**Universidad Tecnológica Nacional**

Proyecto Final

Sistema de ensayo y calibración de sensores inductivos de aplicación nuclear.

*Autores:*

* Longo, Enzo Renato.
* Sigura, Milton Nahuel.

*Director:* Ing. Taglialavore, Eduardo.

*Proyecto final presentado para cumplimentar los requisitos académicos*

*para acceder al título de Ingeniero en Electrónica*

*en la*

**Facultad Regional Paraná**

Febrero de 2018

**Declaración de autoría:**

Nosotros declaramos que el Proyecto Final “Sistema de ensayo y calibración de sensores inductivos de aplicación nuclear” y el trabajo realizado son propios. Declaramos:

* Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero en Electrónica, en la Universidad Tecnológica Nacional, Regional Paraná.
* Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
* Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
* Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
* Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firmas:

Fecha:

Agradecimientos:

Agradezco por sobre todo a mi madre por haberme brindado la posibilidad de acceder a la carrera universitaria y por el apoyo incondicional estos años. Luego agradezco a mis hermanos por su comprensión y ayuda en todo momento. Un gran agradecimiento a la familia Sigura por su apoyo y contención desde el primer año de facultad. Y un agradecimiento a mi amigo y compañero Milton por dejarme formar parte de éste emocionante proyecto.

Longo Enzo Renato

En primer lugar quiero agradecer a mis padres, por todo el apoyo que me han brindado a lo largo de mi vida, por su confianza y tolerancia, a mi hermano que me presto su ayuda en incontables ocasiones, a mis abuelos tíos y primos que me acompañaron e incentivaron para continuar adelante y particularmente a mi tío Aldo Sigura, quien supo contagiarme desde chico el amor por esta profesión.

A los compañeros y amigos que hice a lo largo de la carrera, por todos los buenos momentos vivido, por su compañerismo y por empujarme a seguir.

También quiero agradecer a la facultad regional Paraná, al personal docente y no docente que la compone por haberme permitido aprender y crecer durante todos estos años.

Finalmente a Enzo, por haber aceptado ser mi compañero de proyecto.

Sigura Milton Nahuel.

Universidad Tecnológica Nacional

*Abstract*

Facultad Regional Paraná

Ingeniero en Electrónica

**Sistema de ensayo y calibración de sensores inductivos de aplicación nuclear.**

Longo, Enzo Renato.

Sigura, Milton Nahuel.

**Abstract:**

This project deals with the design and manufacture of a test and calibration system, of inductive sensors, to be used in the stages of research, development and calibration of the position sensors of the control and safety bars of the modular elements nuclear reactor CAREM 25.

To carry it out, an FPGA device was used as support for the electronic hardware, on which an asymmetric load multiprocessor system and shared DDR III memory was implemented, the 4 MicroBlaze processors are in charge of controlling the temperature parameters of a 6-stage oven with control in staircase, the position and dynamics of movement with a hardware closed loop stepper motor, the acquisition of data in real time and the communication with the PC through the MODBUS TCP protocol respectively.

Once concluded, the simultaneous simulation of the temperature parameters with an accuracy of 0.1 ° C, the dynamic positioning with a margin of error of 0.02 [mm], the acquisition of data in real time and its subsequent overturning to the data analysis system was achieved. All this repetitively from the implementation of a system of projection and protection of experiences.

**Keywords:**

Desig environment, Movement dynamic control, Multiple cascade control, Real time data adquisition, Multiple processor system.

**Resumen:**

El presente proyecto trata el diseño y fabricación de un sistema de ensayo y calibración, de sensores inductivos, para su uso en las etapas de investigación desarrollo y calibración de los sensores de posición de las barras de control y seguridad a ser utilizadas en el reactor nuclear de elementos modulares CAREM 25.

Para llevarlo a cabo se utilizó como soporte del hardware electrónico un dispositivo FPGA sobre el que se implementó un sistema multiprocesador de carga asimétrica y memoria DDR III compartida, los 4 procesadores MicroBlaze se encargan de controlar los parámetros de temperatura de un horno de 6 etapas con control en escalera, la posición y dinámica de movimiento con un motor paso a paso de lazo cerrado por hardware, la adquisición de datos en tiempo real y la comunicación con la PC mediante el protocolo MODBUS TCP respectivamente.

Una vez concluido se logró la simulación simultánea de los parámetros de temperatura con una precisión de 0,1°C, el posicionamiento dinámico con un margen de error de 0,02[mm], la adquisición de datos en tiempo real y su posterior volcado al sistema de análisis de datos. Todo esto de forma repetitiva a partir de la implementación de un sistema de proyección y resguardo de las experiencias.

**Palabras Clave:**

**Entorno de desarrollo, Control de dinámica de movimiento, Múltiple control en cascada, Adquisición de datos en tiempo real, Sistema de múltiples procesadores.**

*Reconocimientos:*

Al Ing. Maximiliano Sonnaillon por su recomendación para la realización de este trabajo y al Ing. Gustavo G. Katzenelson quien nos brindó su valioso tiempo y amplia experiencia para guiarnos en el desarrollo del mismo.

Al equipo de trabajo de la “División de desarrollos Electrónicos” de Centro Atómico Bariloche, Juan Pablo Mercado, Juan Pablo Medina, Matías Lorenzon, Ing. Leonardo Ortiz, Ing. Horacio Mendieta, Ing. Eduardo Taglialavore e Ing. Iván Nomdedeu por su disposición y apoyo técnico.

A nuestros amigos Santiago Cuestas, Marcelo Lissa, Nicolás Wagner, Andrés Pagura, Ezequiel Tello de Meneses, Javier Acosta y Emiliano Estrubia por la ayuda que nos brindaron, tanto por sus consejos técnicos, como por su soporte moral.

**Dedicado a:**

Nuestros padres, hermanos, familiares y amigos.

Índice:

[Capítulo 1: Introducción 1](#_Toc507513203)

[1.1. Fundamentación. 1](#_Toc507513204)

[1.2. Objetivos. 3](#_Toc507513205)

[Capítulo 2: Desarrollo. 5](#_Toc507513206)

[Descripción general. 5](#_Toc507513210)

[2.1. Controlador Principal. 6](#_Toc507513212)

[Introducción. 6](#_Toc507513213)

[Diseño propuesto. 6](#_Toc507513214)

[Problemas y soluciones implementadas. 15](#_Toc507513215)

[Desarrollo del firmware. 16](#_Toc507513216)

[Resultados parciales. 20](#_Toc507513217)

[2.2. Controlador de Temperatura. 20](#_Toc507513218)

[Introducción. 20](#_Toc507513219)

[Diseño propuesto. 21](#_Toc507513220)

[*Hardware Electrónico.* 21](#_Toc507513221)

[*Hardware Mecánico.* 21](#_Toc507513222)

[*Simulador del sistema.* 25](#_Toc507513223)

[Problemas y soluciones implementadas. 26](#_Toc507513224)

[*Control en cascada.* 27](#_Toc507513225)

[*Calibración del sistema.* 28](#_Toc507513226)

[Desarrollo del firmware. 31](#_Toc507513227)

[Resultados parciales. 33](#_Toc507513228)

[2.3. Controlador de Movimiento. 34](#_Toc507513229)

[Diseño propuesto. 34](#_Toc507513230)

[*Hardware Electrónico.* 34](#_Toc507513231)

[*Hardware Mecánico.* 37](#_Toc507513232)

[Problemas y soluciones implementadas. 40](#_Toc507513233)

[Desarrollo del firmware. 43](#_Toc507513234)

[Resultados parciales. 44](#_Toc507513235)

[2.4. Sistema de adquisición de datos. 45](#_Toc507513236)

[Introducción. 45](#_Toc507513237)

[Diseño propuesto. 45](#_Toc507513238)

[Problemas y soluciones implementadas. 48](#_Toc507513239)

[Desarrollo del firmware. 49](#_Toc507513240)

[Resultados parciales. 52](#_Toc507513241)

[2.5. Interfaz HMI. 53](#_Toc507513242)

[*La interfaz de hardware.* 53](#_Toc507513243)

[Software de simulación de entorno controlado. 54](#_Toc507513244)

[*La interfaz de software.* 54](#_Toc507513245)

[*Generación de una trayectoria.* 55](#_Toc507513246)

[*Simulación de la trayectoria programada.* 57](#_Toc507513247)

[*Ingreso de parámetros de configuración.* 60](#_Toc507513248)

[Almacenamiento de datos generados. 61](#_Toc507513249)

[Resultados parciales. 61](#_Toc507513250)

[Capítulo 3: Resultados. 63](#_Toc507513251)

[Capítulo 4: Análisis de Costos. 64](#_Toc507513252)

[Capítulo 5: Discusión y Conclusión. 66](#_Toc507513253)

[Capítulo 6: Literatura Citada. 69](#_Toc507513254)

# Lista de figuras:

[Figura 1: Placa de desarrollo ARTY. 8](#_Toc507512850)

[Figura 2: Fuente switching 9[V]. 9](#_Toc507512851)

[Figura 3: Circuito tablero de comando. 10](#_Toc507512852)

[Figura 4: Controlador del tablero de comando. 14](#_Toc507512853)

[Figura 5: Diseño de horno. 22](#_Toc507512854)

[Figura 6: Resistencia tipo abrazadera. 23](#_Toc507512855)

[Figura 7: Relé de estado sólido. 23](#_Toc507512856)

[Figura 8: UAD ADAM4018+ 24](#_Toc507512857)

[Figura 9: Controlador PID DTA4848 24](#_Toc507512858)

[Figura 10: Software de simulación de horno 25](#_Toc507512859)

[Figura 11: Primer prueba de horno 26](#_Toc507512860)

[Figura 12: Segunda prueba de horno 27](#_Toc507512861)

[Figura 13: Ajuste de controladores PID 30](#_Toc507512862)

[Figura 14: Prueba de control en escalera. 31](#_Toc507512863)

[Figura 15: Prueba de control en escalera. 31](#_Toc507512864)

[Figura 16: Motor Nema 24H2A9830 35](#_Toc507512865)

[Figura 17: Driver Leadshine M542 36](#_Toc507512866)

[Figura 18: Fuente switching 37](#_Toc507512867)

[Figura 19: Encoder G40B-6-400-2-240 37](#_Toc507512868)

[Figura 20: Cabezal de sujeción y desplazamiento 39](#_Toc507512869)

[Figura 21: Actuador lineal 39](#_Toc507512870)

[Figura 22: Curva de trayectoria S 42](#_Toc507512871)

[Figura 23: Adaptador RS485 46](#_Toc507512872)

[Figura 24: Sensor Accu-Coder Tru-Trac. 47](#_Toc507512873)

[Figura 25: Esquema adaptador del sensor de contraste 48](#_Toc507512874)

[Figura 26: Firmware sistema de adquisición de datos 50](#_Toc507512875)

[Figura 27: Panel de control 54](#_Toc507512876)

[Figura 28: Interfaz de generación de trayectoria. 57](#_Toc507512877)

[Figura 29: Interfaz de simulación. 60](#_Toc507512878)

[Figura 30: Interfaz de configuración 61](#_Toc507512879)

# Lista de Tablas:

[Tabla 1: Parámetros sistema externo del horno. 29](#_Toc507512988)

[Tabla 2: Parámetros controladores PID externos del horno. 29](#_Toc507512989)

[Tabla 3: Parámetros sistema interno del horno. 30](#_Toc507512990)

[Tabla 4: Parámetros controladores PID internos del horno. 30](#_Toc507512991)

[Tabla 5: Comandos de simulación del controlador de temperatura 33](#_Toc507512992)

[Tabla 6: Variables a medir y sus características. 51](#_Toc507512993)

# Lista de diagramas:

[Diagrama 1: Bloques del sistema. 5](#_Toc507513045)

[Diagrama 2: Controlador Soft-Core MicroBlaze. 12](#_Toc507513046)

[Diagrama 3: Controlador principal. 13](#_Toc507513047)

[Diagrama 4: Memoria de uso compartido. 15](#_Toc507513048)

[Diagrama 5: Firmware MODBUS Ethernet. 17](#_Toc507513049)

[Diagrama 6: Firmware controlador principal. 19](#_Toc507513050)

[Diagrama 7: Controlador de temperatura. 21](#_Toc507513051)

[Diagrama 8: Control en cascada. 28](#_Toc507513052)

[Diagrama 9: Wind-Up Integral 28](#_Toc507513053)

[Diagrama 10: Firmware controlador de temperatura. 32](#_Toc507513054)

[Diagrama 11: Controlador de movimiento. 34](#_Toc507513055)

[Diagrama 12: Sistema de control de lazo cerrado del motor 40](#_Toc507513056)

[Diagrama 13: Sistema de adquisición de datos. 45](#_Toc507513057)

[Diagrama 14: Interfaz de generación de trayectoria. 56](#_Toc507513058)

[Diagrama 15: Interfaz de simulación. 58](#_Toc507513059)

[Diagrama 16: Interfaz de comunicación. 60](#_Toc507513060)

# Lista de abreviaciones:

MSAC: Mecanismo del Sistema de Ajuste y Control.

MSER: Mecanismo del Sistema de Extinción Rápida.

CAREM: Central argentina de elementos modulares.

CNEA: comisión nacional de energía atómica.

PC: Computadora Personal.

FPGA: Matriz de puertas programables (del inglés Field Programmable Gate Array).

HMI: Interfaz Humano Maquina.

PMOD: interfaces de conexión de 6x2 pines hembra, propietarios de la firma Digilent.

ARTY: plataforma de desarrollo basad en FPGA de la firma Digilent.

AXI: Bus de interconexión de IPs.

ASIC: Circuito integrado de aplicación específica.

IP: Funciones lógicas pre configuradas de propiedad intelectual

RISC: Computador con conjunto de instrucciones reducidas.

RAM: Memoria de acceso aleatorio.

ROM: Memoria de solo lectura.

DDR: “*Double Data Rate”*, módulos de memoria RAM con la capacidad de transferir simultáneamente datos por dos canales distintos en un mismo ciclo de reloj

FLASH: tecnología de almacenamiento de memoria ROM.

SPI: Bus serial de interfaz de periféricos.

UART: Transmisor – Receptor asíncrono universal.

FTDI: *“Future Technology Devices International”*

USB: Bus serial universal

VHDL: Lenguaje especifico de descripción de hardware.

ACM/IEEE: Publicación científica focalizada en las redes de comunicación.

LWIP: *“lightweight IP”*

BSD: “*Berkeley Software Distribution”*

# Capítulo 1: Introducción

El diseño de sensores utilizados en ambientes hostiles, se encuentra limitado por la incapacidad de los diseñadores de recrear las condiciones de utilización real. Estos impedimentos se sortean, en las primeras etapas, mediante la utilización de software de simulación y diversas herramientas que permiten obtener datos aproximados. Sin embargo, en algunos casos, estas soluciones resultan insuficientes debido a la gran variedad de factores intervinientes.

Este problema surge en la etapa final de investigación de los sensores inductivos de posición, a utilizarse en los actuadores MSAC y MSER del reactor experimental “CAREM 25”, que lleva adelante la “División de Desarrollos Electrónicos” del “Centro Atómico Bariloche”. En este caso resulta necesario contar con un entorno adecuado para la prueba del dispositivo, que además provea al investigador de elementos de contraste para poder comprobar funcionalidad y comportamiento ante distintas situaciones de uso.

Se plantea a lo largo de este trabajo el desarrollo de un entorno integrado de simulación, medición y contraste, adecuado a las tareas de prueba y calibración de estos sensores.

## Fundamentación.

El concepto CAREM (Central Argentina de Elementos Modulares) nació a principios de la década de 1980, cuando la Comisión Nacional de Energía Atómica (CNEA) decidió avanzar por primera vez sobre el diseño de su propio reactor nuclear de potencia.

En el año 2006, a partir del Decreto 1107, el Poder Ejecutivo Nacional declaró de Interés Nacional “la construcción y puesta en marcha del Prototipo de Reactor CAREM para la generación nucleoeléctrica de energía”, lo que impulsó a la CNEA para la creación de la Gerencia CAREM, otorgándole al proyecto entidad formal e institucional (Disposición 17 - 01/12/2006) dentro de la esfera de su Gerencia de Área Energía Nuclear (GAEN).

Comenzó entonces un proceso de recuperación y clasificación de la información de ingeniería; se aglutinaron grupos técnicos de la CNEA y se contrató personal necesario para completar especialidades, mientras se concretó el primer costeo del reactor.

Uno de los componentes principales y que marca un salto tecnológico respecto de similares modelos de reactor, es el mecanismo de posicionamiento de las barras de control (MSER), que se encuentra inmerso en el interior del reactor. Junto a este nace la necesidad de un instrumento de medición, capaz de determinar la posición efectiva de dicha barra. En principio esto no es una tarea complicada, sin embargo resulta serlo si se consideran las condiciones de funcionamiento del dispositivo.

Se ha diseñado un sensor, basado en la medición de la variación en los parámetros de un inductor colocado sobre el mecanismo de posicionamiento. Según lo indican Sparza D’Ovidio y Taglilavore (1999) se destacan en el mismo su robustez mecánica, posibilidad de usar materiales resistentes a las condiciones de trabajo, usa solo dos cables para su conexión y es un método sin contacto lo que garantiza la ausencia de desgaste.

Las primeras pruebas experimentales de este dispositivo realizadas en el año 2006 han arrojado excelentes resultados, sin embargo las mismas fueron realizadas sobre un modelo que no consideraba la temperatura a la que se sometería el sensor. Pruebas posteriores revelaron un preocupante decremento en la sensibilidad del sensor cuando este es sometido a temperaturas similares a la de trabajo.

A partir del año 2006 se trabajó en la determinación de los parámetros que producen la perdida de sensibilidad, y variaciones en las mediciones al someterlo a temperatura, luego de detallados estudios de materiales y diversas formas de sensor, se ha logrado un prototipo funcional.

Nace en este punto la necesidad de ensayar este prototipo en un ambiente comparable al de funcionamiento final, dada la complejidad de este entorno, se determinaron las principales variables a recrear de forma de obtener un grado elevado de certeza sobre la funcionalidad del sensor. Estas son:

* Temperatura:

Como se expresó anteriormente, la temperatura del ambiente en el que se encuentra inmerso el sensor aparece como el principal elemento de variación sobre las características física del sensor.

* Posicionamiento.

Contar con un mecanismo de posicionamiento similar al que se utilizara permite comprobar la efectividad de la medición a realizar y es de vital importancia en el ensayo de los prototipos.

* Dinámica de movimiento.

La posición de la barra es la variable a medir, sin embargo no se puede despreciar la dinámica del movimiento que lleva la barra de una posición a otra, poder emular este factor entregará una mayor certeza sobre el desempeño del sensor desarrollado.

Se propone entonces la construcción de un sistema capaz de recrear las variables antes detalladas, de forma de poder controlar de forma repetitiva los parámetros de ensayo del sensor y verificar su funcionamiento.

Se contemplan:

* La adecuación de un horno existente, capaz de recrear las condiciones de temperatura.
* La construcción de un mecanismo de movimiento que puede recrear los movimientos de la barra de control y su dinámica.
* El desarrollo de un sistema integrado capaz de controlar todos los parámetros de forma sencilla.

Luego de concluida la labor de diseño del sensor, se comenzara con la etapa de producción, se prevé que en esta etapa será necesario contar además con un sistema capaz de realizar la calibración.

La calibración es el proceso de comparar los valores obtenidos por un instrumento de medición con la medida correspondiente de un patrón de referencia (o estándar) bajo condiciones específicas.

Teniendo en cuenta que el sistema a desarrollar, se encarga de generar las condiciones específicas para la calibración, se contempla el agregado de un instrumento de contraste de mayor precisión que el sensor, con el fin de incluir la capacidad de calibrar el sensor. Este punto es quien le entrega mayor importancia y lleva la vida útil del proyecto a la par de la vida útil del reactor.

De esta forma se contempla la problemática inicial de la etapa de investigación y la evolución de ésta durante la etapa de producción, se plantea entonces una solución integradora a ambas, que será durable en el tiempo, escalable y con la capacidad de adaptarse a las nuevas tecnologías a lo largo de su vida útil.

## Objetivos.

A partir de las necesidades de los investigadores, se plantearon los siguientes objetivos a tener en cuenta:

* Simulación de Temperatura:

Capacidad de realizar calentamientos lineales y gradientes de la camisa del sensor, de manera automática.

* Simulación de movimiento:

Realizar el posicionamiento de la barra de control en todo el rango de estudio, de forma repetitiva entre ensayos.

* Simulación de la dinámica de movimiento:

Controlar velocidad y aceleración de los movimientos. Realizar movimientos complejos de forma repetitiva.

* Adquisición periódica de datos:

Luego de cada ensayo se deberá disponer de los datos completos de la simulación.

* Calibración:

Capacidad de contraste un orden de magnitud por encima de la precisión deseada del sensor.

* Manejo de datos:

Contar con un sistema de almacenamiento sistematizado de los ensayos.

* Control:

Capacidad de controlar todos los aspectos del ensayo desde una PC.

# Capítulo 2: Desarrollo.

El siguiente diagrama muestra de forma sintética, los elementos componentes del sistema a implementar y su interrelación.

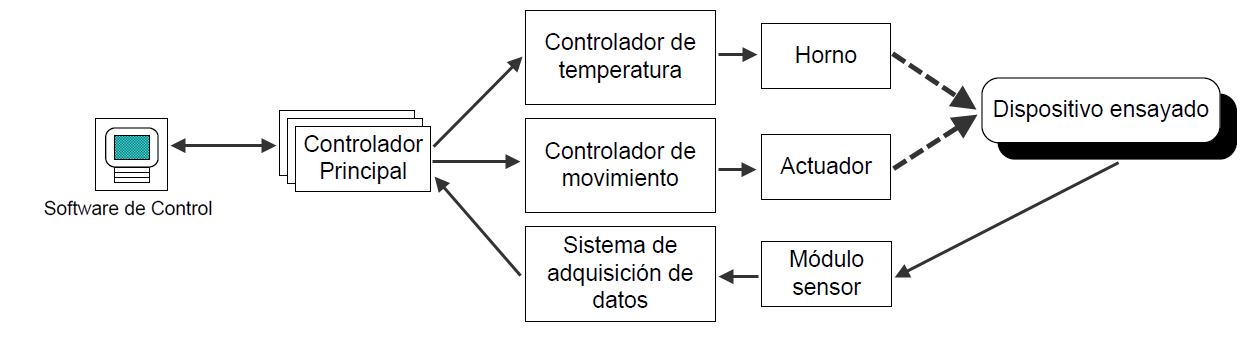


Diagrama : Bloques del sistema.



### Descripción general.

El software de control es el encargado de comunicar al usuario con el resto del sistema, el mismo se instala en una PC, y permite al usuario ingresar los parámetros del ensayo, administrar los resultados y hacerlos visibles, se encuentra comunicado directamente al controlador principal que comanda las unidades de control y de adquisición de datos.

El controlador principal es parte del hardware externo y conforma una unidad dedicada, capaz de actuar independientemente de la PC si fuera necesario. Su función principal es interpretar los parámetros de configuración ingresados por el usuario y comandar las distintas unidades para cumplir con los mismos. Se comunica con el controlador de temperatura y el de movimiento para ejecutar las acciones, y con el sistema de adquisición de datos, para verificar la correcta ejecución de las mismas a la vez que almacena los datos del ensayo para luego transmitirlos al software de control.

El controlador de temperatura es el encargado de controlar la temperatura en los distintos puntos del horno, de acuerdo a los parámetros requeridos. Consta de 6 lazos de control capaces de mantener la temperatura de cada una de las etapas del horno.

El controlador de movimiento como su nombre lo indica, se encarga de efectuar el movimiento requerido, es capaz de realizar movimientos de velocidad y aceleración controladas dentro del rango de ensayo.

El modulo sensor es quien realiza la comunicación de los sensores de movimiento, temperatura y contraste del hardware de simulación, adicionalmente se encarga de obtener los datos del sensor bajo ensayo. Se conecta directamente al sistema de adquisición de datos que administra el almacenamiento de los mismos y su disponibilidad hacia las otras unidades de control. Esta unidad es de vital importancia ya que se encarga del almacenamiento en tiempo real de los datos del ensayo.

El Horno y el Actuador junto con algunos elementos que no son visibles en el diagrama, conforman el hardware mecánico necesario para el desarrollo del ensayo.



## Controlador Principal.

### Introducción.

Como su nombre lo indica, el controlador principal conforma la unidad encargada de controlar e intercomunicar al resto de unidades del sistema. Su diseño debe contemplar los requerimientos de las unidades anexas para poder cumplir de forma correcta con su función.

Las temperaturas de funcionamiento extremas, la fuerza necesaria para realizar el movimiento y el entorno de trabajo entre otros, hacen necesario que esta unidad sea capaz de actuar independientemente de la PC en el caso de una desconexión, una falla repentina o cualquier eventualidad que pueda afectar al sistema.

### Diseño propuesto.

Para la elección del tipo de sistema a utilizar, sus componentes y distribución, han de analizarse los requerimientos del sistema. Uno de los principales parámetros a considerar para un sistema de control, es la velocidad a la que ha de reaccionar (Ogata, 2010), en este caso serán los tiempos de los sistemas de control de temperatura, posición y adquisición los que condicionen este factor.

* Tiempo de acción del control de temperatura = 10[s].
* Tiempo de acción del controlador de posición = 100[us].
* Tiempo de acción del controlador de muestreo = 1[ms].

Los tiempos de acción resultan variados, podemos dejar de lado el control de temperatura, sin embargo el controlador de posición requiere de 10000 actuaciones por segundo, esto limita los dispositivos capaces de realizar la tarea, se suma a esto la rigidez de tiempos que implica el control necesario para emular la dinámica de movimiento y la cantidad de cálculos que este implica.

Otra variable a considerar a la hora de seleccionar el sistema surge del tiempo de acción del controlador de muestreo y la necesidad de un comportamiento independiente del sistema. Analizando el volumen de datos sobre la base de una experiencia de 24hs, un tamaño de muestra de 52 bytes y una frecuencia de muestreo de 50Hz, se obtiene:

Esta capacidad de almacenamiento ha de ser considerada, y a la vez con esto se asocia la capacidad de transferencia de datos una vez concluida la experiencia.

El protocolo de comunicaciones a utilizar, por características de diseño requeridas es MODBUS, la capacidad de transferencia de este protocolo viene dada por la velocidad de la interfaz base de comunicaciones, esta interfaz a su vez determina el hardware de comunicaciones requerido.

El protocolo MODBUS esta normalizado sobre dos interfaces de comunicaciones muy populares, la interfaz serie (RS-232, RS-485, etc.) y la interfaz Ethernet, dada la complejidad y el costo de la segunda, analizaremos las prestaciones de la primer opción.

El principal factor es la velocidad de transferencia de datos y con esto el tiempo necesario para la transmisión de los 214,4[MB] de información, sobre una velocidad de 115200 baudios, el tiempo de transferencia, incluyendo el uso de datos propio del protocolo, obtenemos:

La velocidad de transferencia es demasiado baja, de forma que será necesaria una interfaz Ethernet.

Se analizaron dos opciones para el diseño del controlador principal y los módulos de control:

* + - 1. Implementar sobre un mismo controlador, los cuatro módulos componentes del sistema, y mediante el uso de un sistema operativo de tiempo real, administrar el tiempo necesario para cada tarea de forma de lograr el objetivo deseado.
      2. Utilizar múltiples controladores, cada uno con su función específica, interconectados de forma de tener un control continuo sobre cada unidad sin necesidad de sistemas de administración del tiempo de proceso.

La primera opción resulta de difícil implementación en firmware, mientras que minimizaría el hardware necesario a solo un controlador, reduciendo el costo de montaje y diseño. La segunda opción por su parte, garantiza un control determinístico y fiable, robustez ante eventos externos y una relativa simplicidad del firmware mientras que cuadriplica la cantidad de hardware necesario.

Luego de analizar el hardware disponible para la implementación de cada alternativa, se decidió el uso de un dispositivo FPGA, que permite la integración de los cuatro procesadores a medida y dentro de un solo dispositivo de silicio, esto además brinda una velocidad de intercomunicación muy alta sin la necesidad de buses de comunicación cableados o complicados diseños de hardware.

La empresa Digilent ofrece un hardware basado en un dispositivo FPGA, que provee además una memoria DDR 3 de la capacidad suficiente y un módulo de comunicación Ethernet, la placa de desarrollo ARTY es la seleccionada para el montaje del hardware.

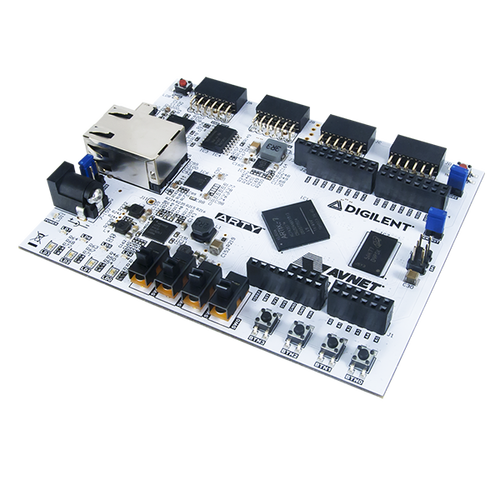


Figura : Placa de desarrollo ARTY.

Las características más sobresalientes de ARTY son:

* Xilinx Artix-35T FPGA (xc7a35ticsg324-1L)

5,200 slices (cada uno contiene 6 LUTs y 8 flip-flops).

1,800 Kb de memoria RAM de bloque rápida.

Cinco unidades de manejo de reloj, cada una con un PLL.

90 slices DSP.

Internal clock speeds exceeding 450MHz.

Programable por JTAG y por Flash.

* Sistema

DDR3L de 256MB de 16 bits @ 667MHz.

Quad-SPI Flash de 16MB.

Programador USB-JTAG.

Alimentación por USB o cualquier fuente de poder de entre 7V y 15V.

* Conectividad

Ethernet 10/100 Mbps

Puente USB-UART

Bus I2C

4 conectores Pmod

Como fuente de alimentación se utilizó una fuente switching de 9[V] 2[A], su reducido tamaño, muy bajo costo y prestaciones, la hacen ideal para este uso.



Figura : Fuente switching 9[V].

El controlador principal se encarga además del control del tablero de comando que funciona como interfaz HMI de la máquina, independiente del software de la PC.

Dentro de los elementos principales del tablero de comando se encuentran:

* 1 Pulsador de encendido.
* 1 Pulsador de apagado.
* Parada de emergencia.
* 4 Leds indicadores de estado general.
* 5 Leds bi-color indicadores del estado de los elementos.

Se destaca la inclusión de la parada de emergencia, dadas las temperaturas y el rango de posiciones que se controlan, resulta indispensable colocar un elemento de fácil acceso capaz de detener por completo el funcionamiento de toda la maquina independientemente del estado de esta.

El conjunto de las señales correspondientes al tablero de comando se conectan mediante un puerto PMOD a la placa de desarrollo ARTY, junto a este hardware se diseñó un controlador dentro del FPGA para realizar la conexión al bus AXI, se detallara más adelante en este mismo capítulo su funcionamiento.

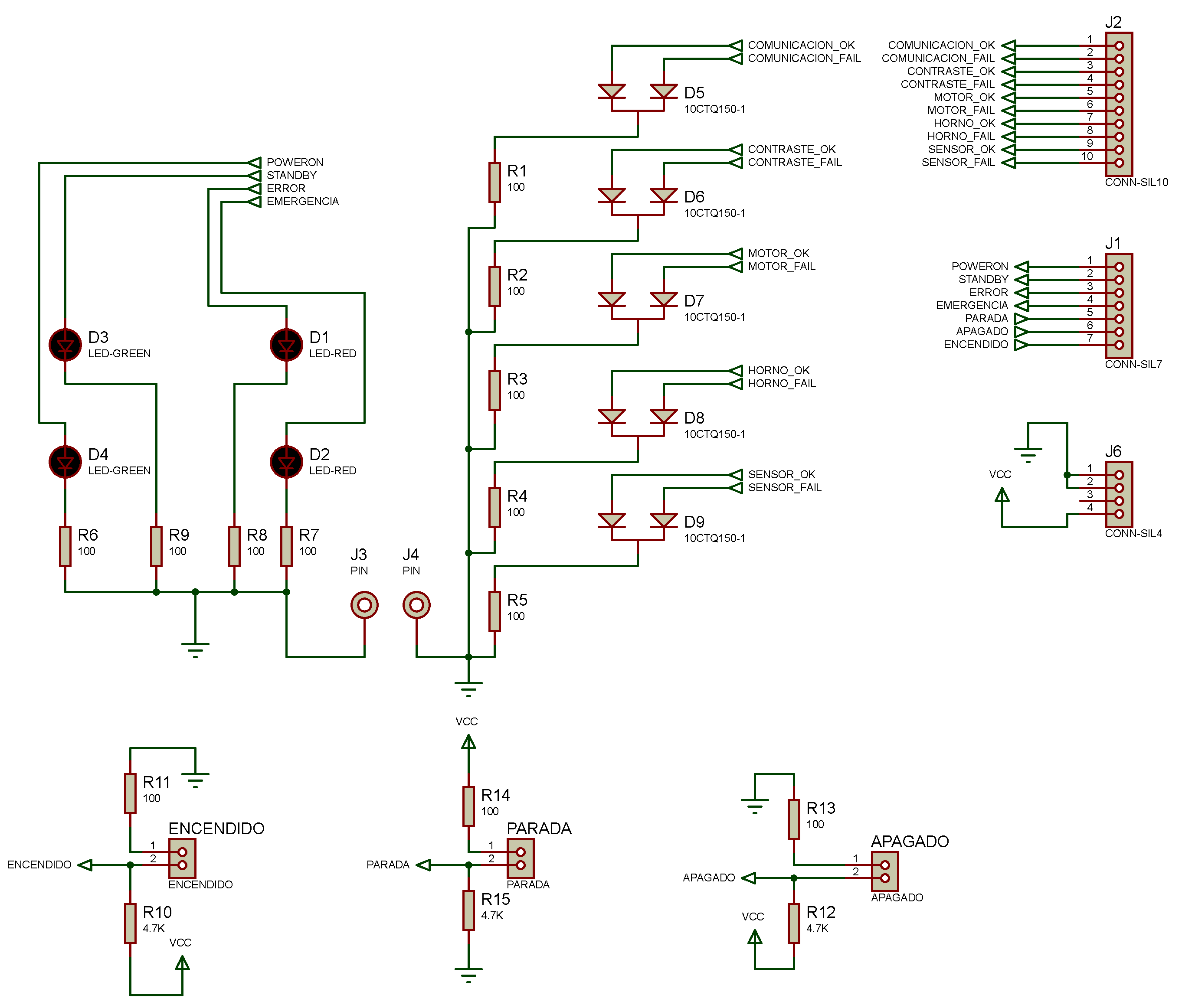


Figura : Circuito tablero de comando.

El resto del hardware se implementa dentro del dispositivo FPGA, cabe recordar que en el nivel más alto, los FPGAs son chips de silicio reprogramables, utilizan bloques de lógica pre-construidos y recursos de ruteo programables, para implementar funcionalidades personalizadas en hardware, son completamente reconfigurables y al instante toman una nueva “personalidad” cuando se les compila una diferente configuración de circuitos.

Los FPGAs combinan lo mejor de los ASICs y de los sistemas basados en procesadores. Ofrecen velocidades temporizadas por hardware y fiabilidad, pero sin requerir altos volúmenes de recursos para compensar el gran gasto que genera un diseño personalizado de ASIC. El silicio reprogramable tiene la misma capacidad de ajustarse que un software que se ejecuta en un sistema basado en procesadores, pero no está limitado por el número de núcleos disponibles. A diferencia de los procesadores, los FPGAs llevan a cabo diferentes operaciones de manera paralela, por lo que éstas no necesitan competir por los mismos recursos. Cada tarea de procesos independientes se asigna a una sección dedicada del chip, y puede ejecutarse de manera autónoma sin ser afectada por otros bloques de lógica. Como resultado, el rendimiento de una parte de la aplicación no se ve afectado cuando se agregan otros procesos.

Dentro de sus características más destacables se encuentran:

* Rendimiento
* Tiempo en Llegar al Mercado
* Precio
* Fiabilidad
* Mantenimiento a Largo Plazo

Para el diseño del hardware primeramente se debe seleccionar que tipo de procesador utilizar, se tiene en cuenta el costo del IP(Intelectual Property) del procesador, sus prestaciones, el tipo de bus de interconexión y las plataformas de software para su posterior programación.

En el mercado hay una amplia variedad de IP’s de distintas empresas, entre las que se destaca ARM, Intel, Altera y Xilinx, con una gran variedad de modelos y precios, las prestaciones son similares para cada proveedor, se busca entonces el modelo con una interfaz que ofrezca amplia compatibilidad y con la plataforma de desarrollo de firmware más adecuada.

Las prestaciones del controlador MicroBlaze, junto a su compatibilidad con los dispositivos de Xilinx, su plataforma de desarrollo de firmware muy documentada y con más de 8 años de trayectoria lo hacen adecuado para su uso en esta aplicación.

El procesador embebido MicroBlaze es un computador con un set de instrucciones reducido (RISC) optimizado para su implementación en FPGAs Xilinx, un diagrama de bloques del mismo se observa en el Diagrama 2.

Es altamente configurable permitiendo la selección de los parámetros específicos para cada aplicación, sus características estáticas incluyen:

* Registros de propósito general de 32 bits.
* Instrucciones de 32 bits con tres operadores y dos modos de direccionamiento.
* Bus de direccionamiento de 32 bits.
* Bus de interconexiones AXI.

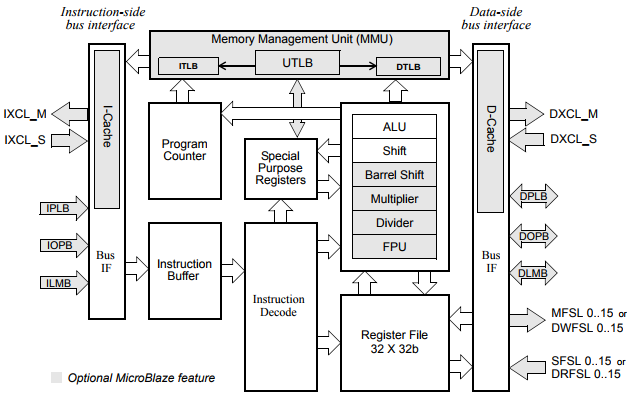


Diagrama : Controlador Soft-Core MicroBlaze.

Mientras que dentro de las características opcionales utilizadas se destacan:

* Cache de instrucciones configurable.
* Cache de datos configurable.
* Multiplicador de 32 y 64 bits en hardware.
* Divisor de 32 y 64 bits en hardware.
* Unidad de punto flotante en hardware.
* Lógica de debug por hardware.

Adicionalmente se utilizaron IPs de Xilinx compatibles con bus AXI, sus detalles y características resultan extensos, se mencionaran a continuación los más importantes para no abundar innecesariamente en detalles.

* Controlador de interrupciones programable AXI

Controlador de hasta 256 interrupciones, priorizables por grupo o individualmente.

* Reloj programable AXI

Reloj doble con registros de 32 y 64 bits, programable de hasta 400MHz.

* UART Lite AXI

UART liviana de velocidad fija.

* UART 16550 AXI

Uart de velocidad, datos, paridad y parada configurables.

* Ethernet Lite AXI

Bus de comunicaciones Ethernet, liviano adecuado al hardware.

* Controlador QUAD SPI AXI

Controlador de dispositivos Quad SPI

* Interconectador AXI
* Interfaz de memoria DDR AXI

Bloque de interfaz de memoria DDR3, lectura escritura y refresco.

* Unidad de manejo de reloj

Conversor de señal de reloj con multiplicador y divisor, hasta 2,2GHz.

Con la utilización de estas unidades, se propone el siguiente diseño.

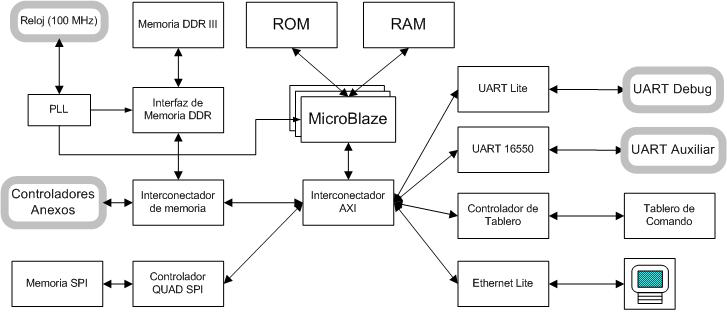


Diagrama : Controlador principal.

En la imagen se puede observar, en el centro el procesador, a la derecha los componentes de comunicación con el exterior y a la izquierda, los elementos de intercomunicación internos hacia el resto de los controladores.

Como se mencionó anteriormente, el procesador MicroBlaze, puede ajustarse a las necesidades del diseño, se puede configurar la cantidad de memoria de programa, memoria de datos, caches, interrupciones y unidades internas de computo, sin embargo cada parámetro adicional tiene un coste en hardware, que es limitado según la capacidad del FPGA utilizado.

En este caso la cantidad de memoria interna disponible en el FPGA para la asignación de RAM y ROM necesarias para la ejecución del firmware no es suficiente, entonces se selecciona el uso de una caché de 4KB para la memoria de programa y se asocia el mismo a la DDR III, de esta forma se puede almacenar el programa en la memoria DDR. Sin embargo esta memoria es volátil, por lo que se coloca el controlador SPI para acceder a la memoria FLASH y en el arranque descargar el programa y transferirlo hacia la memoria DDR.

Dentro de los periféricos exteriores podemos ver dos UART’s, la primera se coloca junto con un chip FTDI que hace de puente USB y permite el debug del procesador, es una conexión exclusiva de configuración que no será accesible al usuario, solo se usa para testeo y configuración durante el proceso de instalación.

La UART auxiliar como su nombre lo indica, corresponde a un enlace alternativo al Ethernet para la comunicación mediante protocolo MODBUS, desde este enlace, podrán hacerse las configuraciones iniciales del protocolo, necesarias para la conexión Ethernet.

El módulo Ethernet es el encargado de interpretar y controlar el hardware correspondiente a esta conexión, es un controlador propio de Xilinx y esta hermanado al hardware del fabricante Texas Instruments DP83848J.

Por último el controlador del tablero de comando corresponde al módulo de control necesario para administrar las funciones propias del tablero y generar la interfaz AXI necesaria para comunicar el mismo con el bus de datos del controlador. Este fue desarrollado ad hoc en código VHDL, hacia el exterior, cuenta con 17 puertos de E/S encargados de censar el estado de los botones y controlar los leds, mientras que hacia el bus AXI cuenta con un solo registro de escritura – lectura que se encarga de controlar el funcionamiento.

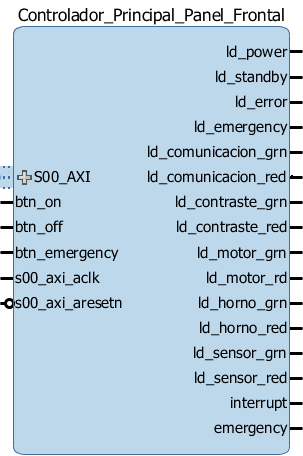


Figura : Controlador del tablero de comando.

Dentro de las características a destacar se encuentra el funcionamiento del encendido y apagado controlado por interrupciones a fin de evitar un funcionamiento tipo pooling del procesador, además de una interrupción dedicada para la parada de emergencia. Adicionalmente se agregaron dos relojes para controlar el tiempo de presionado del botón de apagado y encendido, siendo estos controlables por software entre 1 y 7 segundos.

### Problemas y soluciones implementadas.

Los principales problemas que se presentan en los sistemas multi procesador, donde estos trabajan de forma independiente, es el uso de los recursos compartidos, en particular de la memoria y la comunicación de las unidades (ACM/IEEE 1993).

Se analizaron diversos métodos de comunicación entre los controladores, interfaces SPI, interfaces paralelas, comunicaciones tipo serie y otros, sin embargo, todos estos métodos requieren del uso de tiempo de procesamiento para el arbitraje de la comunicación, esto resulta en un inconveniente a la hora de desarrollar el firmware y puede ralentizar el funcionamiento del sistema. Un método que no requiere de la intervención del procesador en cada transacción de información es la interconexión mediante el uso de memoria compartida, esto resulta sencillo para conectar dos dispositivos, se utiliza una memoria de doble puerto y se resuelve rápidamente, sin embargo para más de dos es más complicado.

El método utilizado se basa en la utilización de un módulo interconectador de bus AXI, con múltiples entradas tipo esclavo y una salida tipo maestro de bus, este módulo se encarga del balance de la carga de transferencia de datos de forma automática, también establece la prioridad y evita las operaciones de escritura superpuesta o de lectura escritura simultánea. Su uso resulta valido para este tipo de sistemas, donde la carga de transferencia es desbalanceada y no se responde a una secuencia de acceso (Mori et al. 1993).

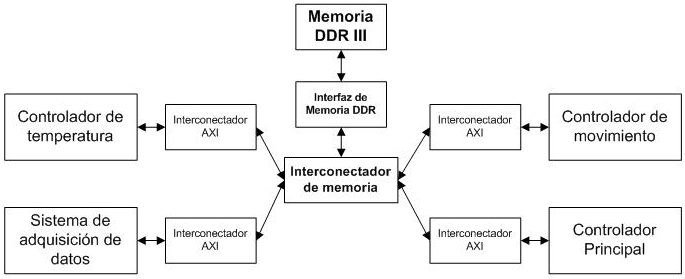


Diagrama : Memoria de uso compartido.

La implementación del sistema mencionado anteriormente dio excelentes resultados para la comunicación entre los procesadores, se mapeo la memoria DDR dentro de cada una de las tablas de direcciones de los procesadores de forma que cada uno accede a los datos de los anexos como a sus propias direcciones de memoria. Esto requiere un especial cuidado a la hora de asignar el uso de cada parte de la memoria, se establecieron límites para el uso privado de cada uno de los procesadores, un espacio de uso común a todos y por último el espacio necesario para almacenar los datos de la simulación de forma segura hasta ser volcados a la PC.

A pesar de estos cuidados, el procesador principal no se comportó de la forma esperada, se verificaban retardos en la escritura de las zonas de memoria que le correspondían además de operaciones de lectura erróneas, luego de varias pruebas se pudo determinar que esto se debe a la cache de datos interna del procesador, fue necesario entonces desactivar el uso de la cache para las secciones de memoria compartida, luego de esto el funcionamiento fue el esperado.

### Desarrollo del firmware.

El firmware del controlador principal, se divide en dos grandes secciones que funcionan de forma independiente, la primera es la comunicación con la PC junto con la implementación del protocolo MODBUS mientras que la segunda comprende el control del resto de las unidades de control. La configuración y el muestreo.

Como se mencionó en la sección anterior, la comunicación con la PC se realiza mediante el protocolo Ethernet, este tipo de comunicación si bien es muy común en el mercado informático, ha comenzado a ganar terreno en la industria y particularmente en el control hace solo algunos años, su implementación es costosa y depende de una capa de hardware de elevada complejidad de diseño y montaje, además de una importante cantidad de firmware para su correcto funcionamiento. Para acelerar el proceso de desarrollo del firmware se utiliza la librería integrada LwIP.

La librería lwIP es una pequeña implementación independiente del protocolo TCP/IP que ha sido desarrollada por Adam Dunkels, en el laboratorio de arquitecturas de red y computadoras, en el instituto sueco de ciencias de la computación. El objetivo de la implementación TCP/IP de LwIP está centrado en reducir la cantidad de RAM utilizada mientras se mantiene un protocolo TCP completo. Esto la hace indicada para su uso en sistemas embebidos con apenas unas decenas de kilobytes de RAM y alrededor de 40Kb de ROM disponibles. Adicionalmente esta librería es de licencia BSD, lo que la hace prácticamente libre al público, con mínimas restricciones.

A continuación se detalla mediante el esquema 5, el funcionamiento del firmware encargado de la comunicación con el software de la PC, la lectura se realiza en el menor tiempo posible que es resultado de las tareas que este ejecutando el controlador, dadas las características del protocolo, se ha diseñado de forma de no superar el tiempo máximo de respuesta establecido, además se utilizan las funciones nativas para mantener la compatibilidad, no se adiciona ninguna función propia.

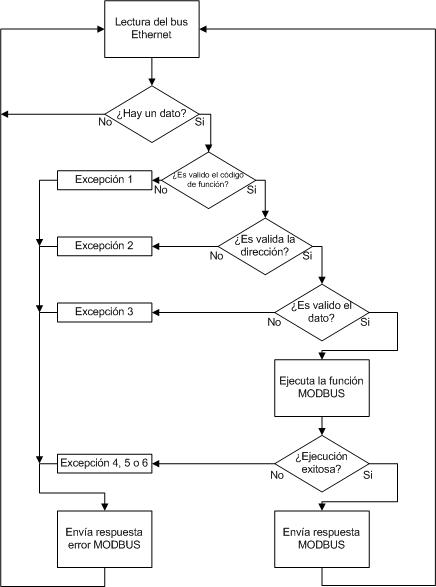


Diagrama : Firmware MODBUS Ethernet.

Se implementan para el protocolo las funciones:

* 3 – Lectura múltiple de registros, permite leer hasta 122 registros de 16 bits, consecutivos.
* 6 – Escritura simple de registro, permite la escritura de un solo registro de 16 bits.
* 16 – Escritura múltiple de registros, permite la escritura de hasta 123 registros consecutivos.

Los sistemas basados en dispositivos FPGA, presentan una demora en el arranque que se debe al tiempo de configuración del integrado, mientras que en este caso particular se adiciona el tiempo de carga del firmware a la memoria RAM, el conjunto acumula aproximadamente 14 segundos. Este tiempo de espera resulta molesto si fuera necesario a cada encendido, se diseña el sistema de forma de mantenerlo en un estado “Stand-By” luego de ser conectado a la red eléctrica. Los controladores secundarios se mantienen apagados mientras que el principal se encarga de establecer las condiciones para el encendido del sistema, luego de presionado el botón de encendido, realiza una secuencia de arranque de las unidades y a la vez de diagnóstico del sistema, este tiene que ser rigurosamente correcto para que el sistema pueda arrancar, el orden y las condiciones son:

* Encendido de las comunicaciones.
* Encendido del controlador de temperatura.
  + Comunicaciones con el horno.
  + Estado del horno.
  + Inicialización del control.
* Encendido del controlador de movimiento.
  + Arranque del motor.
  + Estado del motor.
  + Puesta a 0.
* Encendido del controlador de muestreo.
  + Seteo del sensor de contraste.
  + Conexión con el sensor bajo prueba.
  + Estado del sensor bajo prueba.

Luego de realizado el procedimiento, el sistema entra en un bucle a la espera de un comando por parte de la PC, el orden normal es la carga de datos de simulación, luego la ejecución de la simulación y por último la descarga de los resultados, adicionalmente aun ante una falla se puede realizar una configuración si fuera necesario para el correcto funcionamiento del sistema. El diagrama 6 ilustra el funcionamiento descripto anteriormente.

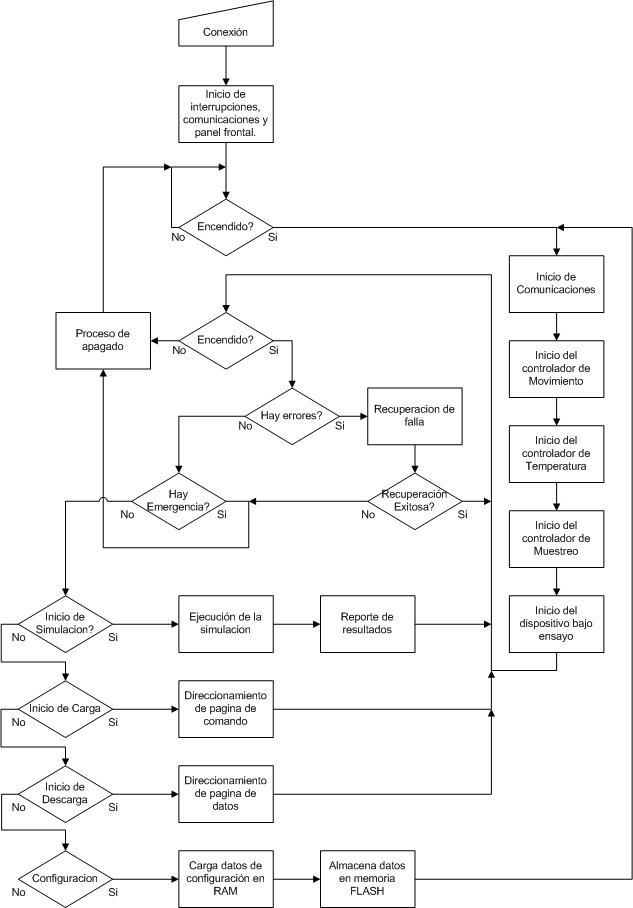


Diagrama : Firmware controlador principal.

### Resultados parciales.

Los resultados correspondientes a esta unidad no se pueden evaluar sin estar el conjunto completo en funcionamiento, sin embargo se pudo establecer de forma previa el funcionamiento correcto de las comunicaciones, con un tiempo de transferencia de entre 6[ms] y 18[ms] por paquete MODBUS, superando ampliamente las expectativas. Se utilizaron softwares dedicados al diseño de dispositivos MODBUS para el testeo y la generación de errores, ante todas las pruebas el comportamiento fue el deseado. Se encontraron fallas de compatibilidad en la negociación de velocidades automática con algunos modelos de enrutadores, por esto se decidió fijar la velocidad por defecto a 100MB/s con la posibilidad de elegir 10MB/s mediante parámetros de configuración.

Además se comprobó con éxito la escritura de las configuraciones en la memoria FLASH, el correcto funcionamiento del panel de control y la escritura y lectura de la memoria compartida.

## Controlador de Temperatura.

### Introducción.

Dentro de los parámetros de funcionamiento de los dispositivos de medición existentes en la industria nuclear, suele ser primordial la temperatura de funcionamiento. Cualquier dispositivo electrónico que se comercialice para estos fines debe contar con un amplio rango de operación y para garantizar este rango se deberá testear el dispositivo de acuerdo a las necesidades de uso del mismo, e incluso más allá de estas, a fin de garantizar la robustez y determinar el comportamiento en situaciones excepcionales.

La unidad de simulación de temperatura, es el elemento encargado de realizar el calentamiento del sensor bajo ensayo, dentro de los rangos establecidos por las condiciones de funcionamiento del dispositivo.

Para este caso en particular, el rango de temperatura comienza en 240 °C y se extiende hasta los 520 °C, mientras que la mínima variación medible, se indicó según los requerimientos en 2,5°C.

Las pruebas experimentales teóricas y prácticas, que se han desarrollado a lo largo de la etapa de diseño del sensor, indican que la temperatura es el factor principal en la variación de las características de medición del mismo [2]. Por lo que resulta de suma importancia su correcto control y medición a lo largo del proceso de ensayo y calibración.

### Diseño propuesto.

### Hardware Electrónico.

De forma similar al diseño realizado para el controlador principal, expuesto en el Diagrama 3, se utilizó como procesador el mencionado MicroBlaze y la estructura de conexión de la memoria también expuesta anteriormente. Se agregan algunos aspectos propios de este controlador, como una salida necesaria para el encendido del horno y el driver necesario para la conexión mediante RS-485 de los componentes de censado y control del horno.

Como se mencionó anteriormente, el procesador MicroBlaze, puede ajustarse a las necesidades del diseño, se puede configurar la cantidad de memoria de programa, memoria de datos, caches, interrupciones y unidades internas de computo, el peso de computo que ha de afrontar este controlador hace necesario un espacio de memoria RAM de 65Kb, además se le adicionan los módulos de multiplicación por hardware de 64 bits, un divisor por hardware de 32bits y una unidad de punto flotante, de esta forma se consigue un código más compacto y un tiempo de ejecución de programa adecuado a la velocidad de control necesaria del horno.

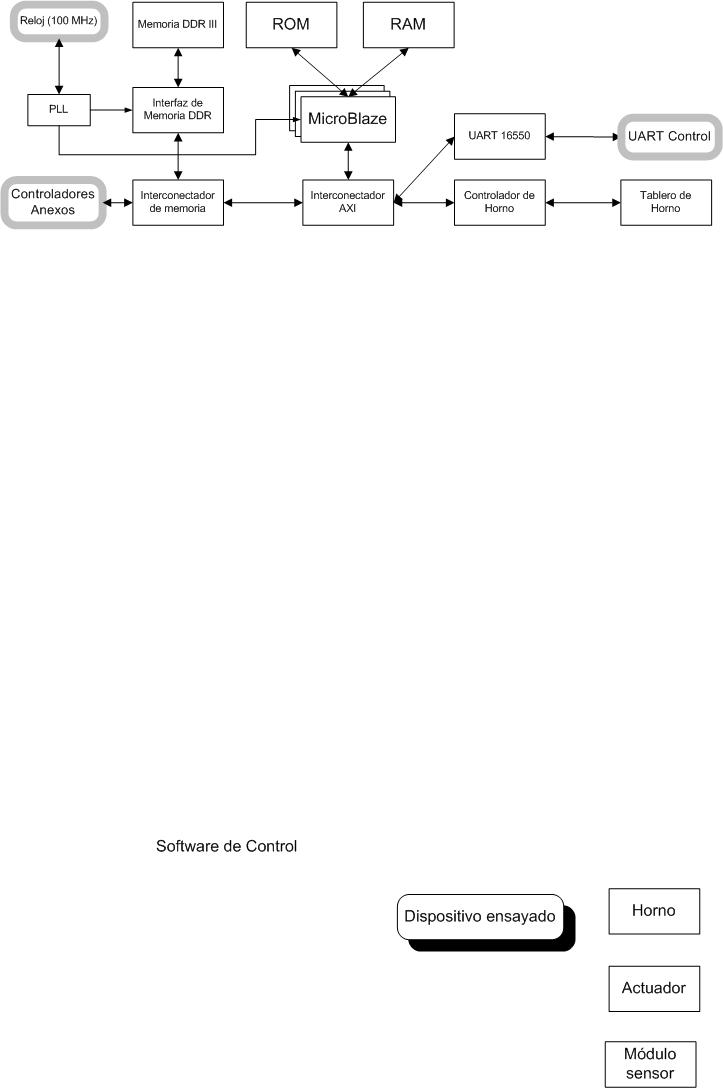


Diagrama : Controlador de temperatura.

### Hardware Mecánico.

Teniendo en cuenta el amplio rango de variación de temperatura, la sensibilidad requerida y la importancia del correcto funcionamiento de esta etapa, resulta indispensable un cuidadoso diseño de la estructura mecánica, acorde a las características y dimensiones del elemento en desarrollo.

Para el esquema del elemento calefactor, se tuvieron en cuenta los parámetros de funcionamiento, las dimensiones y las características del entorno.

Dimensiones del elemento:

* Longitud 1600[mm]
* Diámetro máximo 145[mm]
* Peso 152 [Kg]
* Diámetro del elemento de salida 36[mm]

Parámetros de funcionamiento:

* Temperatura mínima de trabajo 240[°C]
* Temperatura mínima de ensayo 220[°C]
* Temperatura normal de trabajo 320[°C]
* Temperatura máxima de trabajo 460[°C]
* Temperatura máxima de ensayo 520[°C]
* Diferencia máxima de temperatura entre extremos 60[°C]
* Pico máximo de variación de temperatura 5[°C/min]

Características del entorno:

* Distancia al elemento lateral más próximo [60]mm
* Espesor de sujeción superior 60[mm]
* Predominancia de elementos no magnéticos.

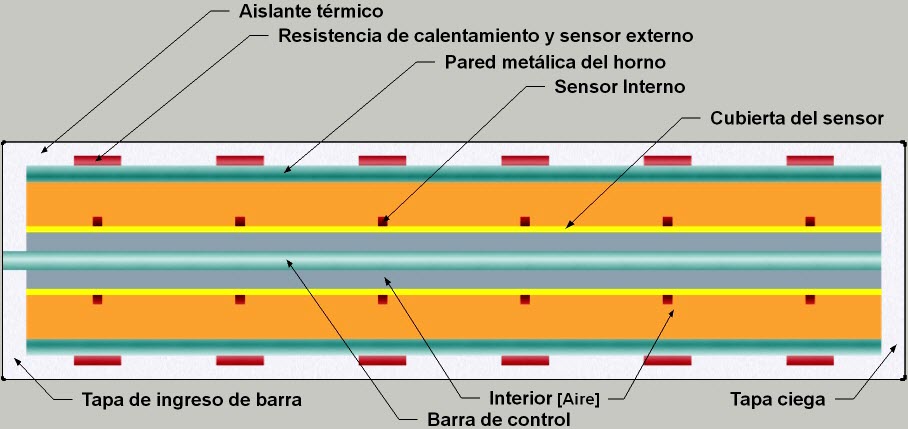


Figura : Diseño de horno.

Puede observarse en la figura 5, el croquis del horno de calentamiento del sensor, la pieza fundamental del mismo es un cilindro de acero inoxidable de aproximadamente 420mm de diámetro y 50mm de espesor de pared, que posee las mismas que las paredes de la cámara del reactor CAREM 25.

Sobre este cilindro se montan 6 elementos calefactores del tipo resistencia de abrazadera, su potencia y formato de montaje las hacen adecuadas para su uso particular. Son capaces de entregar 2000W cada una y llegan de forma segura hasta una temperatura de 600 [°C].



Figura : Resistencia tipo abrazadera.

La potencia de los elementos calefactores hace necesario para su control el uso de conmutadores de potencia, pudiendo estos ser; contactores electromecanicos o relés de estado sólido, por su durabilidad, ausencia de transitorio de conexión y desconexión, velocidad y menor tiempo de activación, se eligió el uso de relés de estado sólido, son capaces de manejar hasta 40[A] con un retardo inferior a un ciclo y un tiempo de conexión mínimo de 2 ciclos.



Figura : Relé de estado sólido.

Para medir las temperaturas en el exterior e interior del horno se utilizaron termocuplas tipo J, construidas a medida a partir de un cable especial para este fin. El dispositivo encargado de interpretar las señales de la termocuplas es la unidad de adquisición de datos ADAM 4018+, permite la conexión de hasta 7 sensores de variados tipos, su configuración y la comunicación de los resultados a través de protocolo MODBUS serie.



Figura : UAD ADAM4018+

Por ultimo para el control de la temperatura de las resistencias se considera el uso de controladores PID, en este caso por su costo y flexibilidad se eligen los controladores DTA4848, de la marca Delta.



Figura : Controlador PID DTA4848

Teniendo en cuenta el peso de la estructura y el costo de transporte, se decidió llevar a cabo el montaje en el lugar de destino del aparato, en este caso, el Centro Atómico Bariloche. La construcción de las partes metálicas fue llevada a cabo por profesionales del mismo centro, mientras que el montaje de los elementos de medición, control excitación de las resistencias fue llevado a cabo por nosotros.

Contando con el hardware, se procede al montaje del sistema de control que se encargará de operar las resistencias de calentamiento a fin de obtener la temperatura deseada sobre la superficie del elemento en desarrollo.

Primeramente se realizó el montaje entre los controladores DTA4848 y las resistencias tipo abrazaderas descriptas anteriormente, luego se cablearon los sensores de temperatura correspondientes y la etapa de actuación de potencia a los mismos controladores.

### Simulador del sistema.

Debido a los amplios tiempos de calentamiento y la gran cantidad de energía que consume cada experiencia, se decidió implementar un simulador del elemento de calefacción, esto permite realizar ensayos más rápidos además de brindar la posibilidad de comprobar los parámetros obtenidos mediante la identificación del sistema realizado anteriormente.



Figura : Software de simulación de horno

El simulador se basa en una serie de 6 modelos idénticos que identifican cada etapa del horno, cada modelo consta de 4 sistemas de primer orden que interactúan, el sistema principal corresponde al calentamiento de la resistencia por acción del PID, mientras que dos sistemas anexos simulan las transferencias de calor derecha e izquierda correspondientes. Un último sistema simula la transferencia desde el borde interior de la pared del horno hasta la camisa donde está ubicado el sensor. Todos los parámetros de estos sistemas se pueden variar para cada etapa del horno. Luego de la identificación y volcado de los parámetros se obtuvo un funcionamiento idéntico en un 98%, lo cual corresponde a un valor más que aceptable para hacer pruebas sin recurrir al costoso calentamiento del elemento real. Para completar el funcionamiento, se emulan los 6 PIDs junto con el módulo de adquisición de datos y la comunicación de estos mediante un adaptador USB a RS485.

### Problemas y soluciones implementadas.

Las primeras pruebas del montaje según se había proyectado, dieron como resultado un calentamiento uniforme en el exterior, esto es coincidente con el comportamiento esperado del controlador PID, los tiempos de establecimiento fueron aceptables, pero se midieron grandes variaciones de temperatura en el centro del horno.

La siguiente imagen térmica refleja el resultado de una experiencia de prueba con una temperatura deseada para la camisa del sensor de 240[°C] y un tiempo de 4000[s] de duración. De inmediato se aprecia una temperatura inferior a la deseada en ambos extremos del sensor de alrededor de 20[°C] y una temperatura superior en el centro de similar magnitud.

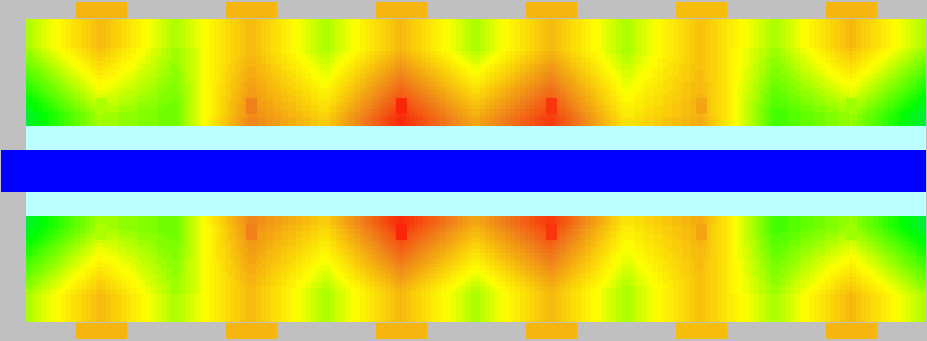


Figura : Primer prueba de horno

Se acusa de la primera irregularidad a la pérdida del horno en los extremos, y de la segunda a la acumulación de calor en el centro. Estos factores sumados a la lejanía de las tomas de información de los controladores respecto del punto a controlar, contribuyen a la dispersión medida.

Para subsanar el error y en vista de que el principal causante era la distancia entre el punto a controlar y el punto de toma de datos, la opción a seguir fue la colocación de seis sensores tipo abrazadera sobre el punto de toma de datos y el posterior cableado de estos a los controladores PID.

La imagen a continuación muestra el resultado de una experiencia igual a la anterior con los cambios realizados.

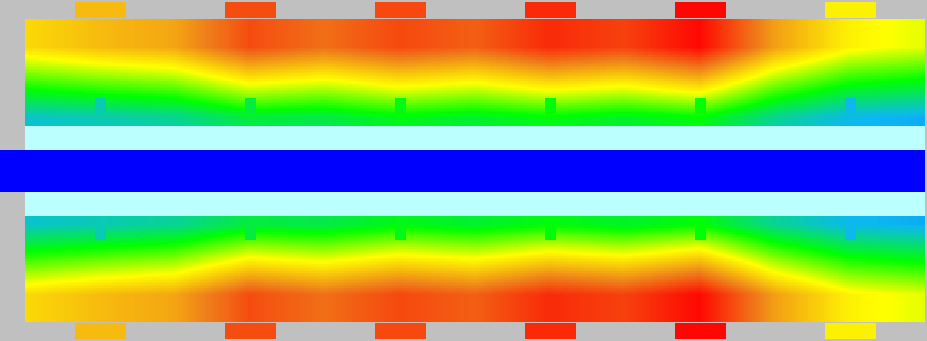


Figura : Segunda prueba de horno

En este caso, no se pudo llegar a la estabilización de la temperatura en los puntos de control debido a un sobrecalentamiento de las resistencias. Este hecho se debe a la gran inercia térmica que presenta el sistema, lo cual se traduce en una sobreactuación de los controladores que aun calibrados, no son capaces de compensar este parámetro del sistema sin llevar más allá de su capacidad de funcionamiento a las resistencias.

### Control en cascada.

Luego de los intentos fallidos anteriores se analizaron las posibles soluciones de control según los métodos tradicionales. Para ambas configuraciones se utilizaron parámetros de calibración de Zigler-Nichols (Adam 2011).y de Cohen-Coon (Ogata 2011).en sus distintas versiones, con resultados negativos para ambas técnicas.

Se analizó entonces la posibilidad de implementar un lazo de control en cascada, en esta configuración la salida de un controlador de realimentación es el punto de ajuste para otro controlador de realimentación, por lo menos. Este control involucra sistemas de control que estén ordenados uno dentro del otro.

Presenta las siguientes superioridades respecto de un control tradicional (Adam 2011).

* Para perturbaciones del lazo interno, los sobrevalores se reducen hasta una décima parte y la integral del error hasta 100 veces.
* Para perturbaciones del lazo externo, los sobrevalores se reducen hasta la mitad y la integral del error hasta 6 veces.
* Eliminar el efecto de algunas perturbaciones haciendo la respuesta de regulación del sistema más estable y más rápido.
* La variable de control alcanza más rápidamente el valor de consigna.
* Mejorar la dinámica del lazo de control.

La primer y última característica lo hace adecuado para corregir los problemas planteados anteriormente.

La estructura de control en cascada tiene dos lazos, un lazo primario con un controlador primario también llamado “maestro” y un lazo secundario con un controlador secundario también denominado “esclavo”, siendo la salida del primario el punto de consigna del controlador secundario y la salida de este la que actúa sobre el proceso.

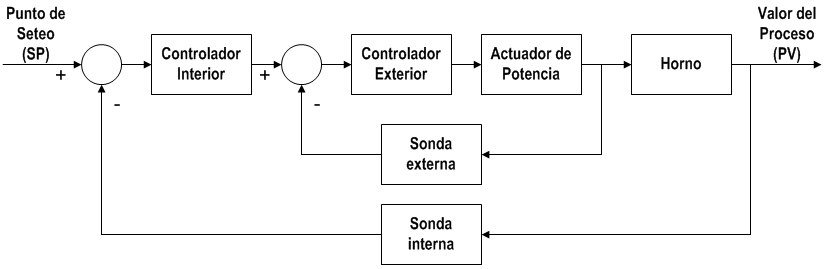


Diagrama : Control en cascada.

En la gráfica anterior se detalla la estructura a utilizar, para el caso del controlador exterior o esclavo, se utilizara el PID existente DTA4848, mientras que el controlador interior o maestro, será implementado por firmware en conjunto con el sistema de control de temperatura.

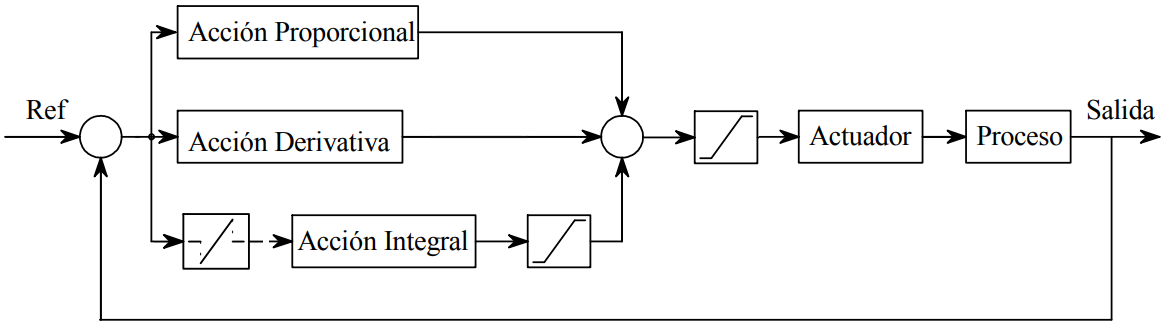


Diagrama : Wind-Up Integral

### Calibración del sistema.

La sintonía de los dos reguladores se efectúa, igual que en controladores en configuración simple pero en dos etapas

1. Obtener un modelo de la parte del proceso incluida en el lazo secundario (modelo de conocimiento o modelo experimental. Sintonizar el controlador secundario por algún método conocido.
2. Obtener un modelo del proceso del bucle secundario con el controlador primario ya ajustado, para luego sintonizar el controlador primario por alguno de los métodos conocidos.

Se han tenido en cuenta las consideraciones de diseño establecidas por (Adam 2011), en las que se detalla para este tipo de controladores:

* El lazo interno se lo diseña ajustándolo muy estrechamente (Es decir, con un valor de ganancia tan alto como sea posible) para dar una respuesta rápida en dicho lazo.
* Luego, el lazo externo es ajustado con el lazo interno en operación o en modo automático mediante algún método clásico de ajuste, (Ziegler-Nichols, Cohen-Coon, entre otros)

Dadas las características del sistema, se decidió la utilizar el método de la curva de reacción de proceso para realizar el ajuste del controlador, esta opción resulta más adecuada que otros métodos como el de oscilaciones sostenidas, ya que no llevan al sistema a sus puntos críticos de operación. A partir de su aplicación se obtuvieron los siguientes parámetros para cada etapa secundaria del horno.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Etapa | Dalay[s] | T[s] | Kk[°C] | S |
| 1 | 64 | 2260 | 442 | 0,000196 |
| 2 | 68 | 2120 | 447 | 0,000211 |
| 3 | 74 | 2150 | 453 | 0,000211 |
| 4 | 72 | 2180 | 451 | 0,000207 |
| 5 | 66 | 2130 | 449 | 0,000211 |
| 6 | 62 | 2240 | 439 | 0,000196 |

Tabla : Parámetros sistema externo del horno.

Con estas se calcularon los valores de ajuste según dos métodos.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Etapa | Kr | Ti | Td | | 1 | 95,9 | 128,0 | 32,0 | | 2 | 83,7 | 136,0 | 34,0 | | 3 | 77,0 | 148,0 | 37,0 | | 4 | 80,6 | 144,0 | 36,0 | | 5 | 86,3 | 132,0 | 33,0 | | 6 | 98,8 | 124,0 | 31,0 |   **Ziegler-Nichols** | |  |  |  |  | | --- | --- | --- | --- | | Etapa | Kr | Ti | Td | | 1 | 106,5 | 145,6 | 23,2 | | 2 | 93,0 | 154,4 | 24,6 | | 3 | 85,5 | 167,8 | 26,7 | | 4 | 89,5 | 163,4 | 26,0 | | 5 | 95,8 | 149,9 | 23,9 | | 6 | 109,7 | 141,1 | 22,4 |   **Coen-Coon** |

Tabla : Parámetros controladores PID externos del horno.

Luego de ensayado el sistema con ambos juegos de coeficientes se obtuvo una respuesta idéntica, se optó por utilizar el método de Ziegler-Nichols por la robustez que ofrece ante los cambios de consigna (Ogata 2011)

La figura 13 muestra la respuesta en el tiempo de las etapas 1 y 3 del horno para cada juego de coeficientes. Se aprecia en la misma que no hay diferencia notable entre la respuesta con uno y otro juego de coeficientes.

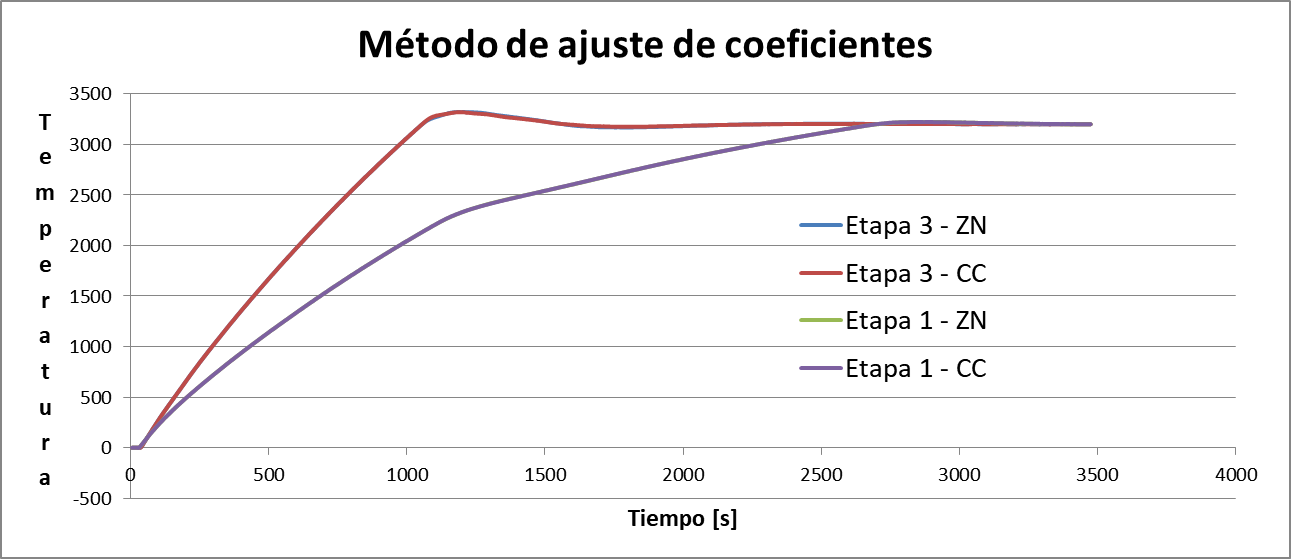


Figura : Ajuste de controladores PID

Con los controladores ajustados se procede a la segunda parte de la calibración, nuevamente se hace uso del método de la curva de reacción de proceso, sin embargo luego de la aplicación de los coeficientes, la prueba sobre la planta resulto en extremo subamortiguada. Posteriormente la verificación del modelo obtenido indicó que la identificación era deficiente. Por esto se decide utilizar en cambio el método de identificaron basado en cuadrado mínimos propuesto por Ogata [4], mediante el cual se obtuvieron los siguientes parámetros.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Etapa | Delay[s] | Kk[°C] | T[s] | S |
| 1 | 594,5688 | 2507 | 1614,231 | 0,000621 |
| 2 | 564,4886 | 2662 | 1065,485 | 0,000999 |
| 3 | 552,5199 | 2758 | 952,6078 | 0,001158 |
| 4 | 552,7684 | 2734 | 958,589 | 0,001141 |
| 5 | 530,7862 | 2603 | 1015,672 | 0,001025 |
| 6 | 593,2552 | 2471 | 1420,333 | 0,000696 |

Tabla : Parámetros sistema interno del horno.

A partir de los anteriores se obtuvieron los siguientes juegos de coeficientes, la prueba ratica revelo que el método de Coen y Coon provee el menor tiempo de establecimiento y un sobreimpulso menor al 10%

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Etapa | Kr | Ti | Td | | 1 | 3,3 | 1189,1 | 297,3 | | 2 | 2,1 | 1129,0 | 282,2 | | 3 | 1,9 | 1105,0 | 276,3 | | 4 | 1,9 | 1105,5 | 276,4 | | 5 | 2,2 | 1061,6 | 265,4 | | 6 | 2,9 | 1186,5 | 296,6 |   **Ziegler-Nichols** | |  |  |  |  | | --- | --- | --- | --- | | Etapa | Kr | Ti | Td | | 1 | 3,6 | 1159,7 | 202,6 | | 2 | 2,4 | 1034,4 | 187,2 | | 3 | 2,1 | 994,2 | 181,7 | | 4 | 2,1 | 995,8 | 181,9 | | 5 | 2,5 | 975,3 | 176,3 | | 6 | 3,2 | 1134,6 | 200,5 |   **Coen-Coon** |

Tabla : Parámetros controladores PID internos del horno.

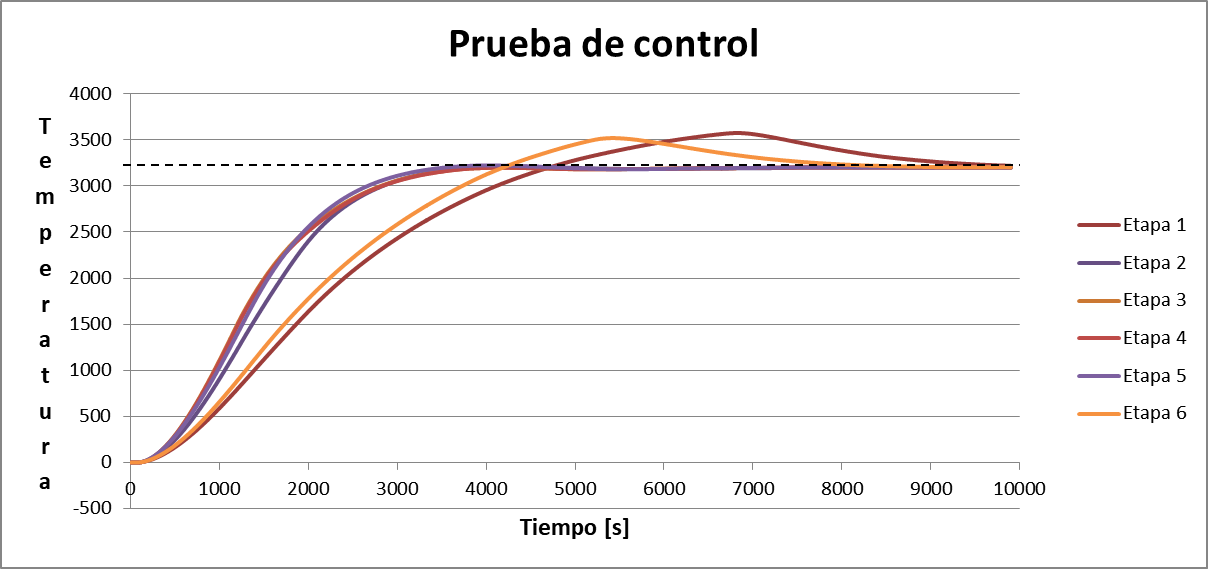


Figura : Prueba de control en escalera.

Puede verse en la gráfica que el tiempo de establecimiento es de alrededor de 3hs, lo que concuerda con as experiencias previas de calentamiento, también resulta razonable la evidente dificultad del sistema en calentar los extremos, estos disipan considerablemente más energía que el resto de los elementos internos.

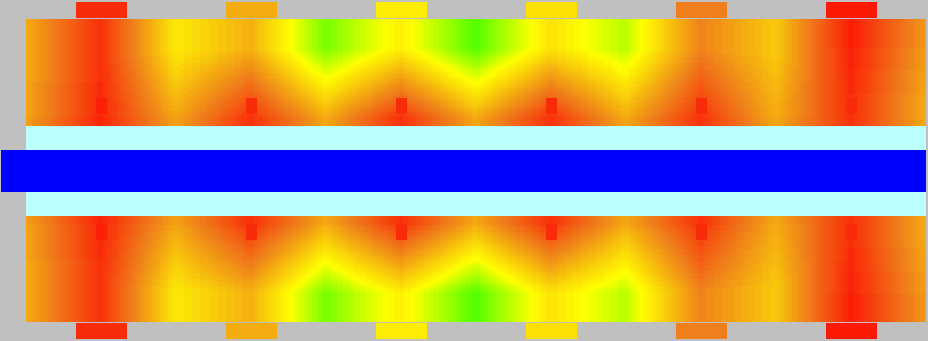


Figura : Prueba de control en escalera.

En la gráfica de temperatura se puede apreciar una distribución uniforme en el centro acorde a lo que muestra la figura anterior, resulta interesante evaluar la temperatura en los elementos calefactores, que será superior en los extremos para compensar la pérdida de los laterales e inferior en el centro por la acumulación de calor que provocan las unidades tan cercanas.

### Desarrollo del firmware.

El firmware del controlador de temperatura, de forma similar al del controlador principal, se divide en dos grandes secciones que funcionan de forma independiente, la primera es la comunicación con el horno junto con el control en cascada del mismo, mientras que la segunda comprende el control de la simulación, y la comunicación con el control principal.

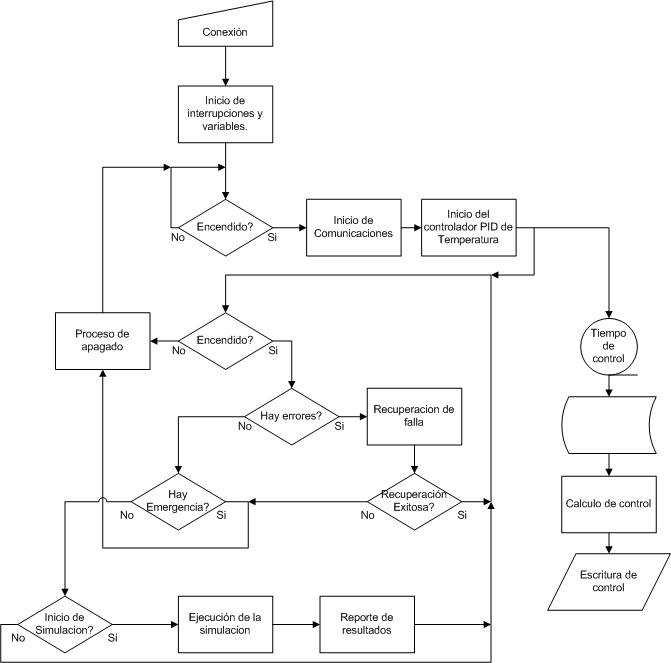


Diagrama : Firmware controlador de temperatura.

Puede observarse en el diagrama que luego de ser conectado el controlador inicializa las interrupciones y las variables, luego queda a la espera del encendido, una vez recibida la petición de encendido, se inicializa la comunicación con el horno, en este momento se lee el estado de los PID’s y los sensores, se inicializan las variables del control en escalera y el reloj de control, desde este punto en adelante, se realizará un ciclo de control cada un tiempo establecido por los parámetros de configuración, independientemente del estado del resto del sistema.

Antes de cada acción de control se leen los registros de PV de cada PID y los registros de la interfaz ADAM 4018+, en base a estos valores y los coeficientes de cada PID establecidos por los parámetros de configuración, se calcula un valor de SP que se escribe sobre cada PID. La comunicación se realiza mediante el protocolo MODBUS sobre una capa de hardware RS485.

Ante una petición de inicio de simulación, el sistema comienza a recorrer la tabla de comandos a la espera de encontrar un comando de temperatura, si no es así, verifica que el registro en ejecución permita la aplicación simultanea de una orden de temperatura, en este caso avanza al siguiente registro o en caso contrario espera a su finalización de esta forma cada procesador regula la ejecución de su incumbencia sin la necesidad de contar con un árbitro externo.

Los comandos de simulación de temperatura se describen brevemente en la siguiente tabla, fueron diseñados de forma de permitir un uso sencillo pero efectivo de la potencia del controlador en escalera aplicado.

|  |  |  |
| --- | --- | --- |
| Comando | Parámetros | Descripción |
| TEMP\_SET\_1 | Temperatura[°C] | Fija el set point de temperatura de la etapa 1. |
| TEMP\_SET\_2 | Temperatura[°C] | Fija el set point de temperatura de la etapa 2. |
| TEMP\_SET\_3 | Temperatura[°C] | Fija el set point de temperatura de la etapa 3. |
| TEMP\_SET\_4 | Temperatura[°C] | Fija el set point de temperatura de la etapa 4. |
| TEMP\_SET\_5 | Temperatura[°C] | Fija el set point de temperatura de la etapa 5. |
| TEMP\_SET\_6 | Temperatura[°C] | Fija el set point de temperatura de la etapa 6. |
| TEMP\_SET\_EQUAL | Temperatura[°C] | Fija el set point de temperatura de todas las etapas al mismo valor. |
| TEMP\_SET\_RAMP | Temperatura[°C] Izquierda y Derecha | Fija el set point de temperatura de las etapas 1 a la 6 según una rampa delimitada por los valores de temperatura izquierda y derecha. |
| TEMP\_WAIT\_STABLE | Porcentaje [%] | Pausa la simulación hasta alcanzar en todas las etapas un error porcentual menor al indicado. |

Tabla : Comandos de simulación del controlador de temperatura

### Resultados parciales.

Los resultados obtenidos luego de la correcta calibración del control de temperatura, fueron sumamente satisfactorios, se mejoraron los tiempos de establecimiento respecto del control simple y se pudo controlar con la precisión requerida la temperatura sobre todos los puntos de control. Es importante destacar que el tiempo requerido para el diseño del control auxiliar y la calibración no estaban considerados y fueron mayores incluso al tiempo estimado para toda la etapa, esto redunda en un costo superior al propuesto inicialmente.

## Controlador de Movimiento.

### Diseño propuesto.

### Hardware Electrónico.

De forma similar al diseño realizado para el controlador de temperatura, se utilizó como procesador el mencionado MicroBlaze y la estructura de conexión de la memoria también expuesta anteriormente.

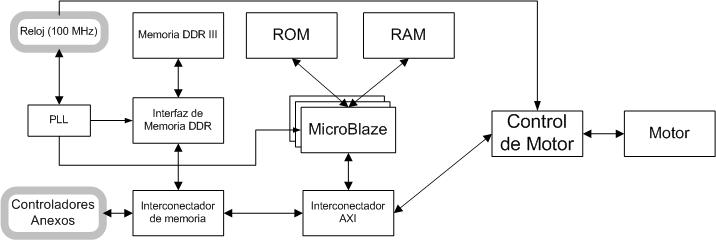


Diagrama : Controlador de movimiento.

Sobresale dentro del diagrama, la inclusión del módulo de control de motor, el diseño del mismo se realizó en código VHDL y represento un desafío su correcto funcionamiento. Genéricamente incluye el control de posición del motor y el decodificador del encoder de realimentación, su diseño se discutirá en detalle en la sección “Problemas y soluciones implementadas” de este mismo capítulo.

La elección del motor se rige por criterios de hardware que serán expuestos en la sección de hardware mecánico, sin embargo el motor es el elemento que determina el consumo de energía y junto a este han de seleccionarse el driver de potencia y la fuente de alimentación de forma de cumplir con los requerimientos de potencia.

Desde el punto de vista del control, la opción inmediata ante un movimiento de precisión, de velocidad controlada y de alto torque es un motor paso a paso. La capacidad de control que brinda de este tipo de motores es ideal para movimientos de precisión, además proveen de un torque muy alto en el arranque, aunque en general su comportamiento en altas velocidades deja que desear.

La disponibilidad en el mercado nacional, el precio junto a las características mecánicas deseadas para el diseño, llevan a la elección del motor Nema 24H2A9830 de la marca Motionking.



Figura : Motor Nema 24H2A9830

Dentro de sus características electro mecánicas más sobresalientes se encuentran:

* Corriente nominal de trabajo 3.0[A]
* Resistencia de cada fase 1.4[Ohms]
* Angulo de rotación de paso 1.8[Deg]
* Precisión de paso ±5%
* Precisión de la resistencia ±10%
* Temperatura máxima de trabajo 80[°C]
* Temperatura normal de trabajo -20[°C] ~ +50[°C]
* Juego Radial 0.02[mm]
* Juego Lineal 0.08[mm]
* Fuerza axial 75[N]
* Fuerza radial 15[N]

Con las características eléctricas del motor, se analizaron las distintas alternativas del control de potencia necesario para el manejo del motor, rápidamente se llegó a la conclusión de que no resulta conveniente desde el punto de vista económico enfrentar el diseño de un driver de potencia, con todo lo que esto requiere, para producir una sola unidad, se buscó entonces en el mercado un producto adecuado para el manejo del motor. El elemento también disponible en el mercado nacional es el driver M524 de la empresa Leadshine Technology Co. Ltd. Este provee además de una corriente de manejo de hasta 4[A], la opción de generar micro pasos, para esto el driver provee a cada fase la corriente necesaria para posicionar al motor en uno o varios puntos intermedios entre un paso y el siguiente. Esta capacidad adicional brinda la posibilidad de una mayor precisión respecto de la original del motor, además de un funcionamiento más suave y con menor cantidad de ruido.



Figura : Driver Leadshine M542

Las entradas de control de paso, dirección y habilitación del motor son foto acopladas, esto brinda una aislación eléctrica de la lógica de control, sin embargo vienen de fábrica adaptadas para un nivel de excitación de 5[V], a partir de una consulta al fabricante se pudo adaptar la entrada para 3,3[V] sin la necesidad de colocar elementos de conversión de nivel adicionales. Dentro de sus características más sobresalientes se encuentran:

* Bajo costo y alto torque
* Alimentación hasta +50[V]
* Corriente de salida hasta 4.2[A ]
* Señales de entrada opto acopladas
* Frecuencia de paso hasta 300KHz
* Reducción automática de la corriente
* Tecnología de control de corriente de tres estados
* 15 resoluciones seleccionables
* Apto para motores de 2 y 4 fases
* Selección de corriente de 8 valores
* Protección contra sobre voltaje y sobre corriente.
* Tamaño pequeño 118x75.5x33[mm]

Luego del driver, se selecciona la fuente de alimentación, por su tamaño reducido bajo coste, menor peso y fiabilidad, se utilizan dos fuentes tipo switching colocadas en serie, la primera de 12[V] 20[A] para alimentar la lógica discreta, y la segunda de 24[V] 10[A], juntas suman 36[V], necesarios para alimentar el driver de potencia visto anteriormente.

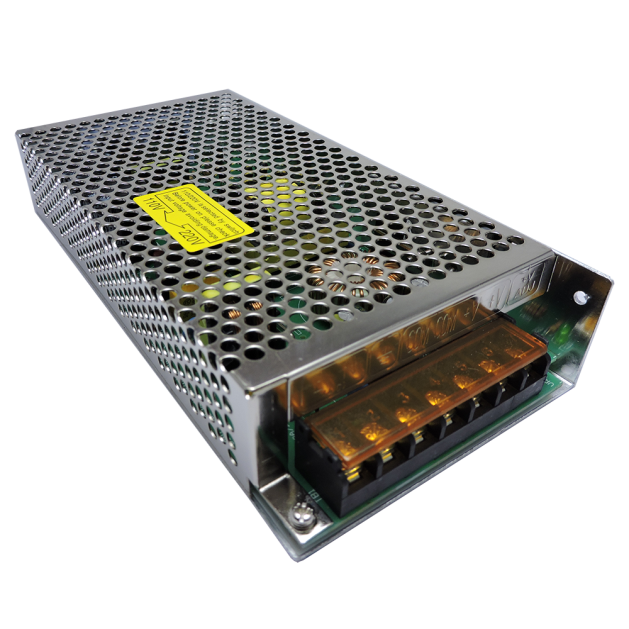


Figura : Fuente switching

Por último se decidió colocar un encoder rotativo acoplado al eje del motor de forma de por contar con un retorno de posición del mismo, el elemento seleccionado es el G40B-6-400-2-24, este encoder se alimenta con una tensión de 12[V] y posee dos salidas desplazas en fase 90°, que conmutan 400 veces por vuelta, esta configuración permite la detección del sentido de giro.



Figura : Encoder G40B-6-400-2-240

### Hardware Mecánico.

Teniendo en cuenta la amplitud del movimiento, la sensibilidad requerida y la importancia del correcto funcionamiento de esta etapa, resulta indispensable un cuidadoso diseño de la estructura mecánica, acorde a las características y dimensiones del elemento en desarrollo.

Para el diseño del sistema de posicionamiento de la barra, se tuvieron en cuenta los parámetros de funcionamiento, las dimensiones y las características del entorno.

Dimensiones del elemento a mover:

* Longitud 2400[mm]
* Diámetro máximo 36[mm]
* Peso 60[Kg]

Parámetros de funcionamiento:

* Máximo desplazamiento lineal 1500[mm]
* Máxima velocidad de desplazamiento lineal 20[mm/s]
* Máxima aceleración lineal 40 [mm/s2]
* Máxima variación de la aceleración o Jerk 120[mm/s3]
* Temperatura máxima de trabajo 460[°C]
* Temperatura máxima de ensayo 520[°C]

Características del entorno:

* Distancia al elemento lateral más próximo [40]mm
* Predominancia de elementos no magnéticos.

Se diseñó teniendo en cuenta los requerimientos anteriores un sistema de actuación lineal, de 1600mm de longitud. Se excede el requerimiento en 100mm para dejar lugar a espacios de calibración y posicionamiento de emergencia.

Para guiar el desplazamiento se utilizaron dos guías cilíndricas de Acero inoxidable montadas sobre perfiles te de aluminio, que permiten el montaje de rodamientos lineales. Dada la longitud del recorrido se seleccionaron diámetros de guía de 16mm, esto garantiza la suficiente estabilidad y robustez para el desplazamiento de la barra.

Ambas guías se montaron con tornillos de acero inoxidable no magnético a una carcasa plegada a medida también de material no magnético.

La trasmisión del movimiento del motor al cabezal de sujeción y desplazamiento se realizó mediante un tornillo calibrado con un paso de rosca de 5mm y una tuerca de bolillas re circulantes para evitar el juego longitudinal del mecanismo que podría generar desperfectos en el posicionamiento. Se acoplo el motor al tornillo mediante un acople diseñado para absorber la vibración de los elementos mecánicos, de forma de proteger al motor como al mecanismo de eventuales golpes.

El encoder G40B-6-400-2-24 se montó mediante el uso de un acople anti shock en el extremo contrario del tornillo de rosca calibrada.

Todos los materiales de montaje como tornillos, arandelas, elementos de sujeción y otros varios fueron construidos de material no magnético, en su mayoría, acero inoxidable de tipo quirúrgico proveniente de Bélgica.

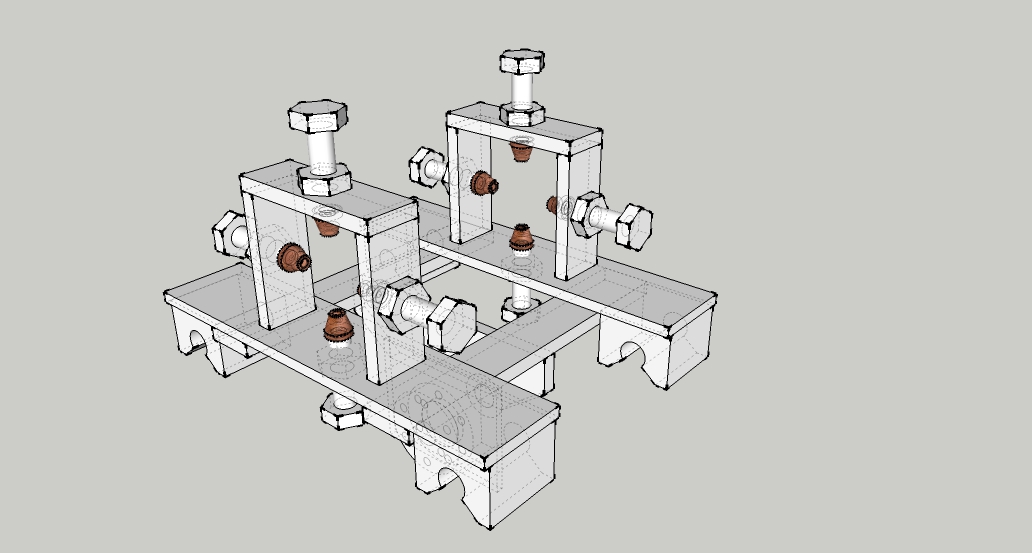


Figura : Cabezal de sujeción y desplazamiento

El cabezal de sujeción y desplazamiento es el encargado de sujetar la barra y transmitirle el movimiento, su construcción integra de acero inoxidable garantiza su rigidez estructural, mientras que los 8 tornillos de ajuste de la barra terminados en punteras de cobre aseguran una sujeción estable y sin ocasionar daños sobre el elemento a desplazar.

Los 8 tornillos de sujeción se encuentran ubicados de forma de permitir mediante su ajuste el correcto posicionamiento de la barra dentro de la máquina y asegurar un ingreso adecuado al sensor bajo ensayo. En este elemento en particular se consideró adicionalmente la conductividad eléctrica, para aislar eléctricamente la barra del resto del sistema se colocaron delgadas planchas de teflón entre el cabezal y los rodamientos lineales logrando de esta forma una aislación eléctrica eficiente.

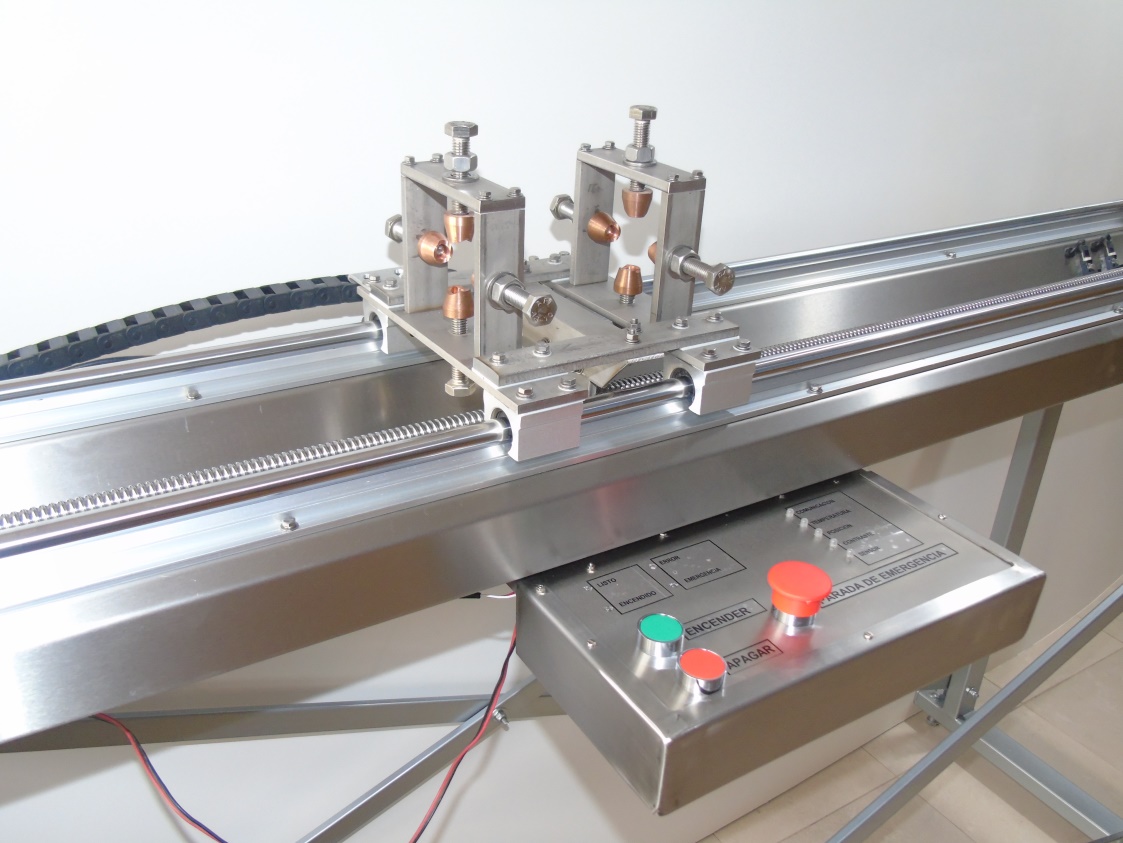


Figura : Actuador lineal

### Problemas y soluciones implementadas.

Al momento de seleccionar el tipo de motor, se consideró la posibilidad de utilizar un motor paso a paso de lazo cerrado, este tipo de motor lleva incluido un encoder que le indica al control la posición real de forma de poder corregir la perdida de pasos, sin embargo su costo es de tres veces el de un motor sin este sistema, y requiere de un driver especial que tiene un costo cinco veces mayor al utilizado, por esto se decidió adquirir el motor mencionado anteriormente y verificar si es necesaria la corrección de la posición, el sistema prevé la colocación de un encoder de forma que resulta fácil medir el error real de desplazamiento del motor.

Luego de las primeras pruebas se probó que existía un error de hasta 3 pasos cada 1000, este se encuentra dentro de los rangos esperados, sin embargo se pudo comprobar que ante un movimiento de una hora sin calibración por puesta a cero, el error absoluto asciende hasta los 12[mm], esto es inaceptable para los requisitos del sistema, como también es inadmisible la necesidad de una calibración a cero luego de comenzada la experiencia.

Para subsanar este error se evaluó la implementación de un control de lazo cerrado capaz de realizar los pasos perdidos al finalizar el movimiento en ejecución, sin embargo esto es poco eficiente en movimientos muy largos o muy cortos, en ambos casos se modifica el tiempo de ejecución del movimiento adicionando un tiempo extra aleatorio que impacta sobre la capacidad del mecanismo de hacer simulaciones de dinámica de movimiento.

Una alternativa conveniente es la aplicación del control del diagrama 12, su implementación se adapta a las tres señales de control provistas por el módulo de generación de trayectoria que se verá más adelante. Estas señales son; la velocidad, la cantidad de pasos de movimiento y la dirección.

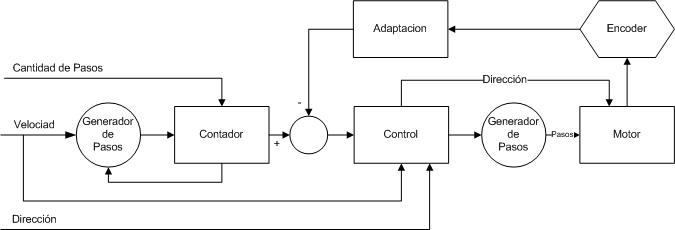


Diagrama : Sistema de control de lazo cerrado del motor

La velocidad alimenta un generador de pasos que internamente consta de un acumulador cuya frecuencia de conteo es igual a la del sistema 83,333 MHZ y que producirá a la salida una señal cuadrada de frecuencia igual a la velocidad requerida, ha de tenerse en cuenta que la velocidad se encuentra expresada en pasos por segundo, que en este caso equivale a pulsos por segundo o frecuencia.

A la salida del generador se coloca un contador que se detendrá al llegar a la cantidad de pasos requeridos, este contador informa al control la cantidad efectiva de pasos a realizar.

El sumador es el primer bloque del control propiamente dicho, calcula el error entre la cantidad teórica de pasos y la cantidad de pasos medidos por el encoder. El bloque control es el encargado de insertar los pasos faltantes o sustraer los sobrantes a lo largo de la trayectoria, para esto toma el mismo parámetro velocidad mencionado anteriormente y le adiciona un valor proporcional al error en el caso de haber una deficiencia de pasos del motor en caso contrario resta un valor proporcional. El segundo generador de pasos funciona de forma idéntica al primero y su salida es la señal de control final del driver del motor.

La dirección del movimiento determina los signos de los errores, acumuladores y es adoptada como referencia por el control para generar la señal de dirección del motor. El algoritmo tiene en cuenta la imposibilidad de generar una inversión brusca de giro, sin embargo es capaz de influir sobre esta señal al final del movimiento para realizar las correcciones necesarias, ya fuera por un error en la cantidad de pasos o por una intervención externa.

El módulo de adaptación, es necesario para convertir la cantidad de pulsos del encoder en un valor de pasos equivalentes a los del motor, en este sistema la sensibilidad del motor es mayor a la del encoder, esto es tenido en cuenta en la tolerancia de error del control.

Luego de resuelto exitosamente el problema de control a lazo cerrado del motor, se determinó que el controlador no podría ser lo suficientemente rápido para generar las rampas de aceleración y velocidad requeridas para la simulación de movimiento, por lo que se decide construir en hardware un módulo capaz de coordinar los distintos tipos de movimientos a partir de los parámetros de entrada:

* Cantidad de pasos.
* Velocidad.
* Aceleración.
* Jerk.

El último de los parámetros mencionados anteriormente no es de uso frecuente, por lo que fue necesario investigar su significado, Chen, Wang, y Wei (2006) lo definen como la relación de cambio de la aceleración en el tiempo, lo que constituye matemáticamente la primer derivada de la aceleración, mecánicamente se lo asocia a la brusquedad con la que se inicia un movimiento o se cambia de velocidad a lo largo de este.

Es usual en los controles de movimiento una curva de velocidad del tipo trapezoidal, que comienza con una rampa de aceleración constante hasta alcanzar la velocidad máxima, este tipo de movimiento tiene un valor de jerk infinito, ya que la aceleración es un escalón y por ende su primera derivada será un impulso infinito.

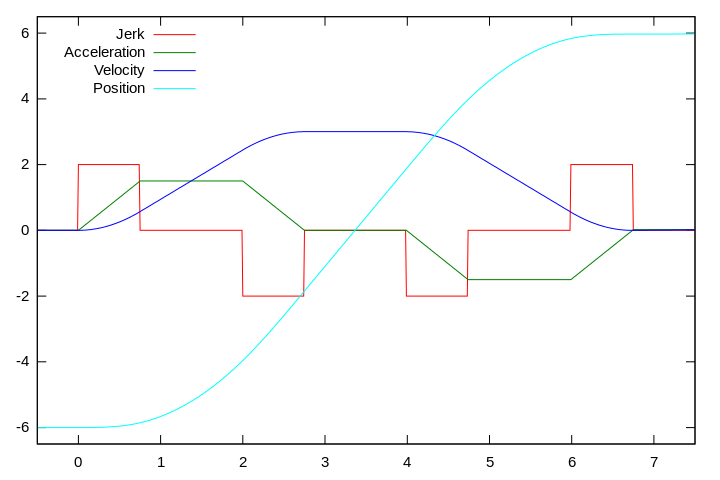


Figura : Curva de trayectoria S

La figura 22, muestra un movimiento con un valor de jerk controlado, pueden observarse siete tramos perfectamente distinguidos en la trayectoria:

1. La primera etapa de la curva presenta un valor de jerk constante, una aceleración que aumenta linealmente mientras que la velocidad lo hace de forma cuadrática.
2. La segunda etapa comienza al alcanzarse el valor de aceleración máxima y en este intervalo se mantiene constante mientras que la velocidad se incrementa linealmente.
3. La tercer etapa comienza en el momento adecuado de forma de al alcanzar la duración del primer tramo haber llegado a la velocidad objetivo.
4. La cuarta etapa es la más sencilla y corresponde a un movimiento de velocidad constante.
5. Las etapas 5 a la 7 presentan características similares a las tres primeras con signo invertido.

Con lo visto anteriormente como guía, se diseñó un módulo a base de contadores anidados capaz de generar cada etapa de la trayectoria aceptando como parámetro de entrada cada uno de los parámetros detallados al comienzo. El tiempo que transcurre entre el final de una etapa y el comienzo de la siguiente es mínimo, se ideó también una metodología de carga de datos oportuna para evitar demoras entre la carga del tramo y su ejecución.

### Desarrollo del firmware.

El firmware del controlador de movimiento, es comparativamente muy sencillo respecto de los anteriores, su funcionamiento es similar, ante la petición de arranque lleva la posición a cero e inicializa el módulo de control del motor, luego queda a la espera de una petición de simulación, en cuyo caso comienza a recorrer la tabla de comandos, ante un comando de movimiento no realiza una interpretación sino que directamente lo carga al controlador de movimiento, esto permite un ahorro de espacio y de capacidad de procesamiento, además garantiza la velocidad de carga necesaria para evitar cortes en el movimiento debido a la ausencia de comandos en el controlador.

La mayor parte de la carga de procesamiento, se desliga en el software, por continuidad y para evitar redundancias, se desarrollara brevemente el funcionamiento del módulo de software encargado de la proyección de la trayectoria.

Los comandos de movimiento a ejecutarse son:

* MOV\_SET\_JERK: Permite cargar un valor de jerk distinto al valor por defecto para la realización del próximo movimiento. Luego de ser utilizado el valor no regresará al valor por defecto, generará una entrada en la tabla..
* MOV\_SET\_ACEL: Permite cargar un valor de aceleración distinto al valor por defecto para la realización del próximo movimiento. Luego de ser utilizado el valor no regresará al valor por defecto, generará una entrada en la tabla.
* MOV\_SET\_AUTOMATIC: Permite al software utilizar los valores de jerk y aceleración necesarios para que los primeros y últimos tres tramos del movimiento duren un porcentaje a elección del tiempo total de desarrollo de los mismos, ha de tenerse en cuenta que el cambio de los parámetros jerk y aceleración previo a cada movimiento generara dos entradas adicionales en la tabla de comandos.
* MOV\_DIST\_VEL y MOV\_TOPOINT\_VEL: Generan la secuencia de comandos necesaria para recorrer una distancia determinada el primero o la distancia necesaria para llegar a un punto en el segundo caso.
* MOV\_DIST\_TIME: Realiza un movimiento de una distancia establecida por comando en un tiempo dado. El análisis de este comando resulta de difícil implementación, se realiza un algoritmo similar al de búsqueda por burbuja que prueba dentro del rango de velocidad la aplicación del comando MOV\_DIST\_VEL hasta obtener el parámetro de velocidad que brinde el tiempo requerido.
* MOV\_TOPOINT\_TIME: Similar al anterior pero en vez de una distancia se mueve hasta un punto.
* MOV\_STOP: Detiene el movimiento, le indica al próximo comando que la velocidad inicial es cero y al comando anterior que la velocidad final es 0.
* MOV\_STOP\_TIME: Detiene el movimiento de forma similar al anterior, pero además demora un tiempo fijado para comenzar uno nuevo.
* MOV\_SET\_SIN\_T: Define el periodo de oscilación para la utilización de los comandos MOV\_SIN\_ATT y MOV\_SIN, su uso es estrictamente necesario antes de cualquiera de los comandos anteriores, no generará una entrada en la tabla de comandos.
* MOV\_SIN\_ATT: Realiza un movimiento sinusoidal atenuado de una amplitud dada por el parámetro a partir del punto actual t con un periodo de oscilación dado por el parámetro T mencionado anteriormente.
* MOV\_SIN: Similar al anterior pero sin atenuación y con una amplitud A fijada por el parámetro del comando.

### Resultados parciales.

Los resultados de esta etapa fueron, luego de un arduo trabajo de diseño del hardware, superiores a los esperados, la interfaz de control del motor se comportó de acuerdo a lo deseado, todos los movimientos son controlables en la totalidad de los parámetros propuestos y de una forma relativamente sencilla.

La electrónica de potencia seleccionada brindo excelentes resultados, en cuanto a la temperatura de trabajo es intermedia, lo que indica una exigencia de acuerdo al diseño.

Por otro lado el hardware mecánico resulto funcional como se esperaba, no presento complicaciones en las pruebas de uso previo, además se destaca la facilidad de montaje de la barra, para la realización de la experiencia. Se prevé sí que la calibración previa a la puesta en marcha requiera de un tiempo considerable.

## Sistema de adquisición de datos.

### Introducción.

El sistema de adquisición de datos, es la unidad encargada de recoger los datos generados por el resto de los controladores y reportarlos al software de la PC para su posterior análisis. Tiene como función principal el registro y almacenamiento de las variables; tiempo, temperatura, posición y el valor del sensor bajo ensayo, su correcto desempeño determinará el funcionamiento de la máquina, y en gran medida su performance.

### Diseño propuesto.

De forma similar al diseño realizado para los controladores, se utilizó como procesador el mencionado MicroBlaze y la estructura de conexión de la memoria también expuesta anteriormente. Se agrega el driver necesario para la conexión mediante RS-485 del sensor bajo prueba, y el controlador para la lectura del sensor de contraste.

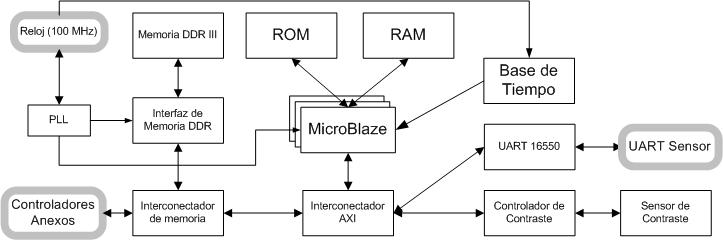


Diagrama : Sistema de adquisición de datos.

La característica distintiva del hardware se centra en la base de tiempo, esta es fundamental para la correcta adquisición de las muestras y más aún para su posterior procesamiento.

Para poder realizar la conexión de la UART del sensor, es necesario colocar una adaptador de niveles que cumpla con los requisitos del estándar EIA-485 para la capa física del modelo OSI, para esto se emplea el integrado MAX487, este permite un nivel de alimentación de 3,3V, adecuado para la tensión de salida de la lógica digital, además permite la implementación de una comunicación half dúplex, mediante dos señales de activación, la primera habilita el modulo receptor y la segunda el modulo emisor, conectándolas e paralelo se logra una comunicación sin eco que se verá habilitada a transmitir colocando un 1 lógico en los pines de habilitación y a recibir colocando un 0 lógico.

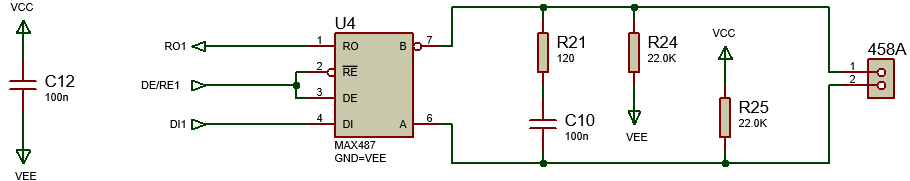


Figura : Adaptador RS485

Puede observarse a la salida una red de adaptación de impedancias, esta fue colocada según lo indica el fabricante para evitar sobre impulsos propios de una línea con impedancia infinita y para filtrar las componentes de muy alta frecuencia.

Dentro de las principales ventajas que brinda este modelo físico se encuentra la capacidad de transmitir en altas velocidades sobre largas distancias (10 Mbit/s hasta 12 metros y 100 kbit/s en 1200 metros) y a través de canales ruidosos, admite 32, 128 o 256 estaciones en 1 solo par, con una longitud máxima de 1200 metros operando entre 300 y 19 200 bit/s. La transmisión diferencial permite alcanzar mayor distancia con una notable inmunidad al ruido, siempre que el bus de comunicación conserve las características de bus balanceado.

Un punto fundamental en la utilidad de la maquina es la capacidad de calibrar el sensor, esta utilidad se plasma en el sistema de adquisición de datos, mediante el sensor de contraste. La calibración es el proceso de comparar los valores obtenidos por un [instrumento de medición](https://es.wikipedia.org/wiki/Instrumento_de_medici%C3%B3n) con la medida correspondiente de un patrón de referencia (o estándar) bajo condiciones específicas, para calibrar un instrumento o un estándar se necesita disponer de uno de mayor [precisión](https://es.wikipedia.org/wiki/Precisi%C3%B3n) (patrón) que proporcione el valor convencionalmente verificable, el cual se utilizará para compararlo con la indicación del instrumento que está siendo sometido a la calibración (Moro Piñeiro 2000) . El objetivo es mantener y verificar el buen funcionamiento de los equipos, responder a los requisitos establecidos en las normas de calidad y garantizar la fiabilidad y la trazabilidad de las medidas.

El primer paso para el diseño de este elemento es la elección del sensor que se encargue del contraste de las medidas, existen diversas opciones y métodos de medición en el mercado, dentro de las opciones evaluadas, se encuentran:

* Los sensores de desplazamiento lineal, cuyo costo y grandes dimensiones los hacen difícil de aplicar en este caso.
* Los sensores de medición laser, que presentan un desvío importante de la medición con la temperatura y son de difícil calibración para aceptarlos como patrón.
* Los sensores ultrasónicos, cuya precisión es insuficiente.
* Los encoders de posición, que tienen un tamaño aceptable y su precisión depende de características propias del hardware.
* Sensores inductivos, que son poco adecuados para el rango de desplazamiento.

Luego de analizar las opciones, se decidió la utilización de un encoder rotativo de posición, su facilidad de montaje, fiabilidad, resistencia a la temperatura, facilidad de calibración, disponibilidad y certificación de calibración de fábrica, motivaron su uso.

El sensor a utilizar es Accu-Coder Tru-Trac de la empresa Encoder Products Company, su fabricante lo describe como una solución versátil para rastrear la velocidad, posición o distancia sobre una amplia variedad de superficies en casi cualquier aplicación. Su brazo de torsión con resorte proporciona una carga de torsión fácil de ajustar, lo que permite que se monte en casi cualquier orientación, incluso al revés. El eje roscado en el eje de pivote es reversible en campo, lo que proporciona acceso de montaje desde cualquier lado. La carcasa del codificador es de un material compuesto duradero y conductivo que eliminará la acumulación de estática. Permite velocidades de operación de hasta 15[m/s].



Figura : Sensor Accu-Coder Tru-Trac.

El modelo adquirido incluye una rueda con borde de goma como el de la figura, con un diámetro calibrado de 150mm, el encoder tiene una resolución de 2500 puntos por vuelta, que se puede duplicar si se miden los flancos en vez de los niveles de conmutación. De esta forma se obtiene una sensibilidad lineal de contraste de 0,03[mm], un valor muy superior a los 0,2[mm] requeridos.

Al igual que la mayoría de los encoders, este provee una salida A y otra B, que corresponden a dos señales en contrafase, que determinarán además de la posición, la dirección de la rotación. La alimentación, y la adaptación de las señales se realiza mediante el circuito la siguiente imagen.

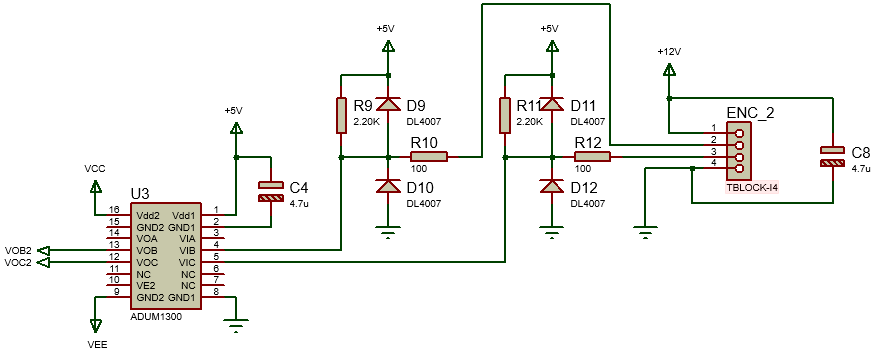


Figura : Esquema adaptador del sensor de contraste

Puede verse en el circuito, la entrada de las señales A y B, luego los diodos D9 a D12, componen una protección contra sobretensiones que pudieran provenir del exterior para proteger la electrónica, se colocan R9 y R11 para evitar estados indeterminados además de brindar compatibilidad con sensores de colector abierto. Por último el integrado ADUM1300 es un aislador electromagnético a la vez que conversor de nivel, este se coloca para aislar eléctricamente la lógica digital de los circuitos que tienen contacto con la máquina y que pudieran recibir descargas. Este integrado funciona de forma similar a un opto acoplador pero permite velocidades de hasta 20MHz, la velocidad de conmutación de las señales representa un impedimento para el uso de opto acopladores que serían más económicos y sencillos de utilizar en este caso.

### Problemas y soluciones implementadas.

La comunicación con el sensor, según se describe anteriormente en este mismo capítulo, se realiza mediante el estándar de comunicaciones EIA-485, más conocido como RS485, su conexión en principio resulta sencilla, se utiliza la misma UART16550 que fue empleada en las comunicaciones RS232, sin embargo se verifico inmediatamente que la UART no dispone de un pin de habilitación para la oportuna activación y desactivación de los drivers de emisión y recepción de datos. Se colocó entonces la habilitación en el pin RTS de salida de la UART, sin embargo el manejo de este pin se hace mediante software, no es automático como se describe en al protocolo, tampoco resulta ser automático para el caso de la UART de la PC, este pequeño detalle supone una carga de software importante para intervenir en la habilitación de los drivers ante cada envío y recepción de datos.

Se implementó entonces una función capaz de generar una señal de habilitación del driver emisor ante cada envío y el resto del tiempo se mantuvo encendido el driver receptor. Aun así al funcionar de forma independiente la FIFO, del sistema de envío de la UART, no se podía determinar correctamente el momento de finalización del último carácter enviado, se aplicó entonces una solución de compromiso, cuya validez se encuentra sujeta a la aplicación de la comunicación en el protocolo MODBUS, se agregó un tiempo de espera reglamentario de 1,5 caracteres al finalizar cada transmisión, esto es válido si tenemos en cuenta que el mínimo tiempo de respuesta desde el esclavo es de 3,5 caracteres.

### Desarrollo del firmware.

El desarrollo del firmware de la unidad de adquisición de datos es similar en su estructura al del resto de los controladores, prevé un inicio o conexión del procesador en el cual se configuran las interrupciones de la base de tiempo y de la UART, luego el sistema queda a la espera del encendido, una vez en este estado, realiza la secuencia de inicio de las comunicaciones con el sensor y el diagnóstico del mismo, de no ser positivo el diagnostico, se coloca en estado de error a la espera de la solución de la falla.

En el estado de encendido el sistema lee periódicamente el estado del sensor, del contraste de posición y las temperaturas del horno. Al entrar en el estado de simulación, se comienza con un muestreo seguido del almacenamiento de los datos en la memoria DDR. En el siguiente diagrama puede observarse este proceso.

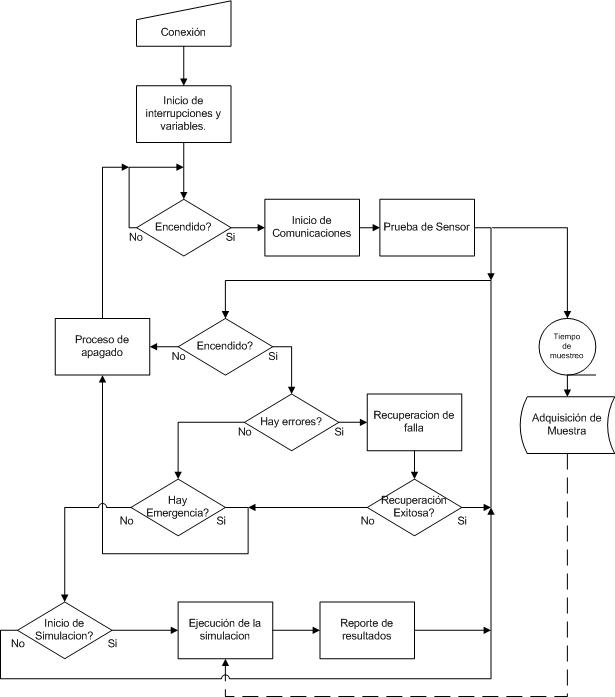


Figura : Firmware sistema de adquisición de datos

Escapan al alcance del diagrama, las características de la las muestras, y los parámetros que definen su obtención. Es fundamental entender que para la correcta adquisición de los datos han de determinarse 3 características fundamentales del sistema:

* La velocidad de muestreo: determinada por la variación más rápida del sistema.
* El rango de muestreo: está determinado por el rango de la variable a medir.
* La sensibilidad: es la mínima variación de la variable que se puede medir.

La siguiente tabla detalla cada variable a medir y sus características

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable | Sensibilidad | Rango | Velocidad | Frecuencia |
| Tiempo | 0,001[s] | 0-10[días] |  | 1[KHz] |
| Posición | 0,001[mm] | 0-1500[mm] | 40[mm/s] | 40[KHz] |
| Temperatura | 0,1[°C] | 0-550[°C] | 1[°C/s] | 10[Hz] |
| Sensor | 1Bit | 4\*64Bits | 5[Muestras/s] | 5[Hz] |

Tabla : Variables a medir y sus características.

En función de lo anterior y teniendo en cuenta los recursos de procesamiento, se fija una frecuencia de muestreo máxima de 1KHz, esto cumple ampliamente con los objetivos planteados y cubre todos los casos posibles.

Luego de interiorizarnos en el trabajo de investigación y la adquisición de muestras, nos damos cuenta que un muestreo periódico gobernado por un tiempo como el que se solicitó, pude brindar una gran cantidad innecesaria de datos, que resultan difíciles de procesar y no aportan información de calidad a la investigación, se propone entonces un método alternativo de captura de datos, basado en distancias periódicas o absolutas. Considerando lo solicitado y el nuevo método propuesto se implementan para el control del muestreo siete comandos que incluyen todos los casos, a continuación desarrollaremos la funcionalidad y el procedimiento de toma de muestras que implica cada uno de ellos.

* MUEST\_PERIODIC\_T: Es el comando más básico, tiene un solo parámetro de entrada que es el tiempo entre muestras, el sistema toma una muestra al recibir el comando y mide continuamente el transcurso del tiempo hasta alcanzar el valor del parámetro, en este momento tomará la siguiente muestra y así repetitivamente.

Su principal limitación es que solo permite un tiempo entre muestras.

* MUEST\_PERIODIC\_T\_STOP: Detiene el muestreo por tiempo, no necesita de ningún parámetro.
* MUEST\_PERIODIC\_X: Este comando se añade como una característica adicional, su comportamiento es similar al muestreo por tiempo, también requiere de un solo parámetro que indica la distancia entre muestras. Se basa en una consulta cada 100[us] al sensor de contraste de forma de determinar si la posición actual es un múltiplo entero de la distancia entre muestras. Para este comando fue necesario adicionar un algoritmo que permitiera la adquisición de la muestra al alcanzar la posición y luego no repetirla hasta detectar un desplazamiento, ya que puede llenarse rápidamente la memoria si se tomaran muestras cada 100[us] hasta que cambie la posición.
* MUEST\_PERIODIC\_X\_STOP: Detiene el muestreo por posición, no necesita de ningún parámetro.
* MUEST\_OT\_ON\_X: Este comando requiere de un parámetro que indicará la posición en la cual se debe tomar la muestra, tomará una sola muestra y luego dejará de estar activo.
* MUEST\_ON\_X: Este comando permite tomar una muestra al pasar por una posición en particular de forma repetitiva, necesita de un parámetro que indicara la posición en la cual se debe tomar la muestra y a diferencia de los anteriores, pueden existir más de una instancia del comando en ejecución, de hecho hasta 256 posiciones distintas pueden ser muestreadas por este comando. Si se quisiera sobrepasar este límite durante la simulación, el controlador informaría un error y se detendría, es trabajo del diseñador de la experiencia no sobrepasarlo.
* MUEST\_ON\_X\_STOP: Este comando permite liberar un espacio en la tabla de muestreo del comando anterior, el parámetro de posición determinara cual liberar.

### Resultados parciales.

Los resultados de esta unidad de forma similar a lo sucedido en casos anteriores, dependen en gran medida del funcionamiento del conjunto, sin embargo se realizaron de forma exitosa muestreos de tiempo periódico, a partir de esta experiencia también se pudo concluir que ante una carga de muestreo excesiva, el tiempo máximo de muestreo que permite el almacenamiento de los datos se ve reducido drásticamente, queda entonces a cargo del usuario diseñar una experiencia con una tasa de muestreo acorde a las necesidades y a las capacidades de almacenamiento.

Adicionalmente se pudo comprobar que aun ante una carga de muestreo superior a la propuesta se logra la lectura del sensor en tiempo y forma, esto es crucial a garantizar para este sistema.

La precisión en el muestreo por posición es un punto muy positivo a destacar, a pesar de no estar entre los objetivos originales, consideramos que supone una utilidad adicional muy valiosa y además su implementación demando escasos recursos.

## Interfaz HMI.

La interfaz HMI (Human Machine Interface), es el medio por el que el usuario es capaz de comunicarse con la máquina, y comprende todos los puntos de contacto entre el usuario y el equipo. El objetivo de ésta interacción es permitir el funcionamiento y control más efectivo de la máquina, desde la interacción con el humano.

Las funciones principales de la interfaz son las siguientes:

* Puesta en marcha y apagado.
* Control de las funciones manipulables del equipo.
* Manipulación de archivos y directorios.
* Herramientas de desarrollo de aplicaciones.
* Comunicación con otros sistemas.
* Información de estado.
* Sistema de ayuda interactivo.

### La interfaz de hardware.

Según su construcción, la interfaz de usuario de hardware, es el conjunto de controles que permiten que el usuario intercambie datos con la máquina, ya sea introduciéndolos mediante pulsadores o leyéndolos mediante indicadores luminosos.

El sistema posee tres botones pulsadores, con las funciones de encender, apagar y por reglamentación, parada de emergencia.

Los testigos luminosos son un total de nueve, de los cuales:

* Dos son de color verde, indican que el sistema se encuentra encendido y conectado a la red eléctrica y otro que el sistema está listo para recibir órdenes.
* Dos son de color rojo para alarma de situaciones de error o falla, ya sea mecánica o lógica, y otro se activa al generarse la parada general del sistema por emergencia.
* Cinco son indicadores bicolores, verde y rojo, que informan acerca del estado y respuesta correcta de los subsistemas, la comunicación con la terminal de escritorio, el contraste en la medición, la respuesta del motor, la respuesta del horno, y la respuesta del sensor a calibrar. En verde se informa un correcto funcionamiento, y en rojo una anomalía o falla.

La disposición de los elementos que componen el panel se observan en la figura 27.

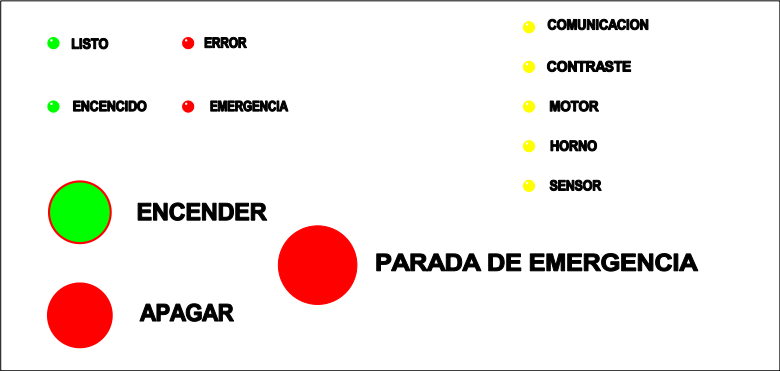


Figura : Panel de control

### Software de simulación de entorno controlado.

### La interfaz de software.

La interfaz de usuario de tipo software, es el programa que permite expresar las órdenes a la maquina o visualizar su respuesta. El programa está compuesto por dos secciones, en la primera sección se genera una trayectoria y en una segunda se lleva a cabo la simulación completa del experimento.

La generación de la trayectoria se realiza mediante una lista secuencial de comandos básicos, los cuales tienen una función determinada en la máquina y pueden requerir de parámetros establecidos por el usuario, que son pre evaluados y analizados en concepto de viabilidad. Se puede decir que es una interfaz de línea de comandos *(Command-Line Interface,CLI)*, alfanumérica que sólo presenta texto. Luego de una correcta evaluación, se generan dos gráficas que pre visualizan la posición en función del tiempo que tendrá la trayectoria programada y la distribución de la temperatura a lo largo del horno, ambas son interfaces gráficas. La trayectoria generada puede archivarse en un documento de extensión “xlsx”, para luego poder ser reutilizado o modificado y así poder iniciar la simulación la cantidad de veces requerida.

En la sección de simulación, una interfaz gráfica nos muestra el estado del horno con respecto a su temperatura. El usuario utiliza un archivo previamente generado en la sección anterior que contiene los comando que realizará la maquina en función del tiempo. Ésta sección posee la capacidad de poder establecer una comunicación continua con la máquina, mediante el protocolo Modbus/TCP, y a la vez de controlar el sistema de regulación de temperatura industrial de un horno eléctrico, mediante el protocolo Modbus RTU sobre una interfaz RS485 que comanda a los seis controladores electrónicos del tipo PID. El programa recibe continuamente información acerca del estado de temperatura en los 12 medidores, y posicionamiento del carro de avance. A medida que los datos son recabados, se almacenan y se muestran el entorno gráfico en modo de colores sobre el diagrama del horno.

Se hace uso de librerías de código libre para el manejo de la información. Entre ellas *“qcustomplot*”, para generar gráficas de línea, y “*QtXlsxWriter-master*”, para manipular y generar archivos .xlsx. La clase “qpainter” forma las imágenes de térmicas. Además, la clase que está basada en el protocolo Modbus (*modbusmaster.h*) hace uso de la librería “*QtcpSocket.h*” del entorno libre Qt para establecer el enlace en el estándar ethernet.

### Generación de una trayectoria.

Con una trayectoria nos referimos a una secuencia de movimientos predeterminados, cuyos parámetros, como la duración en el tiempo o la distancia son colocados por el usuario en una secuencia descripta en una lista. La lista está formada por comandos, que hacen referencia a una acción específica. Estos comandos reciben datos numéricos que representan las dimensiones del movimiento o temperatura. Además, se programan los controladores para actuar en un tiempo establecido en la secuencia o, cuando se cumpla una determinada condición.

La función del usuario en éste entorno es la de describir una secuencia que la máquina sea capaz de ejecutar paso por paso. Para ello, se escriben en un orden particular uno a uno los comandos en la lista, de ser necesario se fijan distancias, tiempos o grados de temperatura a los cuales se desea alcanzar en ese punto de la experiencia. Una vez escrita, se debe pre evaluar su viabilidad mediante la ejecución de un sistema que analiza la combinación de comandos al apretar el botón “Analizar Trayectoria”. Si el análisis devuelve un resultado correcto, se dibujan en las gráficas “Distancia Vs Tiempo” el desplazamiento del actuador a lo largo de la carrera en función del tiempo, y en “Temperatura Vs Tiempo” la distribución final de la temperatura longitudinalmente en el horno. Existe la opción de probar únicamente la ejecución de movimiento al hacer click en el botón “Probar trayectoria”. Éste envía solo la secuencia de movimiento a la máquina para que realice los movimientos pertinentes.

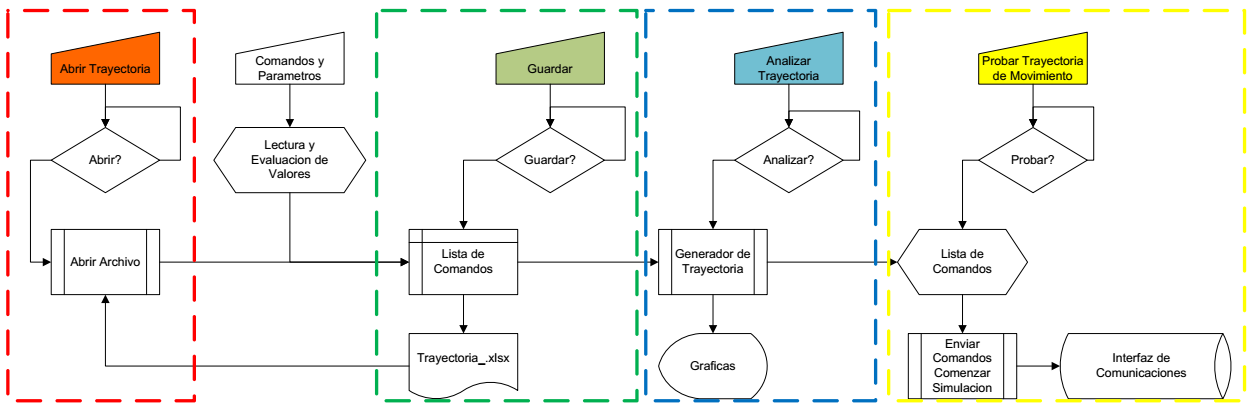


Diagrama : Interfaz de generación de trayectoria.

Es posible guardar la lista confeccionada en un archivo de tipo \*.xlsx que en su nombre contiene el prefijo “*Trayectoria\_*” para su posterior utilización en una simulación o bien para modificarlo según sea necesario. Entonces, éste programa posee la capacidad también de poder abrir los archivos anteriormente generados y pre evaluados.

Un diagrama que ilustra su funcionamiento se observa en la figura 28. Allí se observa que la secuencia de pulsado de cada botón al principio debe ser consecutiva.

* Primero se debe confeccionar una nueva trayectoria o bien, abrir una ya realizada para modificar.
* Segundo se debe guardar y/o analizar trayectoria pueden realizarse a la vez o alternadamente. Sin embargo no se puede probar trayectoria de movimiento sin previamente haber generado un análisis de ésta. Tampoco es posible observar una pre visualización gráfica sin ésta acción.

En la figura, también se observa en detalle los componentes de la sección. El ingreso de línea de comandos está constituido por una lista desplegable con todos los comandos habidos y dos campos de ingreso de datos que se habilitan según el comando seleccionado lo requiera.

Los botones de edición de línea de comando permiten agregar una línea nueva a partir del ingreso, editar el contenido de ésta al seleccionarla y presionar “Editar”, “Borrar” una línea por completo, “Subir” y/o “Bajar” una línea ya agregada. Las gráficas de Posición en función del tiempo y temperaturas en función del tiempo, son estimadas, por eso están compuestas por segmentos de línea recta. El desarrollo de la lista de comandos se observa en la tabla.

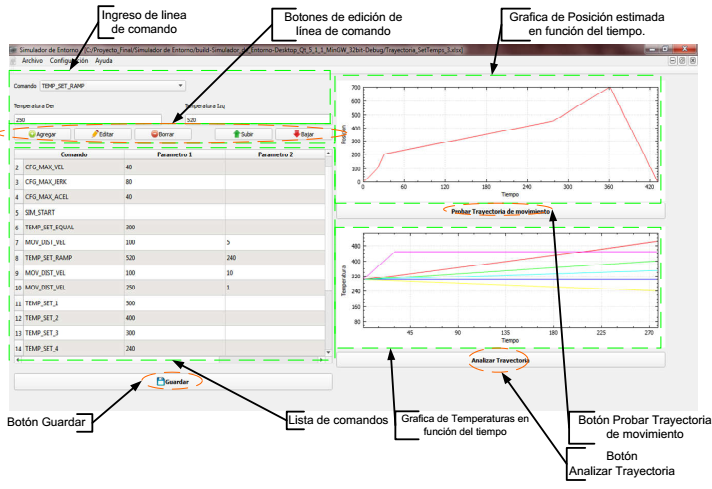


Figura : Interfaz de generación de trayectoria.

### Simulación de la trayectoria programada.

En la sección del programa donde se lleva a cabo la simulación completa de la trayectoria, se ubican visualizadores de valores de proceso que indican el estado del sistema a medida que transcurre el tiempo o actúan los subsistemas.

La tarea del usuario es seleccionar una trayectoria previamente generada, que se encuentra almacenada en un archivo de extensión \*.xlsx. Una vez seleccionada se pre visualizan los movimientos y temperaturas de referencia para el experimento, entonces estamos en condiciones de llevar a cabo la prueba.

Al ejecutar la simulación, la computadora comienza a transferir la información en forma de comandos en registros Modbus para que la máquina comience con su ejecución.

De manera continua, una tabla de registros con estados del proceso de simulación es actualizada en una lectura cada segundo, para poder así graficar las evoluciones de posición y temperatura en el tiempo, además de reconocer cuando inicializa y finaliza la simulación.

Al finalizar el proceso, el usuario puede elegir guardar los datos obtenidos en un archivo de extensión \*.xlsx. Estos datos son obtenidos del muestreo que llevó a cabo la máquina y se encuentran almacenados en una memoria Flash interna, a la que solo el usuario desde el ordenador tiene acceso a través del software, que es el encargado de realizar la gestión de permisos y selección de información a leer, basados en el uso del protocolo Modbus.

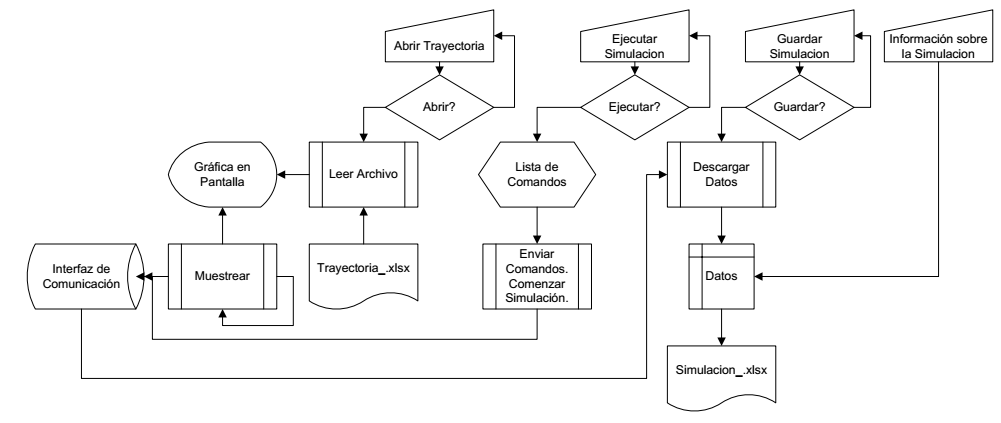


Diagrama : Interfaz de simulación.

Según el diagrama 15, desde el comienzo, siempre que la comunicación se haya establecido, es posible observar en pantalla el estado de los distintos subsistemas de la máquina. Pero, antes de realizar una simulación, se debe abrir un archivo generado en la sección anterior, que contenga la trayectoria deseada. Una vez seleccionada la trayectoria, se es capaz de poder comenzar la simulación al poder enviar la lista de comandos. La acción se realiza al presionar el botón “Ejecutar Simulación”. El experimento no tiene una duración fija conocida, por lo que el soft continuamente verifica su finalización, al leer el registro que lo indica, en el proceso de muestreo. Una vez que se da por finalizada la experiencia, se habilita el botón de “Guardar”, y con ello el proceso de descarga de datos desde la memoria interna del controlador principal para ser almacenados en un archivo .xlsx junto con los campos de detalles de identificación. Los campos de identificación son:

* El título de la experiencia;
* La fecha en la que se lleva a cabo;
* La hora de inicio;
* La temperatura ambiente en ese momento;
* El operario a cargo de la experiencia;
* Una breve descripción de la experiencia que sirva de ayuda para recordar las condiciones en la que se desarrolló.

En la interfaz de usuario observada en la figura 29, se aprecian los elementos que la componen:

* Barra de menús: corresponden al programa principal que contiene las secciones “Nueva Simulación”, “Nueva Trayectoria”, “Abrir Trayectoria”, además de las secciones de ingresos de parametros de configuracion y ayuda.
* Imagen térmica generada: ilustra la sección transversal del horno, discriminando las temperaturas en colores, las temperaturas más frías se muestran en color azul, mientras que las mas calientes en rojo, pasando primero por el celeste, verde y amarillo.
* El display de indicación de posición del actuador: visualiza el desplazamiento en tiempo real en centésimas de milímetro, desde la posición cero de arranque, hasta un valor máximo de carrera de 1500,00 milimetros.
* Botón “Cargar Trayectoria”: ejecuta un diálogo abrir, para seleccionar la trayectoria que deseemos llevar a cabo.
* Campos descriptivos de simulación: es el conjunto de textos que describen la situación cuando se ejecute el ensayo.
* Botón “Ejecutar simulación”: inicia la simulación.
* Botón Guardar: inicia la descarga de los datos obtenidos del procesador de muestreo, almacenados en la memoria, para su inmediato guardado en el archivo de simulación.
* Displays de temperatura deseada y temperatura de proceso, se refrescan cada segundo, son doce en total y se corresponden dos por cada etapa del horno.
* Área de pestañas de datos: está compuesta por cuatro pestañas, las primera corresponde a la gráfica de posición del actuador en el tiempo. La segunda pestaña corresponde a la gráfica de los seis valores de proceso en el tiempo de ejecución. La tercera pestaña corresponde a la tabla de registros que son actualizados cada segundo, de donde se obtiene la información de los estados de la máquina. La cuarta pestaña es una tabla que muestra los valores de las muestras al momento de descargarlas cuando finaliza la simulación. En ella se observa el tiempo, la posición, las seis temperaturas por sección y doce registros correspondientes a variables del sensor a calibrar.
* Barra de estado: está presente en el programa principal, y muestra información acerca de distintos procesos efectuados.

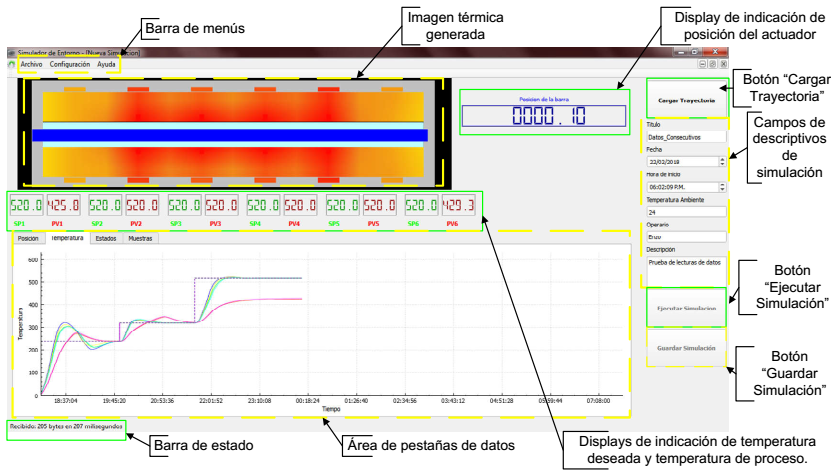


Figura : Interfaz de simulación.

### Ingreso de parámetros de configuración.

Los módulos a configurar son cuatro: Controladores de temperatura, parámetros ethernet Tcp-ip de la máquina, Conexión ModbusTcp y Configuración del sensor a calibrar. Cada uno de éstos recibe una determinada cantidad y tipos de parámetros que son enviados a la máquina, es decir, escritos en la tabla de registros modbus, en su correspondiente dirección de memoria, para que el controlador principal se encargue de estructurarlos en sus sistemas. Los distintos campos de ingresos se encuentran disponibles en el menú “Configuración” de la barra de menús.

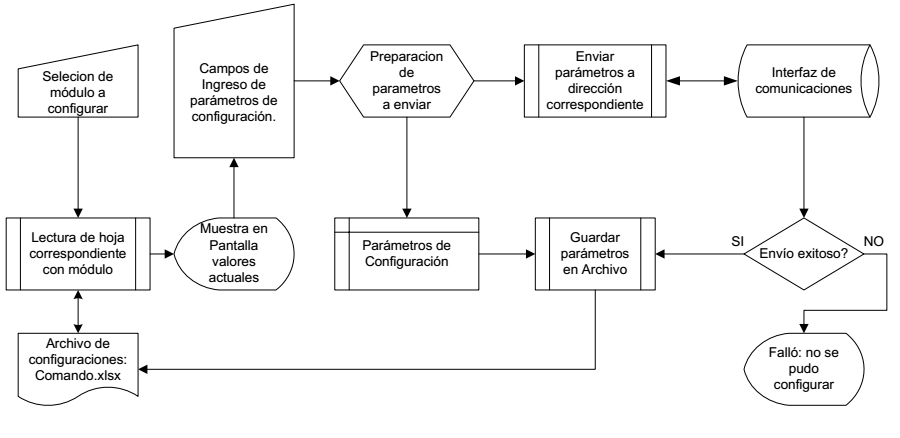


Diagrama : Interfaz de comunicación.

El programa guarda los mismos parámetros en distintas hojas de un archivo, cada cual corresponde a un módulo de configuración. Sólo si el envío de los parámetros de configuraciones es exitoso, se guarda la copia actualizada de los parámetros. El archivo que contiene esta información es Comando.xlsx y cada vez que se selecciona un módulo a configurar, se leen en él las configuraciones previas para ser visualizadas. Si el envío no pudo haberse realizado por algún motivo, las configuraciones no se actualizan en el archivo y un mensaje de falla aparece en la barra de estado.

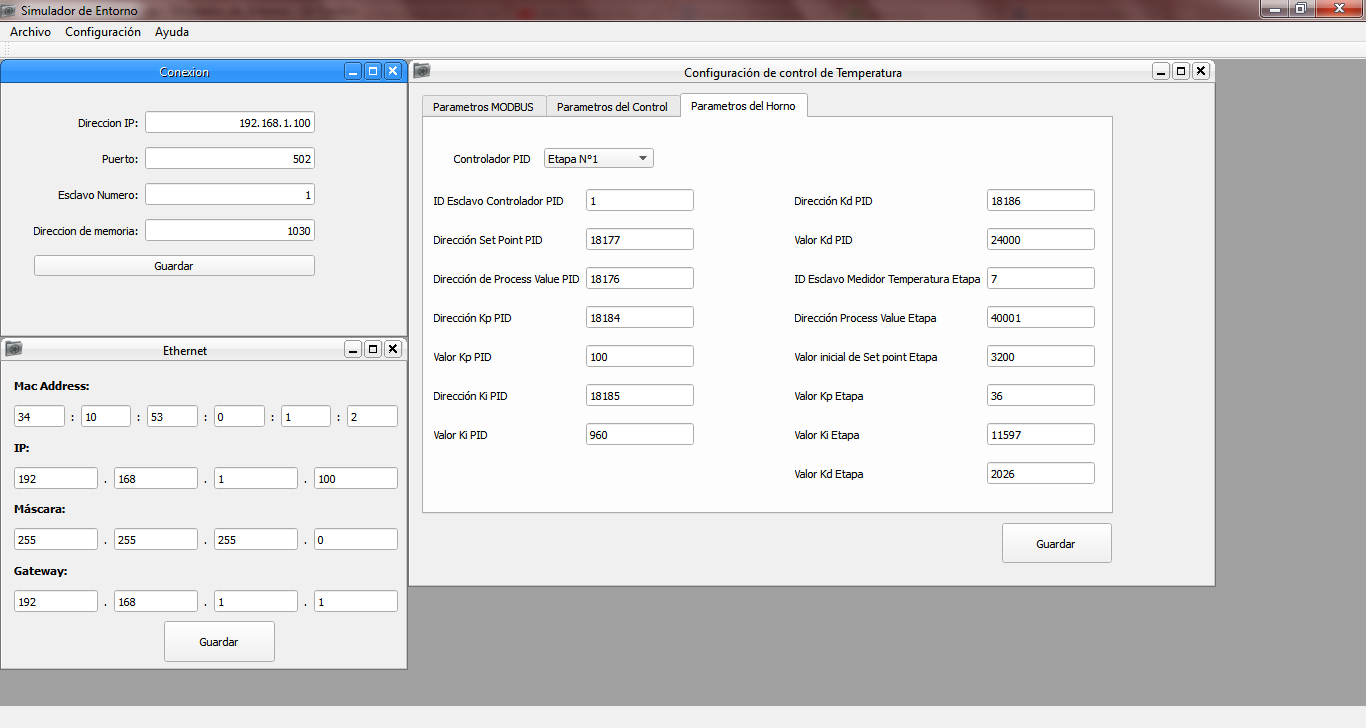


Figura : Interfaz de configuración

### Almacenamiento de datos generados.

Los parámetros de configuración, los comandos de control con sus respectivas características (valores máximos, mínimos, sensibilidad, valor numérico, etcétera), y demás características, se almacenan en un archivo en común, en distintas hojas. Es el archivo “*Comando.xlsx”* y se lee cada vez que se ejecuta una nueva ventana de simulación, trayectoria o configuración. Para el acceso a este se hace uso de la librería “*Qt XlsxWriter-master*”. El archivo es esencial para el funcionamiento del programa en su totalidad, de otra forma es imposible que se ejecute.

### Resultados parciales.

Con respecto al software desarrollado, se obtuvieron buenos resultados en las comunicaciones con el protocolo Modbus Tcp. En una respuesta de 205 bytes, donde 9 son de encabezado y 196 bytes útiles, se obtuvieron tiempos de 18 milisegundos, es decir 87.1 [kbps] (196[B]\*8(b/B)/0.018[s]). Inclusive se efectúa una notable mejora cuando los volúmenes de información son mayores, donde el encabezado representa una mínima proporción de información respecto de la trama y los tiempos de respuesta se vuelven mínimos. Las pruebas se realizaron con un conmutador de red de por medio, que es como se planteó desde un principio, sin notarse cambios en el uso de distintos adaptadores de red (Ethernet y Wifi). Pero, por seguridad se elige que el modo de trabajo sea alámbrico. Se comprobó con software de captura de paquetes (WireShark) que las tramas son reconocidas como Modbus Tcp. El ordenador no presentó ser un retraso o “cuello de botella” en las comunicaciones. Sin embargo, se debe tener un total cuidado en el hilo de ejecución de una secuencia de Modbus. Una vez enviado un paquete de requerimiento de lectura/escritura, el ciclo de espera debe tener dos condiciones de salida, una debe ser hasta que el buffer de entrada alcance la cantidad de datos esperada, o bien la otra hasta cumplido un cierto tiempo máximo de espera, además de que se debe otorgar una prioridad mínima al hilo de ejecución, de lo contrario el programa en el ordenador no responde causando fallas y la perdida de información. Debido a causas como ésta y otras más, sobre todo en la interfaz de usuario, se le otorgó al programa la robustez necesaria para operar en un entorno de trabajo continuo de largas horas o días y la practicidad de poder recuperarse ante fallas e informar la causa.

# Capítulo 3: Resultados.

Los resultados generales luego de la finalización de este proyecto, fueron satisfactorios, llegando en algunos casos a superar los objetivos propuestos e inclusive las expectativas de los desarrolladores.

La implementación del hardware sobre el dispositivo FPGA, si bien propuso un gran desafío de diseño, resultó una opción adecuada para enfrentar las dificultades que proponía el desarrollo. Su uso permitió la implementación del sistema multi procesador, que le brinda a la maquina una gran capacidad sin necesidad de un hardware electrónico excesivamente potente. Esto a su vez permitió la independencia de las principales funciones de unidades de control externas como la PC, brindando mayor seguridad y robustez.

El controlador de temperatura en su conjunto hardware mecánico, electrónico y firmware, logró controlar exitosamente la temperatura con un rango de variación medible de 0,1°C, muy por encima de los 2,5°C propuestos, si bien su desarrollo resultó más complicado de lo esperado, se lograron los resultados esperados.

El controlador de movimiento, en su conjunto, actuador mecánico, hardware y firmware, corresponde con los parámetros requeridos por el equipo de diseño, permite adicionalmente la variación de los parámetros de Jerk y Aceleración que en un principio se consideraban constantes además su precisión se encuentra un orden de magnitud por encima de la requerida (0,02[mm]). Se lograron estas características, gracias al correcto diseño mecánico, la selección adecuada de los elementos de actuación y al sistema de control implementado.

De igual manera el sistema de adquisición de datos se comporta según lo requerido, el agregado de funciones de muestreo adicional provee una forma más dinámica y eficiente de obtención de los datos.

Las interfaces de usuario, gráficas y almacenamiento de información no presentaron complicaciones y resultaron ser muy prácticas e importantes, ya que permitieron una tratativa de la información muy eficiente. La clase que compila la lista de comandos, analiza la viabilidad de la trayectoria y prepara la información para su envío, es de vital importancia para el software en general. Ningún tipo de proceso de simulación debe realizarse sin antes haberse compilado la trayectoria en las funciones de análisis, esto asegura el correcto funcionamiento de la maquina durante la simulación.

# Capítulo 4: Análisis de Costos.

Según lo planteado en el anteproyecto, los recursos materiales para el diseño y desarrollo del proyecto, se obtuvieron a través de empresas especializadas dentro del país. Los componentes electrónicos, sin embargo, son importados y dado el aumento en el precio del dólar estadounidense, el valor en pesos se vio afectado respecto de lo previsto.

En cuanto al tiempo de desarrollo la mayor parte de las unidades se mantuvieron dentro de los estimados, sin embargo una en particular, el controlador de temperatura superó ampliamente el tiempo estimado.

Análisis de costos finales

|  |  |  |
| --- | --- | --- |
| **Categoría** | **Subcategoría** | **Costo** |
| **Hardware Mecánico** |  |  |
|  | Diseño | $ 4.830,00 |
|  | Materiales | $ 14.358,00 |
|  | Maquinado | $ 2.435,00 |
|  | Montaje | $ 4.480,00 |
|  | Unidad Motora | $ 1.338,00 |
|  | Unidad de Contraste | $ 2.040,00 |
|  | Transporte | $ 824,00 |
|  |  |  |
| **Hardware Electrónico** |  |  |
|  | Diseño |  |
|  | Fuente de poder | $ 2.200,00 |
|  | Control de Potencia | $ 2.180,00 |
|  | Control Digital | $ 9.200,00 |
|  | Cableado | $ 935,00 |
|  |  |  |
| **Software Electrónico** |  |  |
|  | Diseño | $ 14.293,00 |
|  | Implementación | $ 28.586,00 |
|  | Verificación | $ 8.875,00 |
|  |  |  |
| **Software HID** |  |  |
|  | Diseño | $ 17.552,00 |
|  | Implementación | $ 26.104,00 |
|  | Verificación | $ 9.480,00 |
|  |  |  |
| **Total** |  | $ 149.710,00 |
|  |  |  |
| **Beneficio** | 20% | $ 29.942,00 |
| **I.V.A.** | 21% | $ 37.726,92 |
|  |  |  |
| **Precio de mercado** |  | $ 217.378,92 |
|  |  |  |
| **Costo de comercialización** |  | $ 29.895 |
|  |  |  |
| **Precio final** |  | $ 247.273,92 |

Cabe destacar que el valor de los ítems se calculó en base a la remuneración establecida para cada rubro por el CIEER (Colegio de Ingenieros Especialistas de Entre Ríos) para cada hora de trabajo, según la fecha en la que se realizó el mismo.

Para la implementación de este primer modelo, se contó con una financiación del costo de materiales proveniente del cliente, mientras que el resto se percibe en dos veces, la mitad al inicio en concepto de adelanto y el resto al entregar el trabajo. Este método de cobro resulta adecuado para un primer modelo y permitió un razonable manejo del presupuesto.

El principal factor de comparación durante el desarrollo proviene del costo de una unidad similar de simulación cotizada por una universidad inglesa, su valor final fue de US$ 120.000 este desarrollo proponía algunos aspectos superiores al logrado, sin embargo la diferencia de costos vuelve el producto terminado, una excelente opción para el mercado nacional, que adicionalmente nos ubica dentro de un mercado muy poco explorado y con grandes posibilidades a futuro.

# Capítulo 5: Discusión y Conclusión.

Para realizar una discusión completa de los resultados planteados en el capítulo 3, recordemos los objetivos planteados en el comienzo del trabajo:

* Simulación de Temperatura, capacidad de realizar calentamientos lineales y gradientes de la camisa del sensor, de manera automática.

Se logró cubrir ampliamente este punto con un sistema que entrega los resultados esperados, sin embargo se destaca en este punto en particular, la alta complejidad que requirió conseguir el método de control adecuado, la solución mediante la aplicación del control en escalera, permitió un manejo muy preciso, pero a la vez implico una calibración del sistema que requiere de mucho tiempo y la implementación adicional de 6 controladores PID a medida, que resultó posible solo gracias a la elección del hardware electrónico de soporte.

Se plantea a futuro, el análisis de los componentes del hardware mecánico que permitan un tiempo de establecimiento menor de los parámetros del sistema.

* Simulación de movimiento, realizar el posicionamiento de la barra de control en todo el rango de estudio, de forma repetitiva entre ensayos.
* Simulación de la dinámica de movimiento, controlar velocidad y aceleración de los movimientos. Realizar movimientos complejos de forma repetitiva.

Respecto del diseño de hardware mecánico, el actuador de movimiento fue la primer unidad encarada, el conocimiento previo y la disponibilidad de maquinaria adecuada para su fabricación permitieron un resultado remarcable en cuanto a precisión, capacidad de uso y prestaciones mecánicas e incluso estéticas, la elección de los materiales fue acertada, aun cuando provocaron un impacto importante en los costos del proyecto, pudimos comprobar que la base del éxito en el sistema de movimiento se encuentra íntimamente relacionada con la construcción de un buen hardware.

El sistema de control encarado presento un desafío de desarrollo que se vio ampliado por la necesidad de ser implementado en hardware, sin embargo las prestaciones fueron notablemente superiores a las esperables de un sistema microcontrolado.

Se plantea a futuro la colocación de un sistema de contraste más sensible, que permita el control de la posición de stall del motor y la salida de la zona de resonancia que no es posible con el encoder actual.

* Calibración, capacidad de contraste un orden de magnitud por encima de la precisión deseada del sensor.

Como se mencionó anteriormente la precisión en el movimiento resulta suficiente para la calibración, sin embargo en este punto, por limitaciones económicas, se seleccionó un elemento de contraste para la calibración no óptimo, para desarrollos futuros el costo de estos elementos tendrá que ser considerado desde el inicio permitiendo así un margen de selección más amplio.

* Adquisición periódica de datos, luego de cada ensayo se deberá disponer de los datos completos de la simulación.
* Manejo de datos, contar con un sistema de almacenamiento sistematizado de los ensayos.

El manejo de datos, su flujo y almacenamiento sistematizado, que se pretendía como objetivo resulto ser totalmente exitoso. Los tiempos de comunicación fueron los esperados y los archivos obtenidos al finalizar cada ensayo se guardan ordenadamente por fecha. Estos elementos contienen cantidades significativas de información en forma de arreglos, correctamente presentadas con rótulos identificativos, para su posterior tratamiento. Si bien el tiempo de descarga de los datos desde la memoria en la máquina, hasta su guardado en archivo es considerable, es el normal para la tecnología utilizada y los volúmenes de información manipulados. Las prestaciones pretendidas por parte del software en estos conceptos son realmente satisfactorias, logrando cumplir por completo las necesidades expresadas.

* Control, capacidad de controlar todos los aspectos del ensayo desde una PC.

El control del sistema fue eficaz una vez que se establecieron ciertas reglas de procedimiento para el inicio de una simulación, la correcta configuración y la lectura de la información. Gracias a esto se obtuvo mayor seguridad al momento de establecer un parámetro de control. Además, gracias al análisis de la trayectoria, la mayor parte de los cálculos se realizan en el ordenador, aliviando significativamente la carga aritmética del procesador de movimiento, transmitiendo el resultado ya pre procesado.

La interfaz de usuario es bastante intuitiva, considerando que será utilizada por personal capacitado y con experiencia en este tipo de entorno. Las gráficas de curvas, las imágenes generadas y los paneles digitales son de gran aporte y visualización, permiten un dimensionamiento y caracterización del comportamiento de las distintas variables durante el transcurso del ensayo, resultando en su mejor comprensión y análisis.

Se pretende en un futuro próximo la mejora de la eficiencia de las líneas de código correspondiente a la trasmisión para lograr mayor rapidez en el flujo de información. Así como también nuevos comandos de trayectoria que permitan realizar movimientos más complejos o configurar diversos parámetros durante la simulación. Luego, ventanas de ayuda que describan el funcionamiento de los comandos disponibles, del mismo modo que una hoja de datos de un set de instrucciones de cualquier máquina.

Un planteo a futuro, sería poder disponer de los mismos datos y entorno grafico en un servidor web para su acceso remoto, con niveles de seguridad correspondientes, y así poder monitorear ensayos de larga duración de manera remota, reduciendo los turnos extendidos que realizan los usuarios, o al menos permitiéndoles realizarlos desde sus hogares.

# Capítulo 6: Literatura Citada.

Adam, E. (2011), *Instrumentación y control de procesos*, Santa Fe Argentina, Ediciones UNL.

Esparza, D. A. (1999), *Sensor inductivo para conocer la posición de las barras de control del reactor nuclear CAREM*, AATN'99, Noviembre, Bariloche Argentina.

Lion, K.S. (1969), *Transducers—problems and prospects*, IEEE Transactions on Industrial Electronics, 16, 2–5.

Mori, Saito, Goshima, Yanagihara, Tanaka, Fraser, Joe, Nitta, Tomita, (1993) .*A distributed shared memory multiprocessor*, ACM/IEEE conference on Supercomputing, Oregon, Estados Unidos.

Moro Piñeiro, M. (2000), *Metrología: introducción, conceptos e instrumentos*. Universidad de Oviedo, Asturias, España.

Chen Y., Wang T., Wei H. (2006), *Linear and Sshape Acc/Dec for CNC machine*, China, Mech. Eng.

Ogata, K. (2010), *Ingeniería de control moderna.* Madrid España, Pearson Educación S.A.

Universidad Tecnológica Nacional

Facultad Regional Paraná

Se certifica que …..................................................... , DNI: …........................... ha realizado la dirección del Proyecto Final:

…........................................................................................................................................................................................................................................................................................................................................

De los alumnos:

* ….............................................................
* ….............................................................
* ….............................................................

Realizada durante el ciclo lectivo: ..............., obteniendo el grupo un calificación final de: …..................

A fin de ser emitida la correspondiente certificación por el departamento de electrónica, se extiende la siguiente constancia.

***Pañoni Sergio Ramos Hector Maggiolini Lucas***