida por el sensor es mayor que la fuerza máxima registrada.

- *fuerzamax = -fzsensor*, la variable “*fuerzamax*” guardará el valor leído por el sensor.

- *If veloz > velomax*, el comando if permite que se ejecute la siguiente función si se cumple que la velocidad actual de la herramienta es mayor que la velocidad máxima registrada.

- *velomax = veloz*, la variable “*velomax*” guardará el valor de la velocidad en el eje Z actual de la herramienta.

Una vez terminado el programa en Polyscope quedará como se muestra en la siguiente imagen.

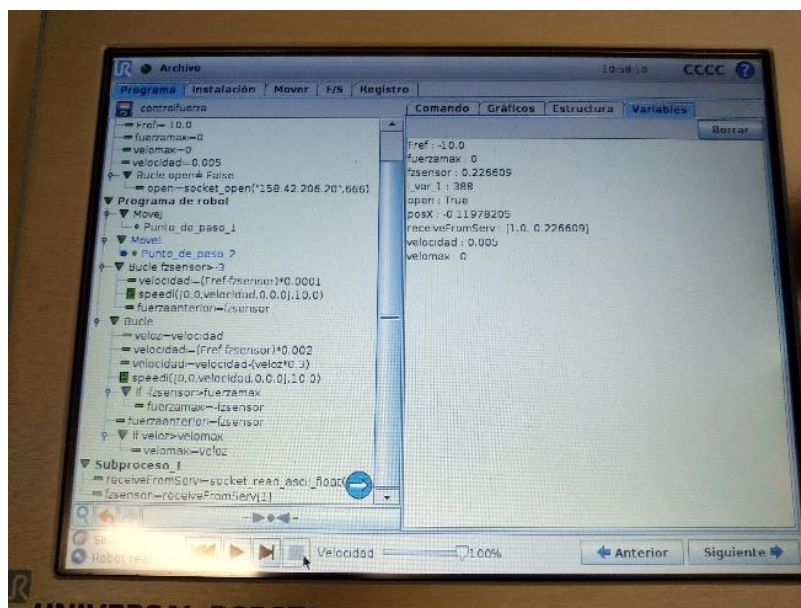


Figura 28- Programa para el control del UR3 en la pantalla de programación

Pruebas

Una vez programado el UR3 para realizar un movimiento vertical controlado por la variable de la fuerza en el eje Z, en LabVIEW la lectura del sensor y establecida la comunicación cliente-servidor entre el robot y el ordenador se procede a realizar el control del robot mediante un control proporcional derivativo. Se han realizado experimentos variando los valores de las constantes proporcional y derivativa y estudiando los resultados como los picos en la fuerza aplicada y la velocidad máxima alcanzada por la herramienta como se muestra en la siguiente tabla. Para estos experimentos se ha utilizado una fuerza de referencia de 10 Newtons para que en caso de sobreesfuerzo el robot no se dañe.

Kp	Kd	Fmax (N)	Vmax (m/s)	Observación
0.001	0	12.059	0.002	Vibración suave
0.002	0	17.11	0.014	Vibración
0.002	0.1	15.45	0.0107	Vibración suave
0.002	0.2	15.75	0.0129	Vibración suave
<u>0.002</u>	<u>0.3</u>	<u>14.59</u>	<u>0.009</u>	<u>Vibración suave</u>
0.002	0.4	15.47	0.0125	Vibración suave
0.002	0.5	15.71	0.0123	Vibración suave
0.002	0.6	16.36	0.0141	Vibración
0.002	0.7	17.04	0.0296	Vibración
0.0025	0	21.49	0.0287	Vibración fuerte
0.0025	0.1	20.59	0.0257	Vibración fuerte
0.0025	0.2	20.85	0.0275	Vibración fuerte
0.0025	0.3	18.36	0.020	Vibración
<u>0.0025</u>	<u>0.4</u>	<u>17.96</u>	<u>0.0172</u>	<u>Vibración</u>
0.0025	0.5	18.75	0.0182	Vibración
0.0025	0.6	20.13	0.020	Vibración fuerte
0.0025	0.7	27.32	0.083	Vibración inestable
0.003	0	23.598	0.0407	Rebote
0.003	0.1	23.001	0.0387	Rebote
0.003	0.2	23.528	0.0374	Rebote
0.003	0.3	23.823	0.0424	Vibración fuerte
0.003	0.4	23.78	0.0393	Vibración fuerte
<u>0.003</u>	<u>0.5</u>	<u>21.84</u>	<u>0.0294</u>	<u>Vibración</u>
0.003	0.6	22.66	0.049	Vibración inestable
0.003	0.7	22.67	0.0569	Vibración inestable

Figura 29- Tabla de resultados con fuerza de referencia de 10 Newtons

Los mejores resultados para cada constante proporcional se han realizado guardando las lecturas del sensor para generar las siguientes gráficas teniendo en cuenta que las fuerzas de compresión del sensor se leen en valores negativos.

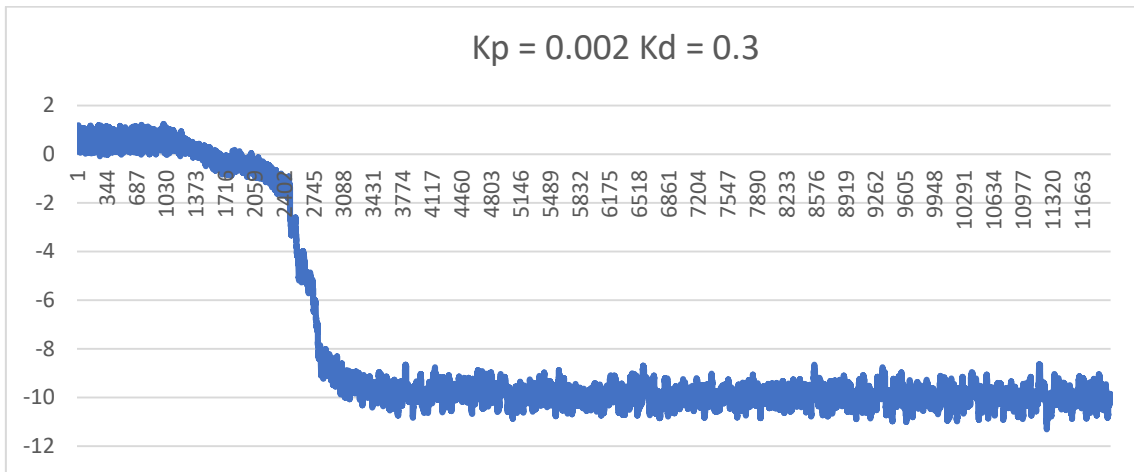


Figura 30- Gráfica de lecturas para $K_p = 0.002$ y $K_d = 0.3$

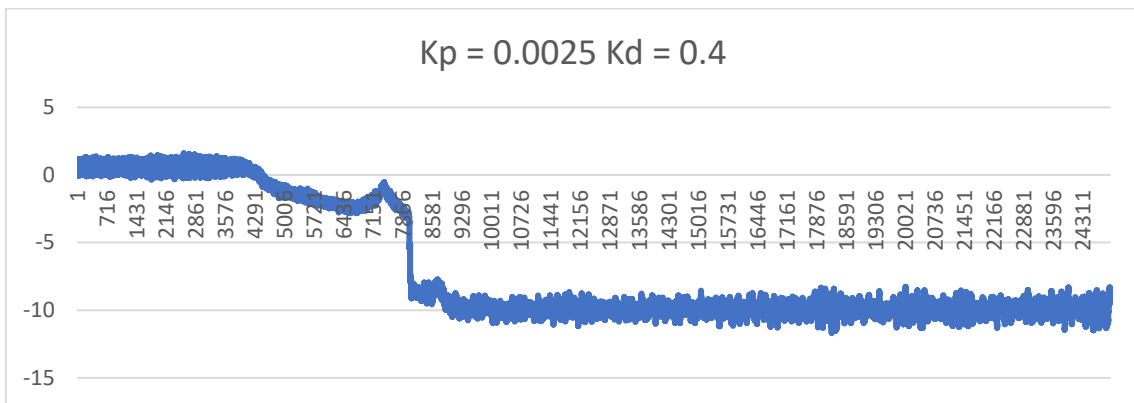


Figura 31- Gráfica de lecturas para $K_p = 0.0025$ y $K_d = 0.4$

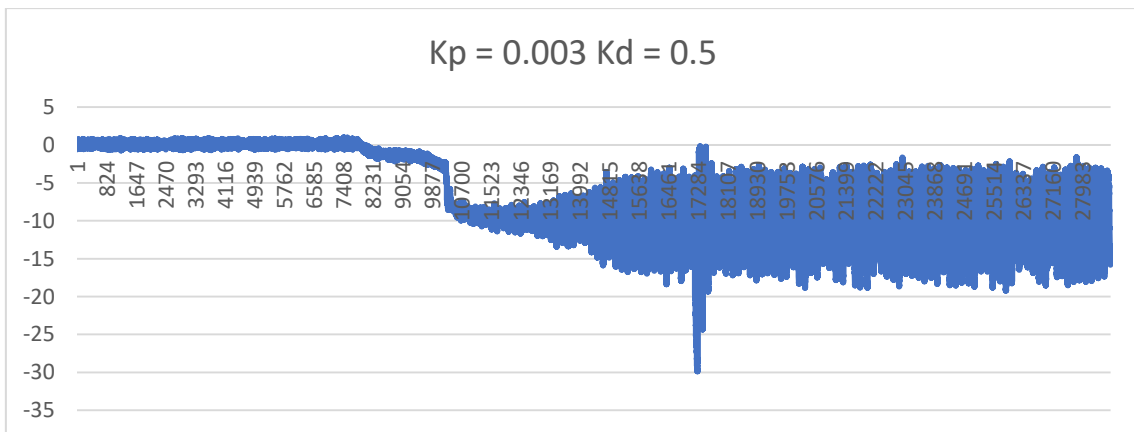


Figura 32- Gráfica de lecturas para $K_p = 0.003$ y $K_d = 0.5$

Después se realizaron pruebas con los mejores resultados con diferentes fuerzas de referencia para estudiar las variaciones en la precisión que esto provocaba.

Fref (N)	Kp	Kd	Fmax (N)	Vmax (m/s)
5	0.002	0.3	7.57	0.0049
5	0.0025	0.4	7.79	0.0063
5	0.003	0.5	10.51	0.0131
15	0.002	0.3	25.999	0.0209
15	0.0025	0.4	29.64	0.047
15	0.003	0.5	31.47	0.0699

Figura 33- Tabla con los resultados para fuerzas de referencia de 5 y 15 Newtons

Como se puede observar al aumentar la fuerza de referencia la diferencia respecto a la fuerza máxima que se alcanza es mayor.

-CONCLUSIONES

Después del desarrollo del proyecto podemos decir que se han cumplido los objetivos propuestos para este trabajo.

Se han adquirido amplios conocimientos sobre la mecánica del robot colaborativo y sus métodos de programación.

Se comprende el funcionamiento de los sensores de fuerza-par y sus múltiples usos.

Se han aplicado conocimientos de teoría de control.

Se ha desarrollado una comunicación entre ordenador y robot de forma funcional.

Y se han diseñado bases para el desarrollo de un control por fuerza preciso para un robot colaborativo con controlador cerrado.

Para futuros trabajos se propone el uso de un sensor de mayor precisión, la implementación de controladores externos más complejos como un control adaptativo, la realización de pruebas con otros bucles internos como la aceleración y el uso de las 3 fuerzas y 3 pares medidos por el sensor para realizar el control con desplazamiento en varias direcciones.

-PRESUPUESTO

Coste en materiales y personal usados en la realización del proyecto:

COMPONENTES	PRECIO (€)	UNIDADES	COSTE (€)
UR3	20000	1	20000
Sensor ATI Delta	5000	1	5000
Ordenador	800	1	800
Adaptador robot-sensor mecanizado	50 / hora	2 horas	100
Herramienta mecanizada	35 / hora	3 horas	105
TOTAL			26005

Figura 34- Tabla del presupuesto en materiales

-REFERENCIAS BIBLIOGRÁFICAS

ATI INDUSTRIAL AUTOMATION. F/T Controller Six-Axis Force/Torque Sensor System. Installation and Operation Manual.

ATI INDUSTRIAL AUTOMATION. F/T Transducer. Six-Axis Force/Torque Transducer. Installation and Operation Manual.

ATI INDUSTRIAL AUTOMATION. F/T Sensor. Data Acquisition (DAQ) Systems. Manual

UNIVERSAL ROBOTS. UR3/CB3 Versión 3.1. Manual de usuario

UNIVERSAL ROBOTS. Versión 3.1 The URScript Programming Language

NATIONAL INSTRUMENTS. Introducción a LabVIEW: <http://www.ni.com/getting-started/labview-basics/esa/>

Control de fuerza y de impacto de un robot con cambio de período de muestreo, Ranko Zotovic Stanisic

Theory of Robot Control, Carlos Canudas de Wit, Bruno Siciliano and Georges Bastin

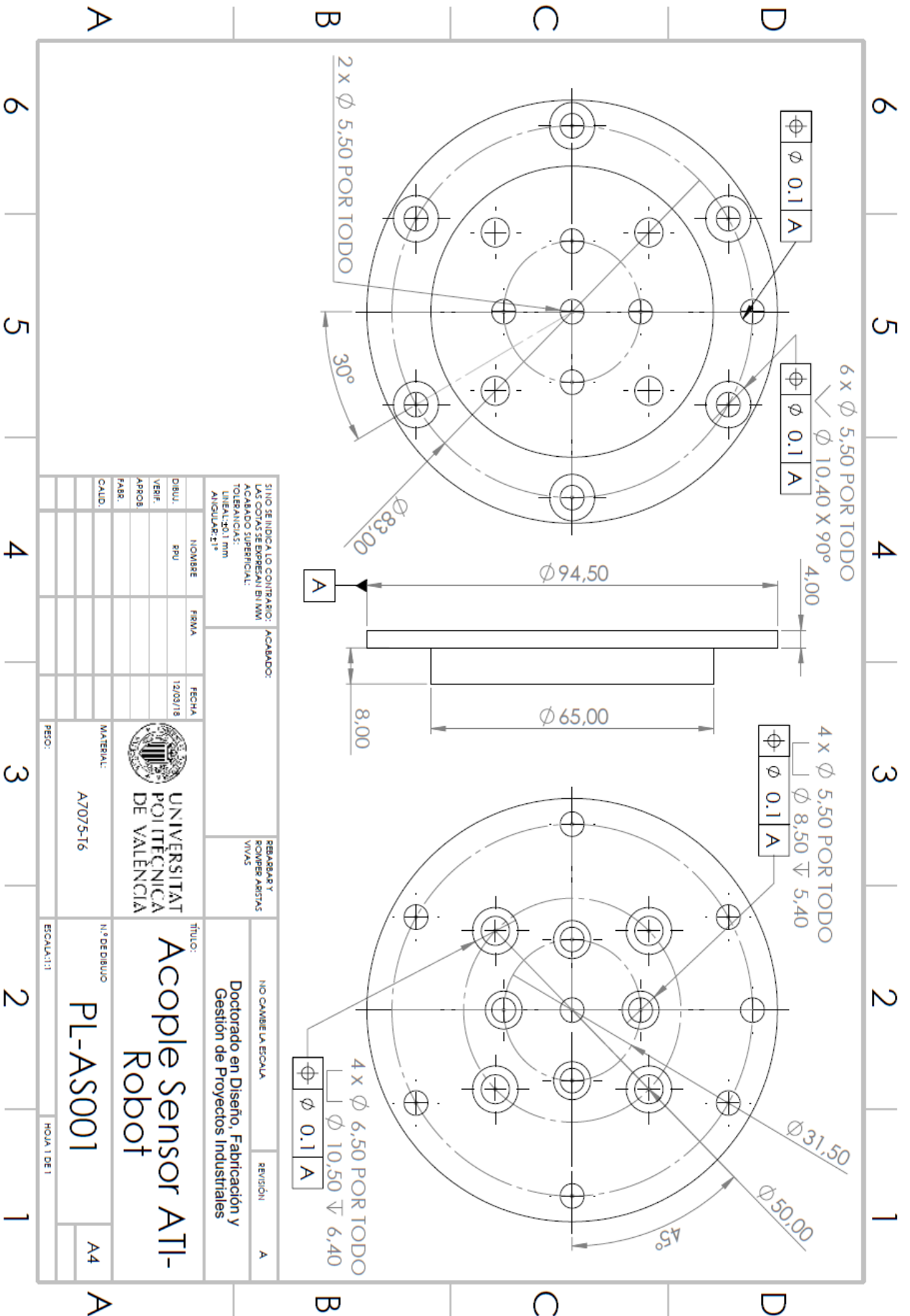
Springer Handbook of Robotics, Bruno Siciliano, Oussama Khatib (Eds.)

Control difuso de fuerza en robótica industrial, J. Norberto Pires, Tiago Godinho

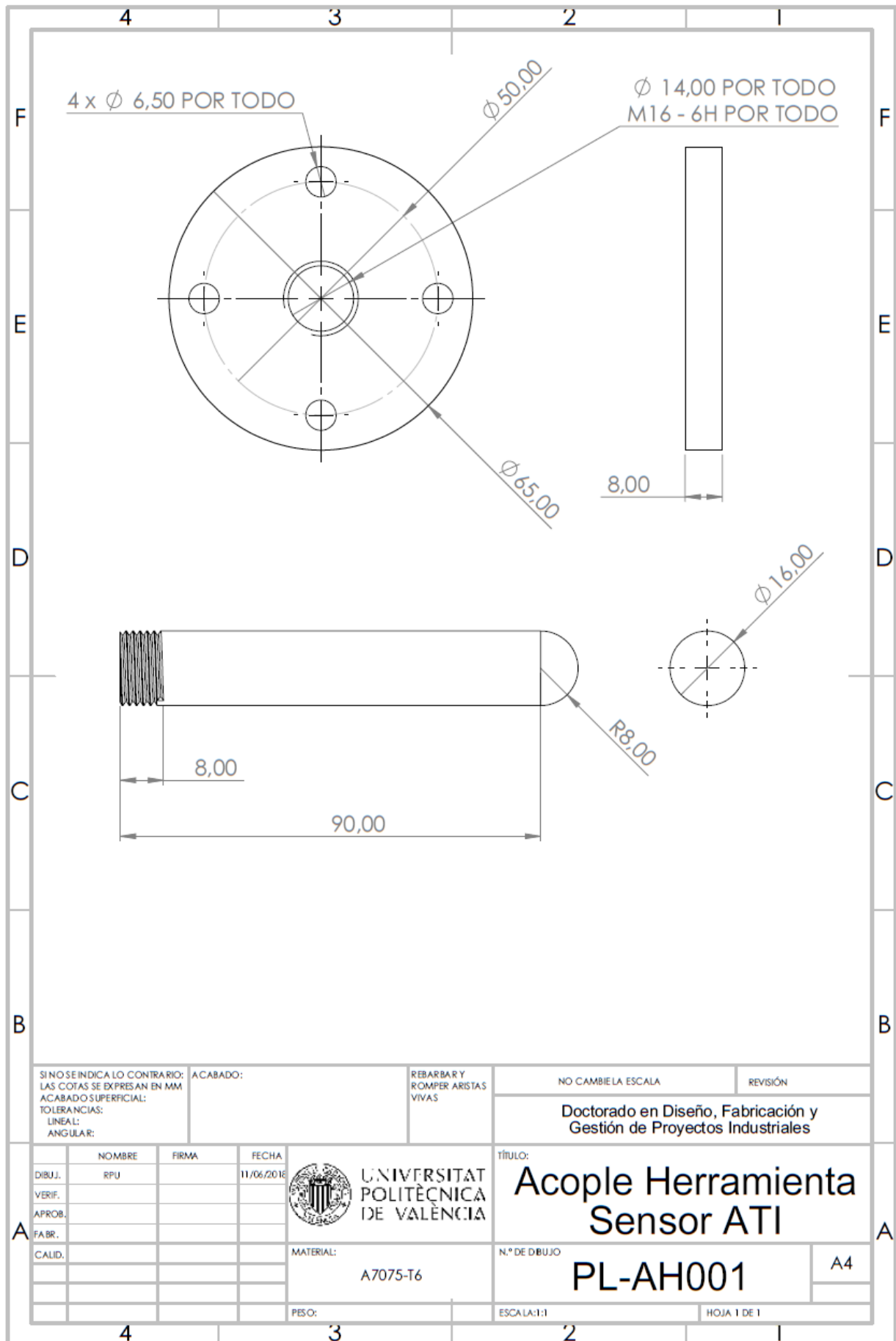
Robot Modeling and Control, Mark W. Spong, Seth Hutchinson, M. Vidyasagar

-PLANOS

-Plano del acople del transductor al robot UR3



-Plano del acople de la herramienta al sensor Delta



SINO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
ACABADO SUPERFICIAL:
TOLERANCIAS:
LINEAL:
ANGULAR:

ACABADO:

REBARBAR Y
ROMPER ARISTAS
VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

Doctorado en Diseño, Fabricación y
Gestión de Proyectos Industriales

	NOMBRE	FIRMA	FECHA
DIBUJ.	RPU		11/06/2016
VERIF.			
APROB.			
FA BR.			
CALID.			



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

TÍTULO:

Acople Herramienta
Sensor ATI

N.º DE DIBUJO

PL-AH001

A4

MATERIAL:

A7075-T6

PESO:

ESCALA:1:1

HOJA 1 DE 1