



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“SISTEMA DE CONTROL DE MOVIMIENTO MONOEJE DE
ALTA PRECISIÓN Y VERSATILIDAD”**

INFORME DE PROYECTO INTEGRADOR

Previo la obtención del Título de:

**INGENIERO EN ELECTRICIDAD ESPECIALIZACIÓN
ELECTRÓNICA Y AUTOMATIZACIÓN INDUSTRIAL**

GABRIEL ANDRÉS GALLARDO MONCAYO

JUAN ANDRÉS SAMANIEGO ASANZA

GUAYAQUIL - ECUADOR

AÑO: 2018

AGRADECIMIENTOS

Agradezco a mis padres por darme la oportunidad de estudiar esta carrera. Al Ing. Alberto Larco por su colaboración en el presente proyecto. A la ESPOLE por el conocimiento y los recursos compartidos. Y a todos los docentes, compañeros y amigos que han influido en mi desarrollo personal y académico,

Gabriel Gallardo Moncayo

AGRADECIMIENTOS

Dejo constancia del más sincero agradecimiento a la universidad por haberme acogido y brindado su apoyo para lograr mi objetivo. A todos los maestros que con dedicación y misión supieron guiarme con su deber de educadores me condujeron por el noble camino del saber.

Con el cariño más grande y sincero, mi reconocimiento, agradezco al Ing. Alberto Larco que con sus amplios conocimientos contribuyo a la realización y culminación del presente proyecto.

A los señores rectores y más autoridades, docentes de dicha institución educativa, que hicieron posible la recolección de información necesaria para realizar este proyecto, a nuestros familiares quienes nos han brindado su apoyo moral y la esperanza de verme culminar la carrera universitaria.

Agradezco también a todas las personas quienes de una u otra manera contribuyeron para llegar con entusiasmo a la meta.

Juan Andrés Samaniego

DEDICATORIA

Este trabajo se lo dedico a mi hermana y padres, por su amor y confianza a lo largo de mis años de estudio.

Gabriel Gallardo Moncayo

DEDICATORIA

El presente proyecto lo dedico a Dios, a mi madre y padre, abuelos, tíos, primos maternos. Al Dr. René Ortega y flia., Sra. Fanny Aguirre y flia. y al Sr. Armangel Aguilar y flia. prueba de su inmenso cariño quienes han estado pendientes y me han brindado su apoyo incondicional, sacrificio de ello ha sido una base fundamental para culminar mi carrera y con ello hoy sea un ente útil a la sociedad y la patria.

Juan Andrés Samaniego

TRIBUNAL DE EVALUACIÓN

Ing. Damian Alberto Larco Gomez
PROFESOR TUTOR

Ing. Janeth Carolina Godoy Ortega
PROFESOR MATERIA INTEGRADORA

DECLARACIÓN EXPRESA

“La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la propiedad intelectual”

Gabriel Andrés Gallardo Moncayo

Juan Andrés Samaniego Asanza

RESUMEN

El uso de robots en la industria está aumentando año tras año y es necesario que los futuros ingenieros puedan diseñar estas soluciones para el mercado local. En este proyecto se plantea el desarrollo de un prototipo de máquina industrial de un solo eje lineal que tiene como finalidad servir de guía para el diseño de robots cartesianos de mayor tamaño. Se debe tener un control preciso en las variables de posición y velocidad del eje mientras se integra una interfaz de usuario que facilite el trabajo del operador agregando versatilidad al sistema. Se utilizó un controlador industrial de gama alta, que incluye funciones para la implementación del control de movimiento, en conjunto con un accionamiento con funciones de servocontrol y un servomotor síncrono de imanes permanentes de alta dinámica. Todos los equipos fueron comunicados con protocolos industriales de alta velocidad, se agregó una pantalla táctil para la interfaz de usuario y algunos sensores que generan valor agregado y seguridad al sistema. La programación permitió operar y cambiar fácilmente entre los modos manual y automático, y editar rutinas de movimiento existentes sin tener que detener el trabajo del motor, además, la precisión obtenida fue de 0.01 mm tanto en altas como en bajas velocidades. Se concluye que el sistema mantiene una precisión excelente incluso con cargas variables, su consumo energético es relativamente bajo y su interfaz permite reducir los tiempos de producción y operación.

Palabras clave: control de movimiento, comunicación industrial, programación de PLC, interfaz de usuario, equipos electrónicos industriales.

ABSTRACT

The use of robots in industry is increasing every year and it is necessary for engineers to know how to design these solutions for local market. In this project it is proposed the develop of an industrial machine prototype with a linear axis that has the purpose of being a guide for the design of bigger cartesian robots. It must have a precise control for position and speed variables of the axis while it integrates a user interface to ease the operator's job adding versatility to the system. There was used a high range industrial controller, that includes motion control functions, together with a servocontrol driver and a high dynamic permanent magnet synchronous servomotor. All equipment was communicated by high speed industrial protocols, there were added a touch panel for the user interface and some sensors to generate value and security for the system. The programming let the user to easily operate and change between manual and auto modes, and to edit movement routines without stopping the motor, also, the precision obtained was 0.01 mm both in high and low speed. It is concluded that the system keeps an excellent precision even in changing loads, its energy consumption is relatively low, and its user interface let to decrease work and production times.

Keywords: *motion control, industrial communication, PLC programming, user interface, industrial electronic equipment.*

ÍNDICE GENERAL

RESUMEN.....	I
ABSTRACT	II
ÍNDICE GENERAL	III
ABREVIATURAS	VI
SIMBOLOGÍA	VII
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS.....	X
CAPÍTULO 1.....	1
1. INTRODUCCIÓN	1
1.1. Descripción del problema.....	1
1.2. Justificación del problema	1
1.3. Objetivos	2
1.3.1. Objetivo General.....	2
1.3.2. Objetivos Específicos	2
1.4. Marco teórico.....	3
1.4.1. Control de Movimiento.....	3
1.4.2. Aplicación del control de movimiento en máquinas cartesianas	5
CAPÍTULO 2.....	7
2. METODOLOGÍA	7
2.1. Concepción del sistema	7
2.1.1. Servomotor	8
2.1.2. Selección del accionamiento adecuado.....	9
2.1.3. Selección del controlador de alto nivel	10
2.2. Objeto tecnológico: Eje de posicionamiento.....	11
2.2.1. Configuración de hardware.....	12

2.2.2.	Límites de Posición.....	14
2.2.3.	Referenciar el eje.....	15
2.2.4.	Lazo de control	16
2.3.	Periferia	17
2.3.1.	Sensores para los límites de posición	17
2.3.2.	Sensor para la referencia	18
2.3.3.	Elemento terminal.....	19
2.3.4.	Manejo manual del sistema	20
2.3.5.	Diagrama eléctrico de la periferia	22
2.4.	Topología de comunicación	23
2.5.	Programación del sistema.....	24
2.5.1.	Instrucciones para el control de movimiento.....	25
2.5.2.	Modo de referencia.....	26
2.5.3.	Modo manual discreto	26
2.5.4.	Modo manual continuo	26
2.5.5.	Modo automático	26
2.5.6.	Importar y exportar programas	28
2.6.	Interfaz de usuario.....	30
2.6.1.	Panel de Mando.....	30
2.6.2.	Edición de programas.....	31
CAPÍTULO 3.....		32
3.	RESULTADOS Y ANÁLISIS	32
3.1.	Lazos de control interno	32
3.1.1.	Calibración del regulador de intensidad.....	32
3.1.2.	Calibración del regulador de velocidad.....	33
3.2.	Resultados y calibración del regulador de posición.....	35
3.3.	Análisis de precisión del sistema.....	38

3.4. Análisis de versatilidad del sistema.....	40
3.5. Análisis de consumo de energía y factor de potencia	43
3.6. Costo monetario de la implementación del prototipo.....	44
3.7. Análisis de costos del proyecto escalado	46
3.8. Factibilidad y rentabilidad.....	49
CAPÍTULO 4.....	50
4. CONCLUSIONES Y RECOMENDACIONES.....	50
4.1. Conclusiones.....	50
4.2. Recomendaciones.....	54
BIBLIOGRAFÍA.....	57

ABREVIATURAS

DC	Corriente Directa
AC	Corriente Alterna
PMSM	Motor Síncrono de Imanes Permanentes
IM	Máquina de Inducción
SM	Máquina Síncrona
FOC	Control de Campo Orientado
VDC	Voltaje de Corriente Directa
PLC	Controlador Lógico Programable
HMI	Interfaz Hombre Máquina
CPU	Unidad de Procesamiento Central
IRT	Tiempo Real Isócrono
DSC	Servo Control Dinámico
IP	Protocolo de Internet
KOP	Programación por Diagrama de Contactos
SCL	Programación en Lenguaje Estructurado
FUP	Programación por Diagrama de Funciones
PI	Proporcional Integrador
VPN	Red Privada Virtual
CAD	Diseño Asistido por Computadora
CNC	Control Numérico Computarizado
3D	Tres dimensiones
NC	Normalmente Cerrado
SD	Seguridad Digital
CSV	Valores separados por comas

SIMBOLOGÍA

W	Vatios
kW	Kilovatios
HP	Caballos de fuerza
A	Amperios
V	Voltaje
mm	Milímetros
mm/s	Milímetros por segundo
µm/s	Micrómetro por segundo
rpm	Revoluciones por minuto
ms	Milisegundos
kg	Kilogramos
°C	Grados centígrados

ÍNDICE DE FIGURAS

Figura 2.1 Esquema general del sistema de control de movimiento.....	7
Figura 2.2 Motor con conectores y acople al tornillo	8
Figura 2.3 Tornillo sin fin	9
Figura 2.4 Opciones para el accionamiento del motor.....	10
Figura 2.5 Controlador de alto nivel utilizado.....	11
Figura 2.6 Catálogo de objetos tecnológicos disponibles	12
Figura 2.7 Diagrama de comunicación entre el controlador y el accionamiento	12
Figura 2.8 Configuración del telegrama.....	13
Figura 2.9 Asignación de las palabras del telegrama	14
Figura 2.10 Configuración de los límites de posición.....	15
Figura 2.11 Rutina para referenciar el eje	16
Figura 2.12 Lazo de control del sistema	17
Figura 2.13 Interruptores de final de carrera	18
Figura 2.14 Sensor láser instalado	19
Figura 2.15 Luz piloto como elemento terminal	20
Figura 2.16 Mando para control manual	21
Figura 2.17 Diagrama eléctrico de conexiones del PLC y la periferia.....	23
Figura 2.18 Topología de comunicación.....	24
Figura 2.19 Algoritmo general del programa	25
Figura 2.20 Catálogo de instrucciones para el control de movimiento.....	26
Figura 2.21 Variables de tipo Step_info.....	27
Figura 2.22 Algoritmo del modo automático	28
Figura 2.23 Diagrama de bloques para la lectura y escritura en la memoria no volátil ..	29
Figura 2.24 Diferencias entre la interfaz antigua y definitiva	30
Figura 2.25 Pantalla de edición de programas	31
Figura 3.1 Diagrama de parámetros del regulador de intensidad	33
Figura 3.2 Respuesta del regulador de intensidad	33
Figura 3.3 Diagrama y parámetros del regulador de velocidad	34
Figura 3.4 Respuesta del regulador de velocidad	35
Figura 3.5 Diagrama y parametrización del regulador de posición	35
Figura 3.6 Respuesta del regulador de posición	36

Figura 3.7 Comparación de respuesta en vacío y con carga.....	36
Figura 3.8 Retardo en el envío de la consigna	37
Figura 3.9 Respuesta de velocidad desde el PLC	37
Figura 3.10 Medición del desplazamiento lineal con sensor láser	38
Figura 3.11 Retardo de acción del modo manual	39
Figura 3.12 Retardo de acción del modo automático	40
Figura 3.13 Pantallas de inicio y panel de mando	41
Figura 3.14 Pantalla de programación.....	42
Figura 3.15 Servidor Web del PLC	42
Figura 3.16 Edición de archivos .csv	42
Figura 3.17 Equipo para la medición de energía	43
Figura 3.18 Mediciones de corriente.....	44
Figura 3.19 Mediciones de energía	44
Figura 3.20 Robot Reis.....	47

ÍNDICE DE TABLAS

Tabla 2.1 Configuración de Hardware	12
Tabla 2.2 Escala de sensibilidad para mando manual.....	21
Tabla 2.3 Asignación de entradas y salidas del PLC.....	22
Tabla 2.4 Asignación de direcciones IP.....	24
Tabla 3.1 Costo de equipos y materiales.....	45
Tabla 3.2 Costos de la mano de obra.....	46
Tabla 3.3 Costos totales del prototipo	46
Tabla 3.4 Parámetros generales de la máquina	47
Tabla 3.5 Parámetros eléctricos de la máquina.....	47
Tabla 3.6 Costo de equipos y materiales del sistema escalado	48
Tabla 3.7 Costos de mano de obra para sistema escalado	49
Tabla 3.8 Costo total del sistema escalado	49

CAPÍTULO 1

1. INTRODUCCIÓN

El siguiente capítulo tiene como objetivo informar al lector acerca de la problemática presente en el medio acerca de la operación de procesos industriales mediante máquinas cartesianas implementadas con motores AC de inducción. Además, pone en conocimiento del lector la idea de implementación de un prototipo de máquina de eje lineal escalable para máquinas cartesianas de gran consumo de potencia. El prototipo ejemplificará las ventajas de su operatividad en comparación con los mecanismos actuales.

1.1. Descripción del problema

Existen industrias, como en el campo de la construcción, que requieren sistemas de control de movimiento en máquinas cartesianas de gran escala para trabajos de impresión, manufactura, entre otros. Los sistemas de movimiento de alta dinámica más frecuentes en nuestro medio son robustos y poseen pérdidas de eficiencia energética. Además, el consumo de los motores antiguos provoca el aumento de armónicos en la red eléctrica, esto causa pérdidas de capacidad nominal en los cables alimentadores y transformadores, un mayor consumo energético y multas por parte de la empresa eléctrica.

Las empresas también desean que una misma máquina sirva para varias tareas, pero deciden contratar al servicio técnico especializado por la dificultad de programación de los controladores, produciéndose costos por la logística y el tiempo perdido con la producción parada.

1.2. Justificación del problema

El proyecto busca una alternativa para mostrar las ventajas de utilizar un sistema de control de movimiento (posición y velocidad) moderno y eficiente. Las máquinas cartesianas no solo requieren un movimiento rápido y preciso, sino también una mayor eficiencia energética.

El prototipo podrá ser utilizado como guía para construir máquinas con distintos fines. Los dos más importantes son la realización de tareas repetitivas o peligrosas para los trabajadores y la reducción de pérdidas en la industria de manufactura y construcción. Estas soluciones no existen en el mercado local, lo cual abre una gran oportunidad de activación económica en el área.

Puede resultar relativamente fácil configurar y programar todas las partes de un control de movimiento para una tarea específica, tener en la memoria del controlador varias recetas con trayectorias a seguir por la máquina, pero permitir al usuario crear sus propias trayectorias y cargarlas a la máquina es algo que solo se ha visto en brazos robóticos industriales actuales. He aquí la necesidad de tener una interfaz amigable para que el usuario pueda configurar fácilmente nuevas trayectorias a seguir, aumentando la versatilidad del producto final y reduciendo los costos de ingeniería a lo largo de la vida útil de la máquina.

1.3. Objetivos

1.3.1. Objetivo General

Diseñar un sistema de control de movimiento que sirva como prototipo para la construcción de máquinas cartesianas de gran escala y costo reducido mediante un eje lineal de alta precisión y versatilidad.

1.3.2. Objetivos Específicos

Determinar la topología de conexión entre los módulos a utilizar para la correcta sincronización de variables en el sistema mediante el uso de protocolos de comunicación industriales.

Estructurar el algoritmo del sistema para el correcto funcionamiento del prototipo mediante la programación en los distintos equipos utilizados.

Demostrar la precisión y versatilidad del prototipo para el cumplimiento de los parámetros establecidos por el cliente mediante la aplicación de diferentes pruebas de movimiento y tiempos de producción.

1.4. Marco teórico

En este marco teórico se explora el uso de un control de movimiento para máquinas cartesianas. En la primera parte se menciona la constitución de los sistemas de control de movimiento y los componentes y técnicas más utilizadas en implementaciones actuales. Posteriormente en la segunda parte se muestran ejemplos de aplicación, en algunos de ellos la máquina cartesiana resuelve el problema por sí sola, mientras que en otros casos se requiere el trabajo sincronizado de una máquina cartesiana y un robot acoplado a ella.

1.4.1. Control de Movimiento

Los robots cartesianos, también llamados *gantry robots*, son dispositivos electromecánicos que usan motores y actuadores lineales para posicionar una herramienta dentro de un área de trabajo delimitada [1]. Un control de movimiento para robots cartesianos permite establecer los puntos por los cuales se requiere que la herramienta pase, la velocidad en cada intervalo y las aceleraciones máximas permitidas.

Comparando un robot cartesiano con un brazo robótico de seis grados de libertad, el robot cartesiano constituye una alternativa más económica (alrededor de un 30% más barato). Al aumentar el tamaño del robot puede levantar más carga sin sufrir deformaciones a diferencia del brazo que incluso ve afectada su precisión al aumentar la carga. Si se lo utiliza en conjunto con tornillos sin fin permite una precisión igual al del brazo y su programación es mucho más sencilla por tener menos grados de libertad [1, 2].

Los sistemas de control de movimiento requieren que varios módulos trabajen en sincronía: controlador, procesador de movimiento y regulador de lazo cerrado. El papel del controlador lo asume un PLC, el procesador de control de movimiento está integrado en el mismo CPU del PLC, mientras que el regulador de lazo cerrado corresponde al *Drive* de cada motor. Se debe asegurar que la información entre los módulos viaje rápidamente y sin interferencias, por lo que se utilizan protocolos de comunicación industrial de alta velocidad como Ethernet IP [3, 4, 5].

Los robots cartesianos a menudo ejecutan servo aplicaciones que usualmente son impulsadas por motores síncronos de imanes permanentes (PMSM) o motores DC sin escobillas, ambos se caracterizan por su alto rendimiento que se evidencia en fácil rotación de su rango de velocidad, full torque en velocidad cero y rápida aceleración y desaceleración [8].

Analizando el caso del PMSM, en el estator no circula corriente de magnetización para producir flujo constante, caso contrario al motor de inducción (IM) por lo tanto el PMSM opera a un factor de potencia más alto que el IM. Además, el PMSM al no poseer excitación DC en el rotor (que es reemplazada por su imán permanente) se exonera de pérdidas rotóricas y mantenimiento regular en las escobillas, lo que lo hace más eficiente que la máquina síncrona (SM). En conclusión, el PMSM tiene una relación mayor entre torque-inercia y densidad de energía, menor masa y momento de inercia en comparación a la IM o la SM con rotor devanado [6].

El control del PMSM es caracterizado por una rotación suave, control de torque completo en velocidad suave, y rápida aceleración y desaceleración. Para lograrlo, se utilizan técnicas de control vectorial referidas también como control de campo orientado (FOC) [8]. Este tipo de control requiere un *encoder* para la retroalimentación de posición, normalmente se lo conoce como servocontrol y el algoritmo ya se encuentra integrado en la mayoría de *Drives* actuales.

Mientras se van explorando los beneficios del uso de robots cartesianos en la industria, los fabricantes han desarrollado sus equipos para que su implementación sea más sencilla, y se puedan reemplazar fácilmente sus partes en caso de daños o expansión. Antes se pensaba que un PLC solo serviría para reemplazar los controles electromecánicos como relés y temporizadores [9]. Pero actualmente los PLCs tienen más capacidades de cómputo, varios lenguajes de programación, y los algoritmos de control de movimiento ya integrados. Los bloques de control de movimiento de cada

fabricante están estandarizados y si quiero agregar más motores al sistema bastaría con cambiar de un *Drive* mono eje a uno multieje, haciendo que la configuración se deba realizar en un solo *Drive* en lugar de en tres diferentes para los tres ejes [10]. Todas estas facilidades se dan actualmente para reducir costos y tiempos de programación e instalación.

1.4.2. Aplicación del control de movimiento en máquinas cartesianas

Existe una técnica de creación de productos llamada manufactura aditiva. Esta técnica se utiliza por ejemplo en la impresión 3D con plástico, pero también se puede utilizar en concreto, metal, material sintético e incluso biológico [11]. Al utilizarla, se reducen los costos de fabricación por la no utilización de moldes y al no producir materiales de desecho.

La manufactura aditiva también ha llegado a la industria de la construcción, haciendo posible la impresión 3D de casas completas. Los robots más utilizados son los cartesianos, por su sencilla programación y bajo costo, pero se dificulta su ensamblaje y transporte por ser de mayor tamaño. Mientras que los brazos robóticos de seis ejes dan más libertades y flexibilidad al trabajo, pero aún no han sido completamente adoptados para esta aplicación. Al utilizar manufactura aditiva en la construcción se reduce entre 35% y 60% del valor de la estructura por no utilizar madera y cana para sostener el cemento hasta que éste se seque [12].

Otra aplicación importante de los robots cartesianos está en reemplazar trabajos peligrosos para los trabajadores humanos, ya que se pueden utilizar para trabajos de pintura en los tumbados de edificaciones, evitando que el trabajador desarrolle lesiones cervicales o problemas oculares y respiratorios por la exposición a las partículas de pintura [7]. También se los ha utilizado para posicionar un robot limpiador de ventanas a través de toda la superficie externa de un edificio, aumentando la velocidad y reduciendo el número de accidentes [3].

Por último, existen aplicaciones donde se necesitan tanto las habilidades de un robot cartesiano como de un brazo robótico. Este es el caso de una empresa alemana que se encarga de la soldadura de estructuras para la construcción de barcos, en donde se necesita la flexibilidad y maniobrabilidad de los brazos robóticos para soldar en cualquier punto del espacio y cambiar de herramienta rápidamente, pero las estructuras a soldar son extremadamente grandes, por lo que se monta el brazo robótico en la plataforma de un robot cartesiano para poder moverlo a través de grandes distancias [13].

Se espera que el mercado del control de movimiento crezca alrededor de 5.5% hasta el 2022. Este crecimiento incluye desde el software del controlador hasta los motores [14]. En el Ecuador, casi el 14% de las empresas, correspondiente al 26% del ingreso de capital del país [15], pueden beneficiarse de estos sistemas para aumentar su rentabilidad.

CAPÍTULO 2

2. METODOLOGÍA

En este capítulo se abordará la propuesta para la realización del proyecto. Se indicarán las partes constitutivas tanto de hardware como de software y el porqué de esas elecciones.

2.1. Concepción del sistema

Un sistema de control de movimiento está formado principalmente de un actuador, una retroalimentación, y un controlador. El actuador corresponde al motor eléctrico y a su accionamiento. La retroalimentación se obtiene por medio de un sensor de posición llamado *encoder*. Por último, el controlador es el encargado de indicar la cantidad de energía que se envía al actuador para que su posición y velocidad sean iguales a las indicadas por el usuario. En este caso, el controlador también incluye la interfaz de usuario y el manejo de los sensores y actuadores adicionales. En la figura 2.1 se observa un esquema general del sistema.

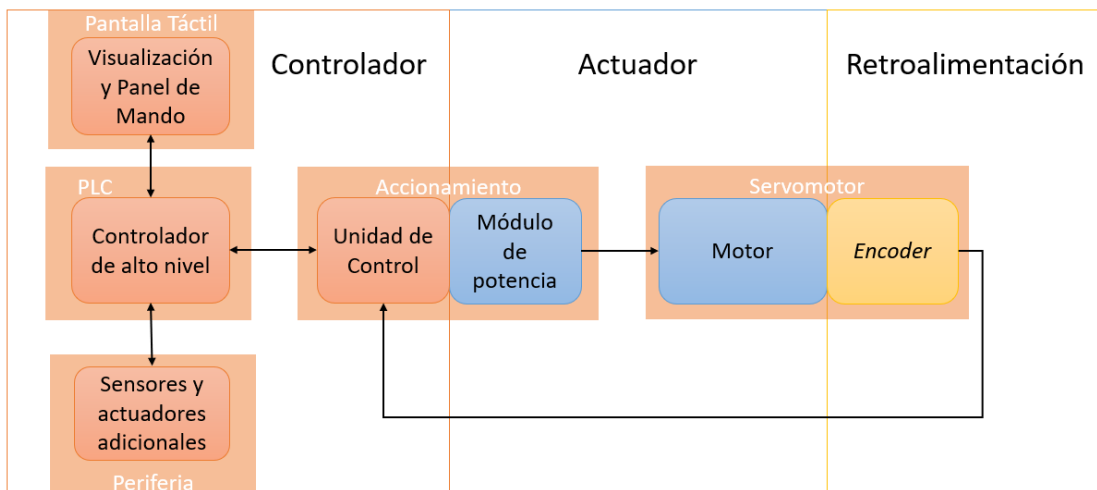


Figura 2.1 Esquema general del sistema de control de movimiento

2.1.1. Servomotor

Se decidió utilizar un servomotor síncrono de ½ HP de serie SIMOTICS 1FK7033-4CF21-1UA0 mostrado en la figura 2.2. Este motor es utilizado para aplicaciones donde se requiere alta dinámica en el movimiento, permitiendo realizar frenados, aceleraciones y cambios de dirección muy rápidamente y con toda la fuerza posible. Además, posee un *encoder* incremental de tipo *resolver* acoplado internamente. Para acceder al valor del *encoder*, se utiliza un conector con tecnología *DRIVE-CLiQ*. Esta tecnología permite leer los datos del motor y del *encoder* directamente desde un accionamiento que posea la misma tecnología sin necesidad de gastar tiempo ingresando cada dato independientemente [16].

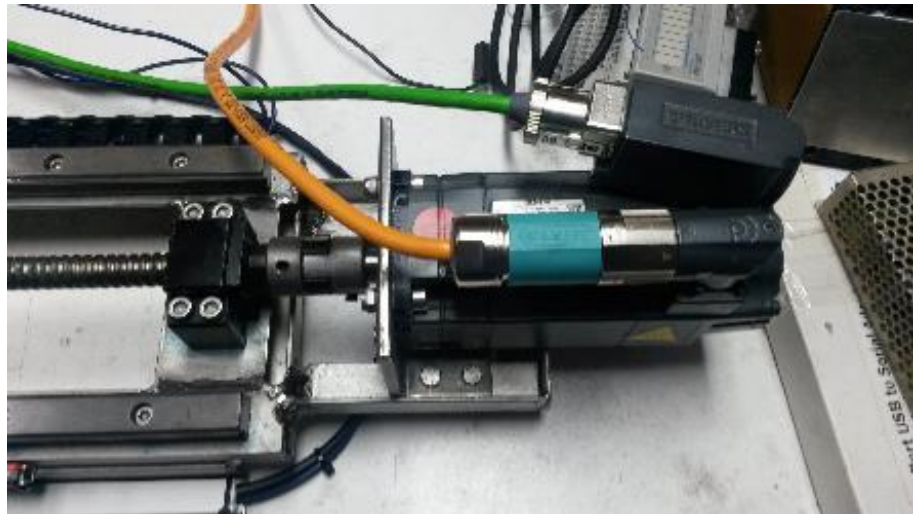


Figura 2.2 Motor con conectores y acople al tornillo

La característica de precisión del sistema es otorgada por medio del acople mecánico entre el motor y el eje lineal. El movimiento rotacional del eje del motor se transforma en movimiento lineal a través de un tornillo sin fin mostrado en la figura 2.3 acoplado a un sistema de rieles, el paso del tornillo determina la distancia horizontal que se desplaza la herramienta por cada revolución del eje del motor, en este caso se tiene un paso de 5 mm/vuelta.

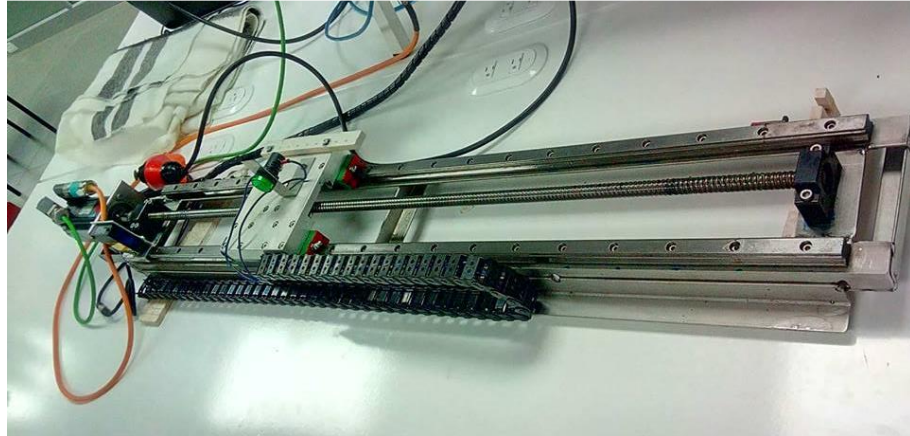


Figura 2.3 Tornillo sin fin

2.1.2. Selección del accionamiento adecuado

Se contaba con dos accionamientos del mismo fabricante, los dos con el mismo módulo de potencia. Este módulo puede entregar una corriente de 4 A y una potencia de 0.75 kW que superan los 2 A y 0.375 kW del motor por lo que energéticamente hablando cualquiera de los dos podía utilizarse. Estos accionamientos se pueden observar en la figura 2.4.

La diferencia entre ellos radica en la tecnología y la capacidad de su unidad de control. Se decidió utilizar el accionamiento de la serie SINAMICS S120 con CPU 310-2-PN en lugar del accionamiento de la serie SINAMICS G120 por los siguientes motivos:

- Posee un puerto específico para la tecnología *DRIVE-CLiQ* del motor.
- Permite la función de servocontrol.
- Permite la comunicación con un PLC mediante el protocolo PROFIdrive I/O IRT (*Isochronous Real Time*).

Las características detalladas de este accionamiento pueden ser corroboradas en [17] y [18].



Figura 2.4 Opciones para el accionamiento del motor

2.1.3. Selección del controlador de alto nivel

El control de movimiento de lazo cerrado es efectuado por la unidad de control del accionamiento, pero se necesita un dispositivo maestro al cual llamaremos controlador de alto nivel para que sirva de maestro y maneje tanto al accionamiento como a la pantalla táctil que servirá de interfaz de usuario. Este controlador también sirve como memoria no volátil para distintas funciones que se indicarán en secciones posteriores. Se contaba con dos tipos de PLC para esta función, un SIMATIC S7-1200 y un SIMATIC S7-1500. Se decidió trabajar con el SIMATIC S7-1500 con CPU 1516-3PN/DP debido a que permite la comunicación con PROFIdrive IRT. Además, el S7-1500 es aproximadamente 4 veces más rápido que el S7-1200 [19]. En la figura 2.5 se puede observar el PLC utilizado.

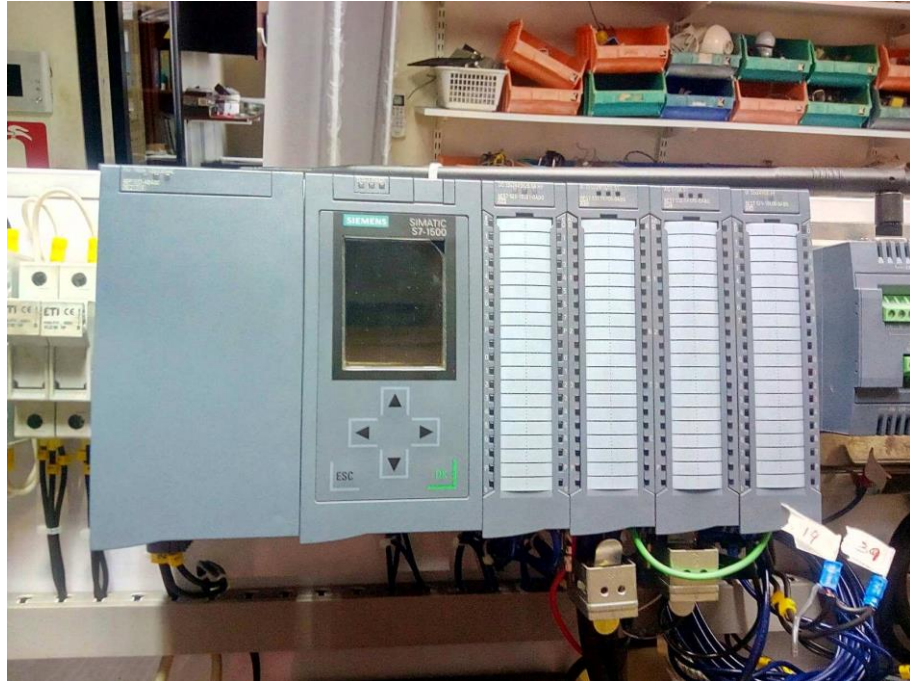


Figura 2.5 Controlador de alto nivel utilizado

2.2. Objeto tecnológico: Eje de posicionamiento

Se necesita que el controlador de alto nivel dedique una porción significativa de su memoria: comunicación con el accionamiento, interpolación de la trayectoria a seguir, al manejo, diagnóstico de alarmas y palabras de estado del accionamiento y motor. Para realizar este trabajo, en la programación del controlador se agrega un objeto tecnológico, específicamente un eje de posicionamiento. Con este objeto tecnológico se tiene un asistente para configurar el eje lineal físico que se posee.

En la figura 2.6 se muestra los tipos de objetos tecnológicos que se pueden agregar al controlador. Se utilizó un eje de posicionamiento porque el control principal es la posición del eje y la velocidad se controla mientras el eje está cambiando de posición. Toda la programación del controlador se realizó en TIA Portal por mandato del fabricante. Toda la información referente a la función del objeto tecnológico fue obtenida de [20].

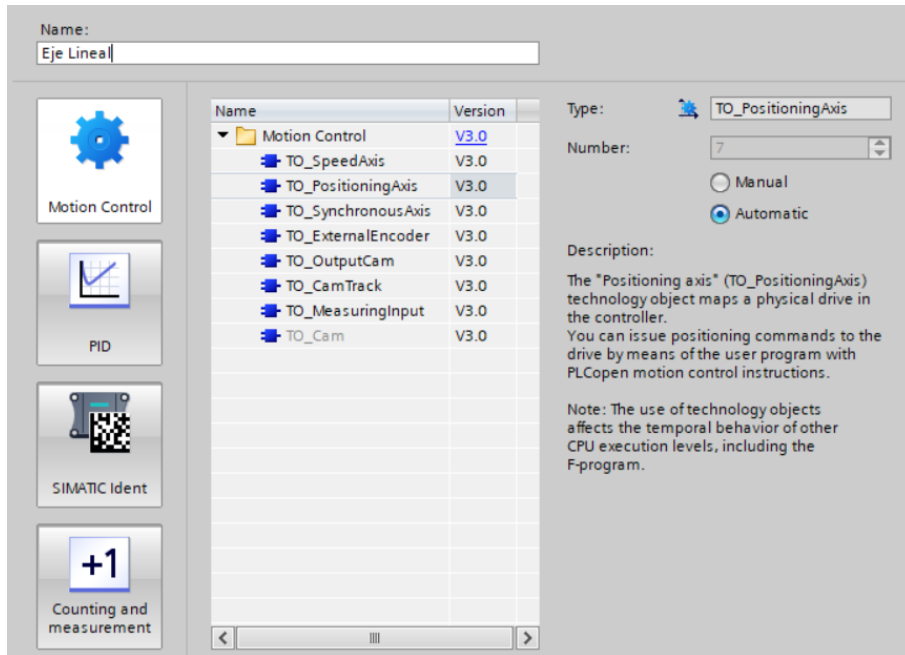


Figura 2.6 Catálogo de objetos tecnológicos disponibles

2.2.1. Configuración de hardware

Dentro del asistente de configuración del objeto tecnológico se ingresaron los datos del eje físico que se está controlando.

Tabla 2.1 Configuración de Hardware

Dato por ingresar	Configuración
Tipo de eje a utilizar	Lineal
Unidades de posición y velocidad	mm y mm/s
Tipo de comunicación con el accionamiento	PROFIdrive
Intercambio de datos con el accionamiento	Telegrama Básico 5

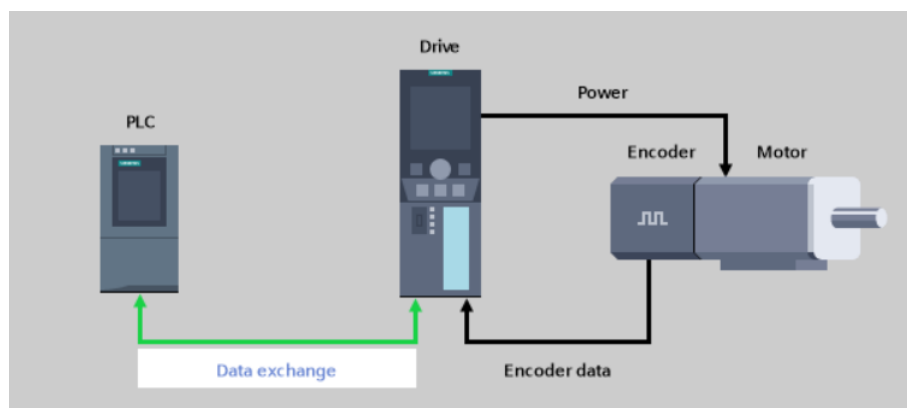


Figura 2.7 Diagrama de comunicación entre el controlador y el accionamiento

El intercambio de datos se realiza por medio de un telegrama básico, específicamente el 5, donde se envían y reciben 9 palabras de datos entre el controlador y el accionamiento. La elección de este telegrama se debe a que nos permite utilizar el Servocontrol Dinámico. Un diagrama explicativo del intercambio de datos se puede observar la figura 2.7.

Se utilizó el software STARTER para acceder a la configuración del accionamiento y establecer la comunicación por telegrama 5. Además, se añadió una extensión de telegrama de 8 palabras para transmitir los valores de temperatura, corriente y par del motor al controlador. Las configuraciones realizadas tanto en el controlador como en el accionamiento pueden observarse en la figura 2.8 mientras que la asignación de las palabras a las variables del controlador y accionamiento son mostradas en la figura 2.9.

Configuración en el controlador

Module	Rack	Slot	I address	Q address	Type
▼ SINAMICS-S120-CU310-2PN	0	0			SINAMICS S120 CU310-2 PN V4.8
▶ nodoejelineal	0	0 X150			SINAMICS-S120-CU310-2PN
▼ DO SERVO_1	0	1			DO SERVO
Module Access Point	0	1 1			Module Access Point
	0	1 2			
Standard telegram 5, PZD-9/9	0	1 3	20...37	12...29	Standard telegram 5, PZD-9/9
Supplementary data, PZD-8/8	0	1 4	38...53	30...45	Supplementary data, PZD-8/8
	0	?			

Configuración en el accionamiento

Vista maestra:					
Objeto	Objeto de accionamiento	N.º	Tipo de telegrama	Datos de entrada	Datos de salida
				Longitud	Longitud
1	SERVO_02	2	Telegrama estándar 5, PZD-9/9	9	9
			Prolongación de telegramas	8	8
2	Control_Unit	1	Configuración libre de telegramas con BICO	0	0
DO que no están asignadas a ningún slot (sin intercambio cíclico de datos)					

Figura 2.8 Configuración del telegrama

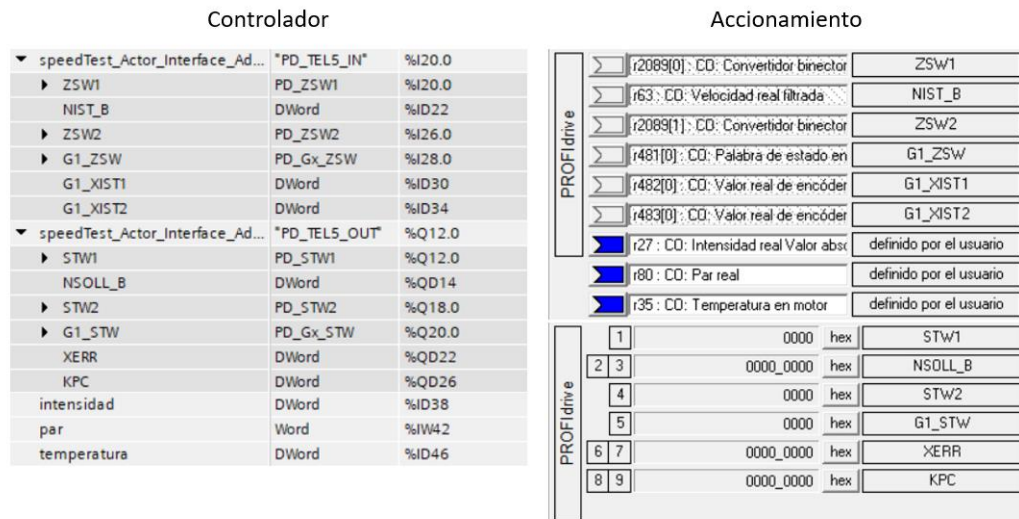


Figura 2.9 Asignación de las palabras del telegrama

Se estableció la velocidad máxima del eje en 250 mm/s que corresponden a 3000 rpm del motor. Por ser un motor síncrono, se puede acelerar a más de 3000 rpm que es su velocidad nominal, pero entonces se tiene una pérdida de la capacidad de torque máximo del motor, una mayor vibración del tornillo, y la reducción de la vida útil.

2.2.2. Límites de Posición

Existen dos tipos de límites para el eje: por software y por hardware. Los límites de software funcionan como alarmas cuando el eje posee una referencia de posición absoluta, y detienen el accionamiento cuando el eje haya alcanzado esos límites. Sin embargo, si existiera un error en la referenciación y el eje se moviera fuera de los límites de software sin saberlo, activaría los límites de hardware deteniendo igualmente el accionamiento. Los límites por software son valores de posición mientras que los límites de hardware son entradas digitales físicas del controlador. El diagrama de la figura 2.10 muestra el lugar y los valores de los límites establecidos.

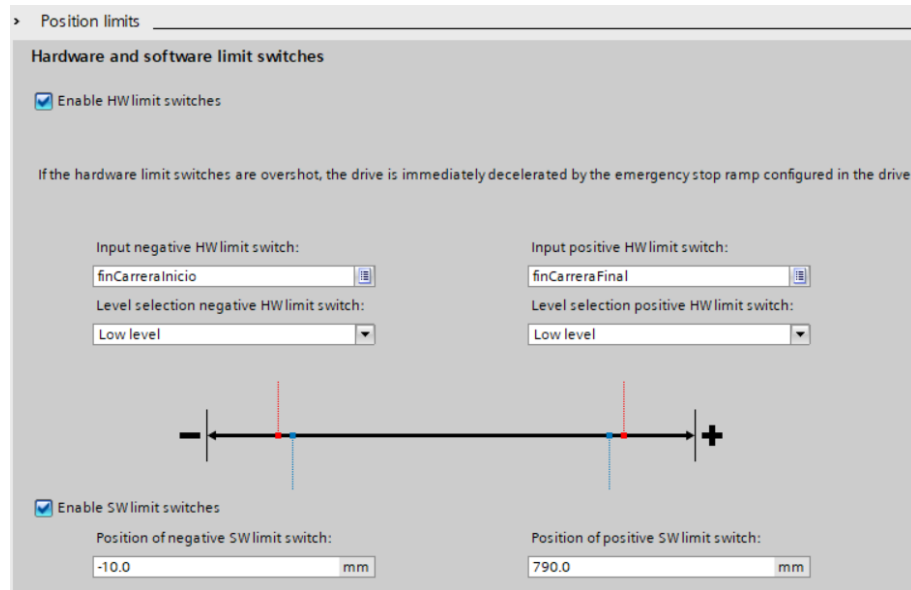


Figura 2.10 Configuración de los límites de posición

2.2.3. Referenciar el eje

Para que el sistema pueda conocer en qué posición absoluta se encuentra el eje, se debe primero tener una referencia. Esta referencia se puede colocar en cualquier parte del camino del eje, en este proyecto se lo colocó casi al inicio del tornillo y representa la posición 0 mm. El objeto tecnológico del controlador posee una instrucción donde ejecuta una rutina de búsqueda del sensor de referencia, este sensor debe ser una entrada digital del controlador. Como el punto de referencia está al inicio del tornillo, se decidió optar por la rutina mostrada en la figura 2.11 donde primero se retrocede hacia la parte inicial del tornillo para posteriormente regresar y ubicarse justo en el cambio de la señal digital.

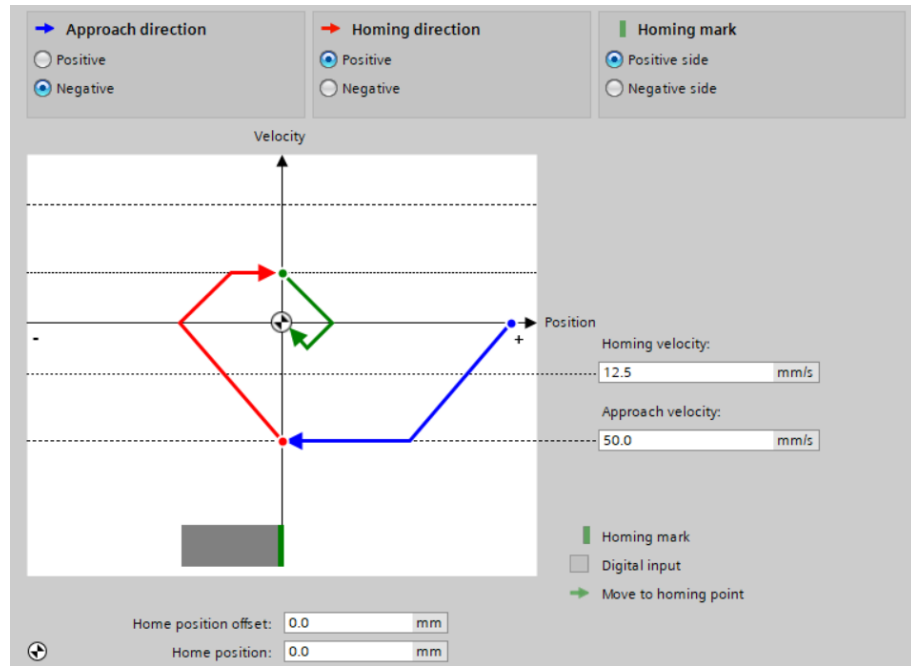


Figura 2.11 Rutina para referenciar el eje

2.2.4. Lazo de control

El controlador solo envía los valores de referencia de posición y velocidad mientras que el accionamiento se encarga del verdadero control de lazo cerrado. Esto ocurre por las velocidades de comunicación, lo más eficaz para un sistema rápido como un control de movimiento es que se distribuya la carga entre el controlador y el accionamiento. En este proyecto se utiliza una opción llamada Servo control dinámico (DSC por sus siglas en inglés).

En este modo, el controlador solamente realiza la operación del cálculo de error de la posición y lo envía junto con la ganancia y el valor de referencia de la velocidad para que el accionamiento actúe. Entonces el control del sistema es repartido exitosamente: El controlador se encarga de convertir los valores del *encoder* en desplazamiento lineal para compararlo con la posición de destino, crear una serie de puntos por interpolación lineal y enviarlos a intervalos de tiempo establecidos por medio del telegrama 5 en comunicación PROFIdrive IRT; mientras que el accionamiento recibe el error de la posición y la consigna de referencia de velocidad y procede a realizar el servocontrol para que ese error se vuelva cero y la velocidad real sea igual a la consigna

de referencia, cerrando así el lazo de control y permitiendo el movimiento preciso del eje. La figura 2.12 muestra el diagrama del lazo de control.

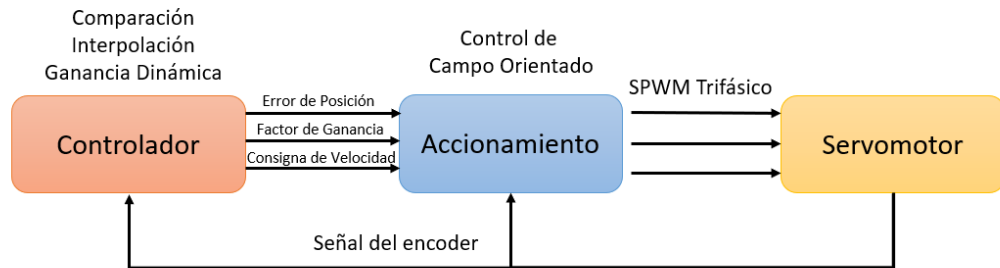


Figura 2.12 Lazo de control del sistema

2.3. Periferia

En esta sección se menciona la forma en que se seleccionaron los sensores y actuadores adicionales.

2.3.1. Sensores para los límites de posición

Como seguridad para evitar que el eje golpee los extremos de la estructura física, se colocó y configuró los límites de posición por hardware. Una de las alternativas fue la utilización de sensores inductivos de proximidad que detecten la presencia de un objeto de características metálicas, pero la dificultad de su instalación mecánica y costo monetario declinaron dicha alternativa. Por otro lado, el interruptor tipo final de carrera es de fácil montaje y su funcionamiento mecánico son ideales para esta aplicación. Los interruptores de final de carrera utilizados se pueden ver montados en la figura 2.13.



Figura 2.13 Interrupidores de final de carrera

Estos interruptores se representan eléctricamente mediante un contacto normalmente cerrado (NC) que se conectan en el módulo de entradas digitales del PLC. Se eligió el contacto normalmente cerrado por seguridad, si el cable se rompe, el controlador interpretará que el sensor fue activado y detendrá el accionamiento. Se colocaron dos interruptores, uno al inicio y otro al final de la estructura como se muestra en la figura 2.10.

2.3.2. Sensor para la referencia

La posición de referencia se definió como *home* y se pensó en utilizar también un interruptor de final de carrera. Realizando algunas pruebas se llegó a la conclusión que el fin de carrera, al ser accionado mecánicamente, producía pequeños deslizamientos por la flexión de la palanca metálica. Entonces, se utilizó un sensor laser de distancia LM10 [21]. El sensor utilizado es el modelo ANR1251 y se conecta al PLC por medio de un controlador ANR5141. Se utilizó la función de comparador para que, cuando la plataforma del eje esté a 50 mm o menos del sensor, se active una salida digital conectada al PLC, que corresponde a la señal de referencia de *home*. Con el sensor laser instalado se evita la descalibración del sistema por referencia. En la figura 2.14 se puede observar el sensor instalado apuntando hacia la plataforma.



Figura 2.14 Sensor láser instalado

2.3.3. Elemento terminal

Para probar el funcionamiento del sistema, es indispensable el montaje de un elemento terminal o herramienta que caracterice la función de la máquina. En nuestro medio, los procesos varían desde soldadura, pintura, corte, transporte o sellado, por lo que para efecto de prototipo se decidió usar una luz piloto que simula la acción de un instrumento terminal cualquiera.

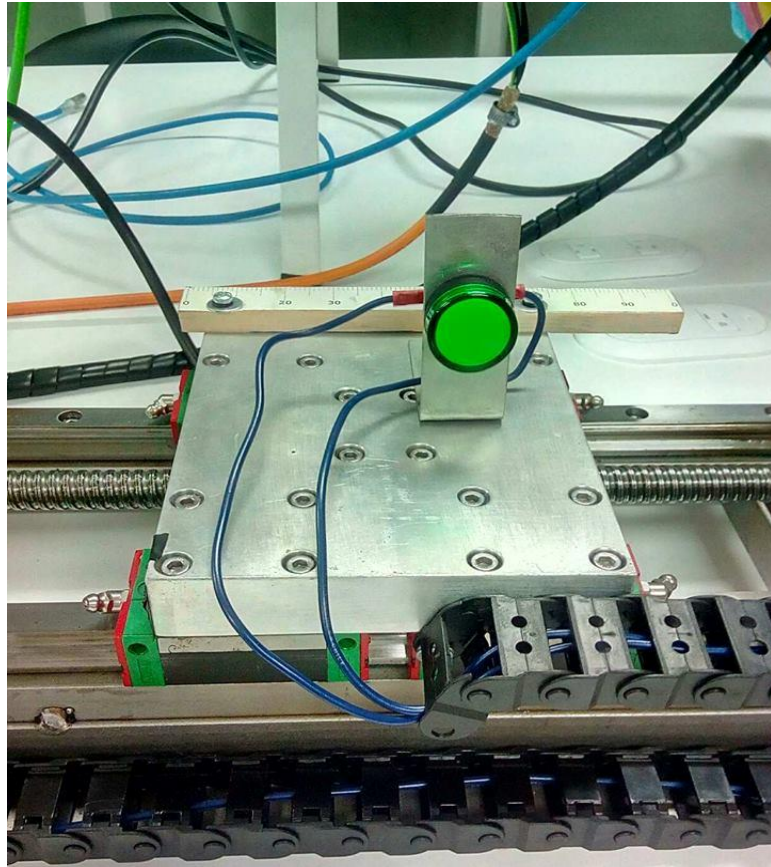


Figura 2.15 Luz piloto como elemento terminal

La luz piloto instalada se puede observar en la figura 2.15 y está conectada al módulo de salidas digitales del PLC.

2.3.4. Manejo manual del sistema

Inicialmente se tenía pensado construir una botonera para realizar la marcha, paro, *jog* y cambio de modo del sistema. Pero el usuario final tendría que presionar los botones para cambiar de modo y acercarse a la pantalla táctil para ingresar por teclado la posición y velocidad, lo cual se comprobó que era incómodo. Entonces se decidió que todo botón o selector estaría integrado en la pantalla táctil y que no habría botonera física. Sin embargo, se diseñó un mando utilizando un *joystick* para mover el eje a una velocidad proporcional a la desviación de este. El prototipo del mando se puede observar en la figura 2.16.



Figura 2.16 Mando para control manual

Al movimiento del *joystick* se le agregó una sensibilidad en el controlador de cinco niveles. Cada nivel aumentaba la velocidad manual máxima en 30 mm/s. Esta escala de sensibilidad no era muy eficiente porque a mínima sensibilidad el eje se movía muy rápido. Entonces se optó por colocar valores discretos a los niveles de sensibilidad haciendo que en los primeros la velocidad sea muy baja, estos valores se pueden observar en la tabla 2.2.

Tabla 2.2 Escala de sensibilidad para mando manual

Valor de Sensibilidad	Rango de velocidad (mm/s)
1	[0-5]
2	[0-20]
3	[0-50]
4	[0-100]
5	[0-150]

El mando también posee un botón que debe ser presionado en todo momento mientras se hace el movimiento manual, dando seguridad de que el eje no se activará si el *joystick* se mueve por accidente.

2.3.5. Diagrama eléctrico de la periferia

La figura 2.17 muestra el diagrama de conexiones entre los sensores y actuadores adicionales y el controlador. También se muestran la asignación de entradas y salidas del PLC en la tabla 2.3.

Tabla 2.3 Asignación de entradas y salidas del PLC

Descripción	Tipo de Dato	Memoria Ocupada	Dirección asignada
Sensor de posición inicial	Bool	1 bit	%I17.1
Sensor de posición final	Bool	1 bit	%I17.0
Sensor láser de desplazamiento (salida de comparador)	Bool	1 bit	%I17.4
Sensor láser de desplazamiento (salida de corriente)	Word	16 bits	%IW6
Pulsador de seguridad joystick	Bool	1 bit	%I17.3
Desplazamiento de joystick	Word	16 bits	%IW0
Luz piloto de elemento terminal	Bool	1 bit	%Q0.0

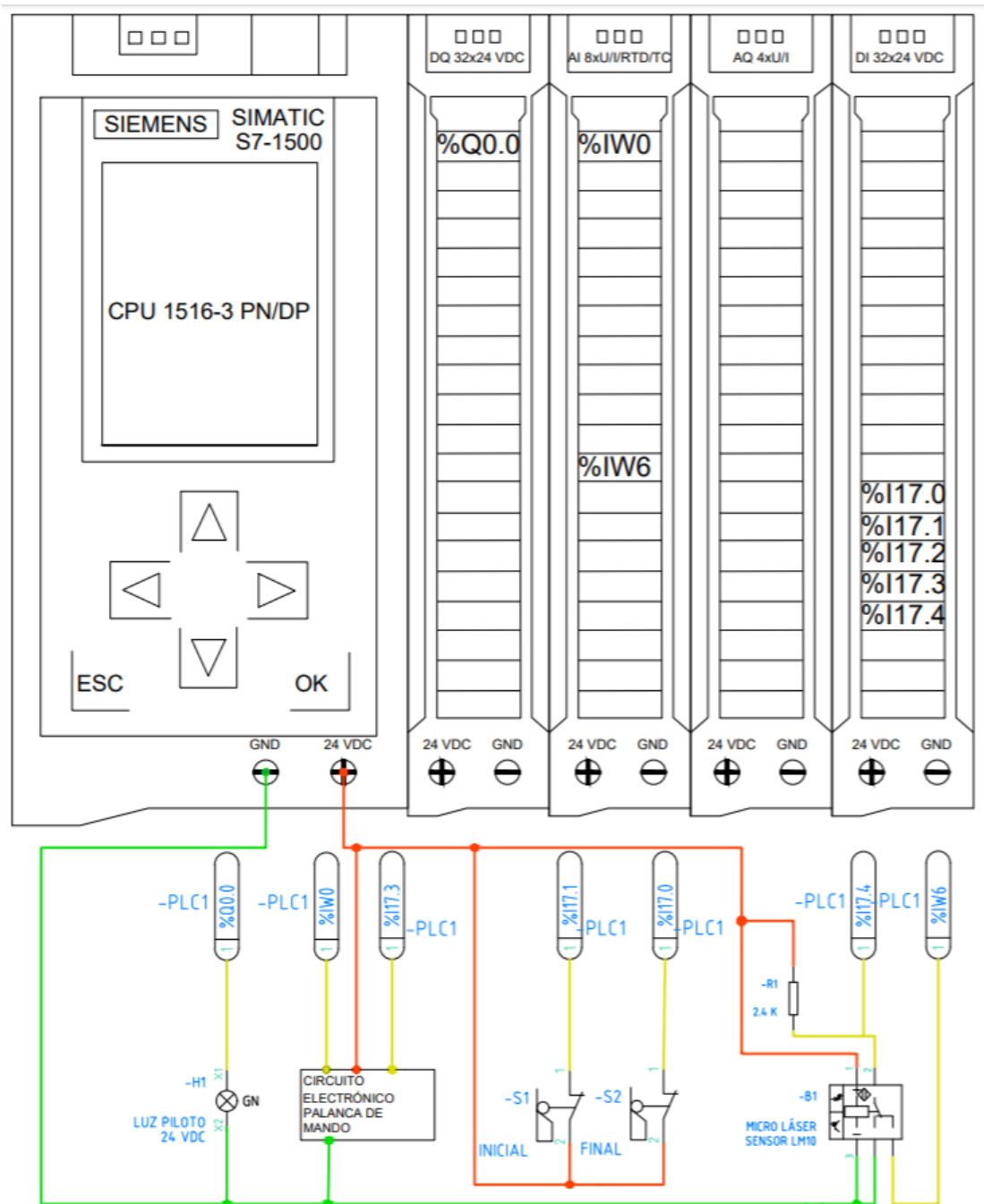


Figura 2.17 Diagrama eléctrico de conexiones del PLC y la periferia

2.4. Topología de comunicación

Como se utiliza la capa física Ethernet TCP/IP para las comunicaciones, cada equipo debe poseer una dirección IP única. La tabla 2.4 muestra la asignación de direcciones IP de cada equipo.

Mediante el protocolo Ethernet se realizó la conexión entre el computador y el PLC enlazándose en la subred llamada PN/IE 2, esta conexión permite la carga de los programas hacia el controlador y la pantalla, el diagnóstico y el acceso al servidor web. La conexión entre el PLC y la pantalla táctil se realizó por medio de la subred denominada PN/IE 1 utilizando el protocolo Profinet, mientras que el accionamiento, aunque se encuentra también en la subred PN/IE 1, utiliza el protocolo PROFIDrive. En la figura 2.18 se muestra un bosquejo de la topología de comunicación.

Tabla 2.4 Asignación de direcciones IP

Componente	Dirección IP
Computador	192.168.1.10
PLC subred PN/IE 1	192.168.0.1
PLC subred PN/IE 2	192.168.1.1
Accionamiento de motor	192.168.0.2
Pantalla táctil	192.168.0.3

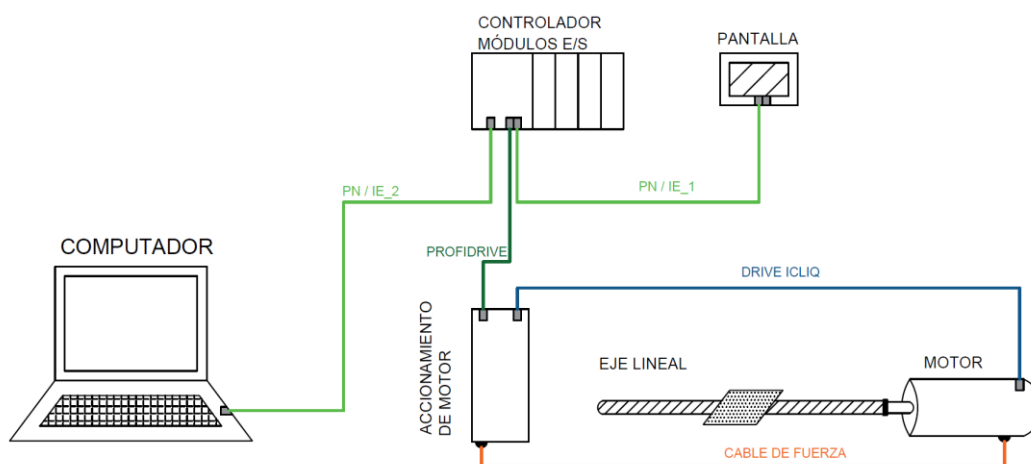


Figura 2.18 Topología de comunicación

2.5. Programación del sistema

Para la programación del PLC se utilizaron tanto el lenguaje de contactos (KOP) como el lenguaje estructurado (SCL). El uso de cada uno dependía de la complejidad de las funciones que se estaban codificando, incluso en algunas funciones se utilizan ambos lenguajes en conjunto. Esta programación no es secuencial y consta de varias partes trabajando en paralelo, una independiente de la otra.

La programación permite controlar el eje de forma manual continua (por medio del mando), manual discreto (colocando valores exactos de posición y velocidad), automática (ejecutando una secuencia previamente creada), visualizar algunos parámetros del motor, crear secuencias de movimiento, y guardar y cargar secuencias en la memoria SD del PLC. En la figura 2.19 se muestra un diagrama de bloques general del programa.

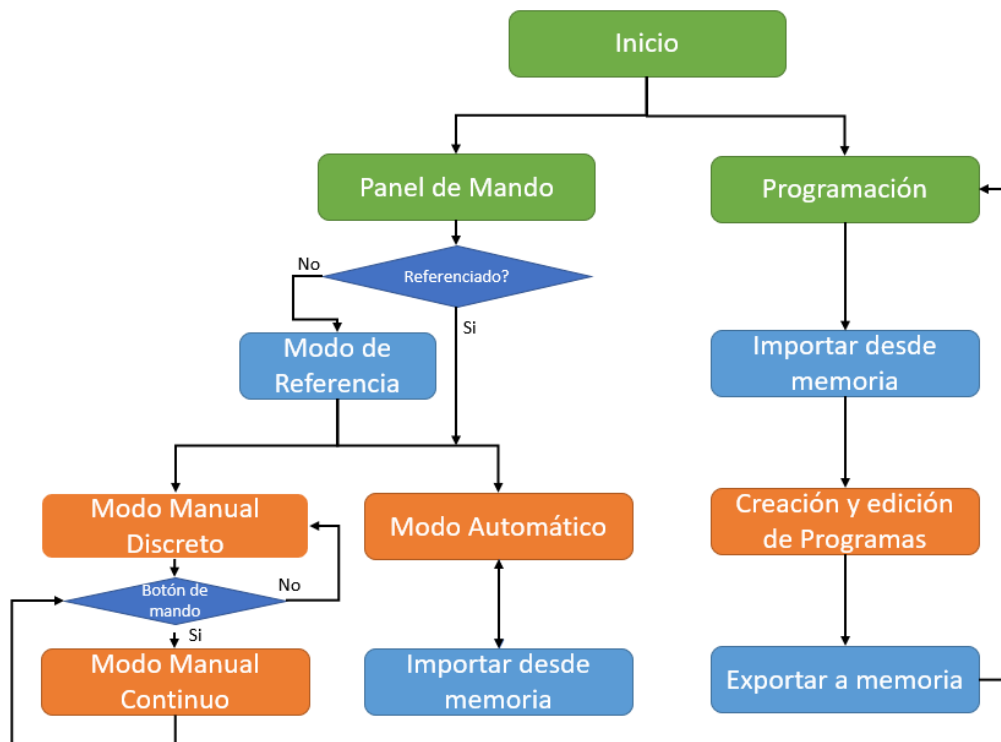


Figura 2.19 Algoritmo general del programa

2.5.1. Instrucciones para el control de movimiento

TIA Portal ofrece una serie de instrucciones para el control de movimiento que se pueden observar en la figura 2.20. El detalle de estas instrucciones se puede revisar en [5].

▼	Folder	Motion Control		V3.0
■	MC_Power	Enable, disable technol..	V3.0	
■	MC_Reset	Acknowledge alarms, r...	V3.0	
■	MC_Home	Home technology obje...	V3.0	
■	MC_Halt	Pause axis	V3.0	
■	MC_MoveAbsolute	Position axis absolutely	V3.0	
■	MC_MoveRelative	Position axis relatively	V3.0	
■	MC_MoveVelocity	Move axis with velocity...	V3.0	
■	MC_MoveJog	Move axis in jog mode	V3.0	
■	MC_MoveSuperimpo...	Positioning axes overla...	V3.0	
■	MC_SetSensor	Switch alternative enco..	V3.0	
■	MC_TorqueLimiting	Activate and deactivate..	V3.0	

Figura 2.20 Catálogo de instrucciones para el control de movimiento

2.5.2. Modo de referencia

Para que el eje pueda moverse a una posición absoluta dentro de su rango de movimiento, primero debe estar referenciado. Cada vez que el usuario quiere mover el eje y este no esté referenciado, primero se le pide que realice la rutina de *home* con la instrucción MC_Home.

2.5.3. Modo manual discreto

En este modo de funcionamiento, el usuario ingresa la posición absoluta de destino y la velocidad. Luego, con la instrucción MC_MoveAbsolute el eje se mueve hasta esa posición.

2.5.4. Modo manual continuo

En este modo, el usuario utiliza el mando para mover el eje a la velocidad indicada dependiendo de la sensibilidad escogida y a la fuerza aplicada en el joystick. Se utiliza la instrucción MC_MoveVelocity porque no se realiza un control de posición.

2.5.5. Modo automático

Para esta parte del programa los valores de posición y velocidad deben estar guardados previamente en arreglos para ser leídos mientras se ejecuta el programa. Se definió entonces un tipo de variable llamado *Step_info*, cada

variable de este tipo tiene información sobre un paso de la secuencia automática: El tipo de movimiento o función, la posición de destino, la velocidad de movimiento, y si se debe o no encender el elemento terminal.

Cada secuencia está guardada en un arreglo de 256 variables de tipo *Step_info*, estas secuencias son conocidas por los usuarios como programas. La figura 2.21 muestra un ejemplo de las variables de tipo *Step_Info*.

▼ Static		
■ ▼ Steps	Array[0..255] of *Step_Info*	
■ ▼ Steps[0]	*Step_Info*	
■ Function	String	''
■ Position	LReal	0.0
■ Speed	LReal	0.0
■ Tool	Bool	false
■ ▼ Steps[1]	*Step_Info*	
■ Function	String	''
■ Position	LReal	0.0
■ Speed	LReal	0.0
■ Tool	Bool	false
■ ▶ Steps[2]	*Step_Info*	
■ ▶ Steps[3]	*Step_Info*	
■ ▶ Steps[4]	*Step_Info*	

Figura 2.21 Variables de tipo Step_info

Una vez que se tiene esta lista, se depura el programa y se obtiene información de él mediante un compilador. Este compilador interpreta la variable *Function*. Si esta variable tiene el valor *Move*, el eje se mueve hacia la posición indicada como en el modo manual discreto. Pero si el valor es *Delay*, el eje detiene su movimiento una cantidad de tiempo igual al valor que tenga la variable *Speed* en ese paso, ignorando la variable *Position*.

Luego de la compilación, se revisa constantemente la salida *Done* del bloque MC_MoveAbsolute para saber cuándo el eje llegó a la posición establecida y tenemos que cambiar a la siguiente posición del programa. La figura 2.22 muestra el algoritmo del modo automático.

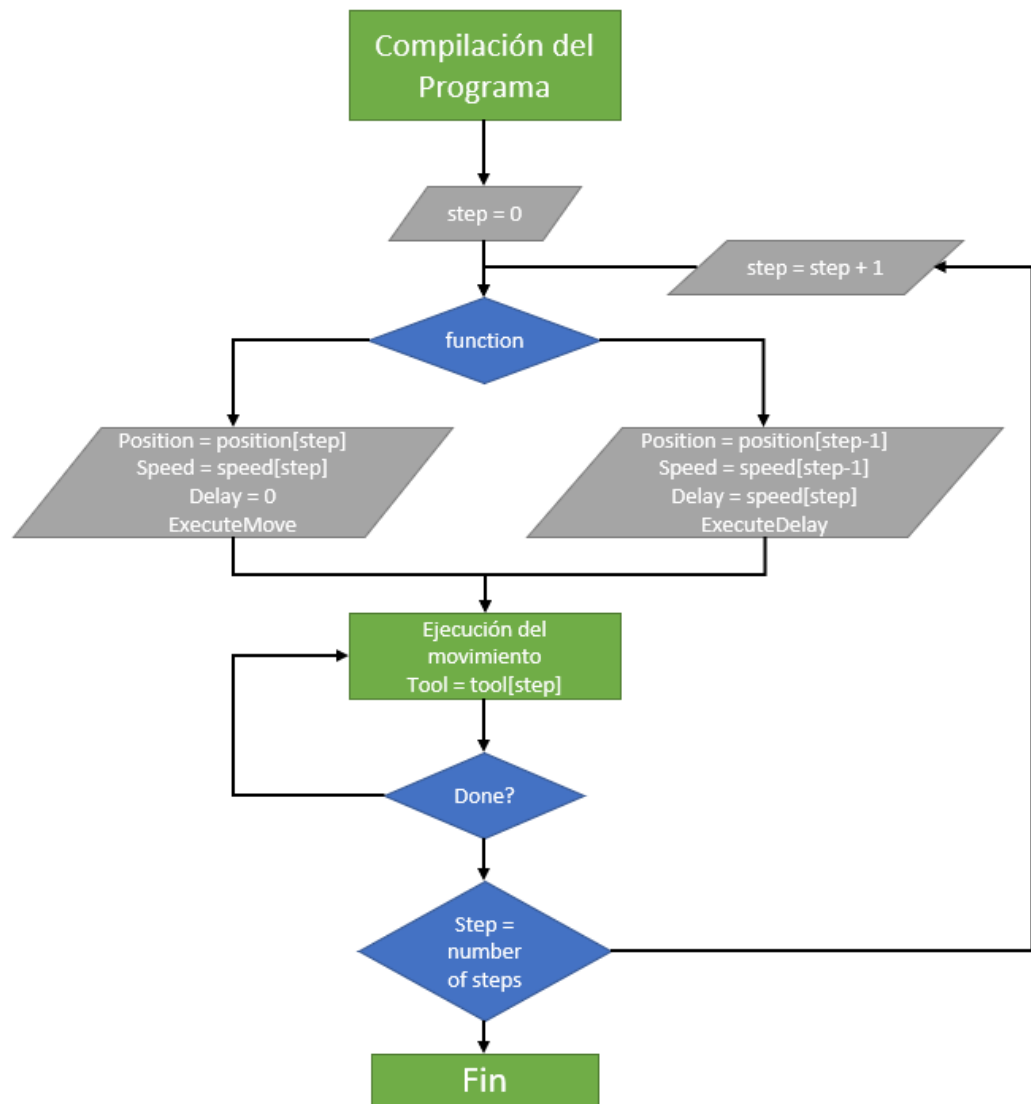


Figura 2.22 Algoritmo del modo automático

2.5.6. Importar y exportar programas

Desde el inicio se conocía que debía de existir un espacio de memoria no volátil para guardar los programas que el usuario fuera creando, la primera opción fue guardar los programas en una memoria de la pantalla táctil y utilizarla como recetas. Pero no se tenía libertad para establecer la estructura de los programas guardados y eran muchas las variables que se necesitaban enviar al PLC por medio de Profinet. Como resultado se decidió utilizar los bloques de *ImportRecipe* y *ExportRecipe* del PLC, con estos bloques, se guarda y lee los arreglos en la tarjeta SD del controlador haciendo que se

pueda elegir la estructura de guardado y que la velocidad de lectura y escritura sea mayor.

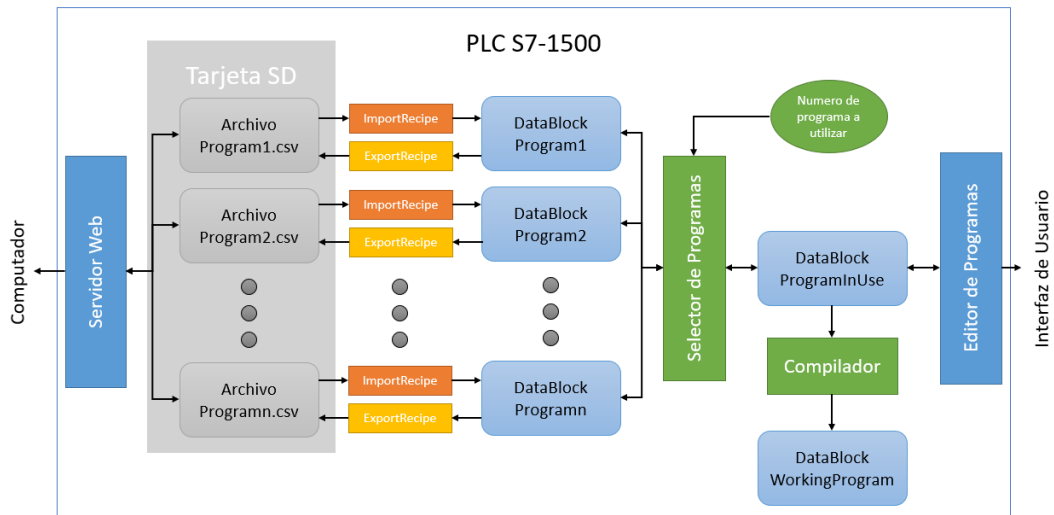


Figura 2.23 Diagrama de bloques para la lectura y escritura en la memoria no volátil

La figura 2.23 muestra el diagrama de bloques para la lectura y escritura de programas en la tarjeta SD. Los archivos creados en la tarjeta tienen un formato .CSV y se puede acceder a ellos por medio del servidor web del PLC, conectando cualquier computador a la subred PN/IE 2. La cantidad de programas que se pueden tener depende de la memoria de la tarjeta, pero no pueden ser cambiados por el usuario, por lo que desde un inicio se debe decidir qué cantidad de programas como máximo se podrán crear, en este proyecto se crearon 10 programas. El nombre de estos programas tampoco puede ser modificado por el usuario porque deben tener el mismo nombre que los bloques de datos del PLC, y estos se definen al momento de su creación.

Para leer la información de cada bloque de datos, se intentó acceder a cada uno según la elección del usuario, pero existían problemas porque se necesitaba crear un compilador y un editor para cada bloque de datos. Entonces se creó la función Selector de Programas para copiar la información del bloque de datos seleccionado hacia el bloque *ProgramInUse* y viceversa. Ahora, solo se necesita un compilador y un editor de programas que actúe sobre *ProgramInUse*.

2.6. Interfaz de usuario

La interfaz tiene la intención de hacer que el usuario cambie entre modos de funcionamiento del sistema y programas guardados en forma rápida, que pueda frenar el eje cuando quiera y apagarlo o encenderlo con un solo botón, y hasta que pueda editar programas mientras se está ejecutando otro sin tener pérdidas en rendimiento o retrasos en la ejecución de la tarea. La interfaz se programó en una pantalla táctil de 7 pulgadas del mismo fabricante que el PLC para facilitar su comunicación.

2.6.1. Panel de Mando

El panel de mando inicial tenía todos los botones directamente conectados a las instrucciones de control de movimiento, por lo que el usuario debía conocer como estos bloques trabajaban para que no ocurriera ningún error. Se mejoró la interfaz para no permitir que el usuario presionara un botón cuando no debía de hacerlo, incorporando validaciones y haciendo aparecer y desaparecer botones y texto según el modo en el que el sistema estaba funcionando. La figura 2.24 muestra las diferencias entre la antigua interfaz (esquina superior izquierda) y la definitiva (resto de imágenes dentro de la figura).

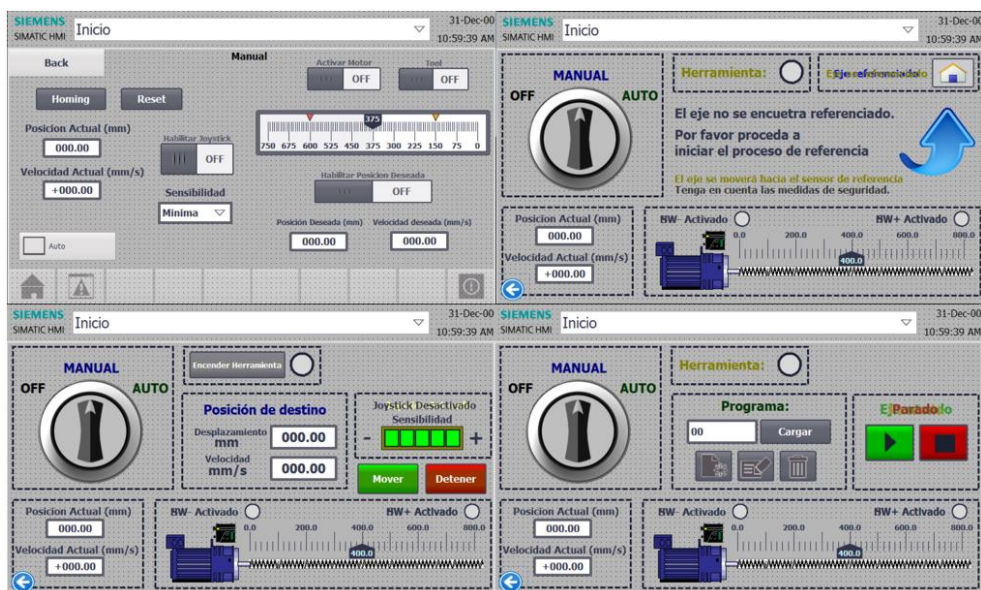


Figura 2.24 Diferencias entre la interfaz antigua y definitiva

2.6.2. Edición de programas

Se diseñó una pantalla para la edición de programas donde, en lugar de cargar todos los pasos, solamente se cargan 10 y se agregan y retiran valores según se presionen las flechas o la barra de desplazamiento, pero siempre manteniendo el número de 10 en la pantalla. La forma de cuadrícula permite una sencilla y rápida edición de los valores y programas dándole mayor versatilidad. La pantalla de edición se puede observar en la figura 2.25.

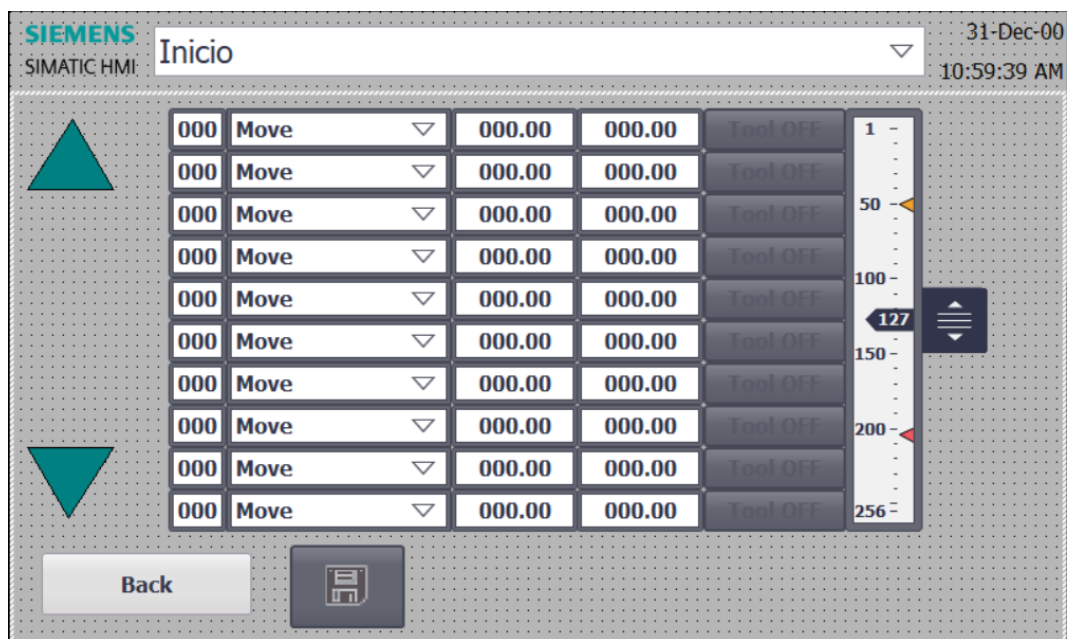


Figura 2.25 Pantalla de edición de programas

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

En este capítulo se muestran los resultados obtenidos de precisión y versatilidad del sistema. Además, se abarca el coste total del prototipo junto con su factibilidad y rentabilidad.

3.1. Lazos de control interno

Para obtener una precisión fina del movimiento del sistema, es necesario un regulador correctamente parametrizado. El sistema posee tres lazos de control ordenados de interior a exterior: regulador de intensidad de corriente, regulador de velocidad, y regulador de posición. Los dos primeros están presentes en el accionamiento del motor y se parametrizan con el software STARTER.

3.1.1. Calibración del regulador de intensidad

El regulador de intensidad es de tipo PI y tiene dos parámetros: ganancia y tiempo de acción de integración. Los parámetros utilizados se muestran en la figura 3.1. Mientras el valor de la ganancia fue obtenido automáticamente cuando se eligió el tipo de accionamiento y motor, el valor de tiempo de integración se redujo a 1 ms para que la respuesta del regulador sea más rápida.

Realizando las pruebas correspondientes se obtuvo una respuesta rápida con un error de estado estable del 5% y tiempo de estabilización de 2,6 ms. Lo que nos interesa de este regulador es que sea rápido, la forma de la respuesta sea fiel a la entrada, y no presente oscilaciones, lo cual se cumple. El error, aunque existe, es pequeño y no es de relevancia por tratarse de un lazo de control interno. La gráfica de la respuesta se puede observar en la figura 3.2.

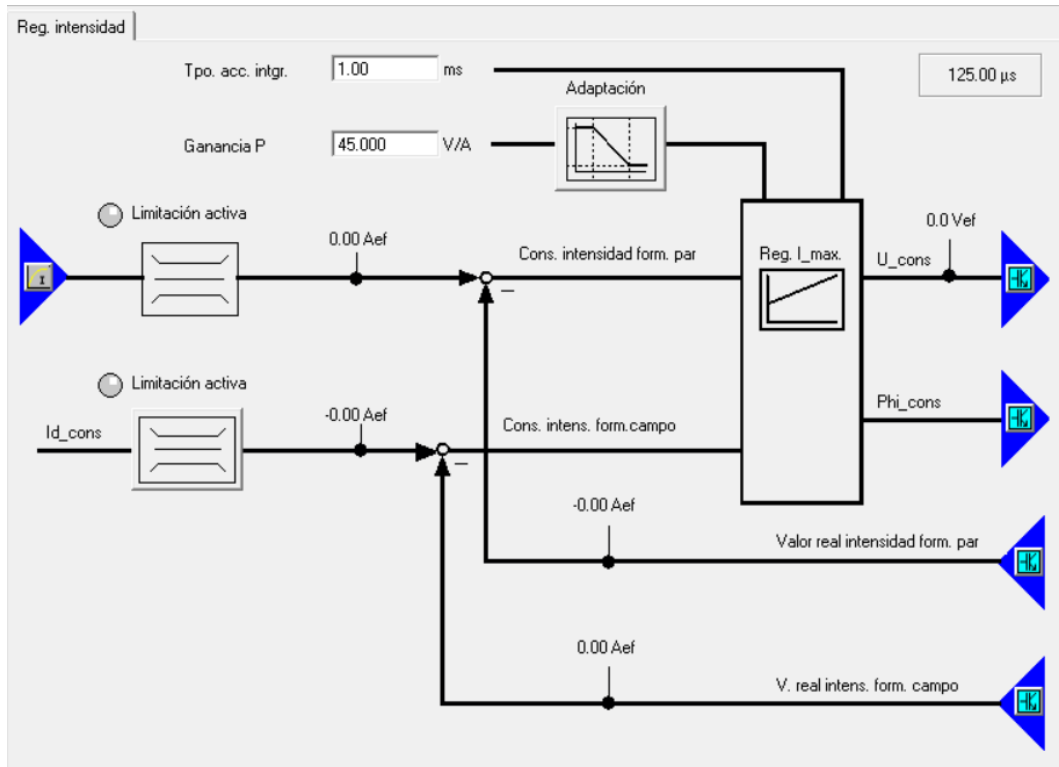


Figura 3.1 Diagrama de parámetros del regulador de intensidad

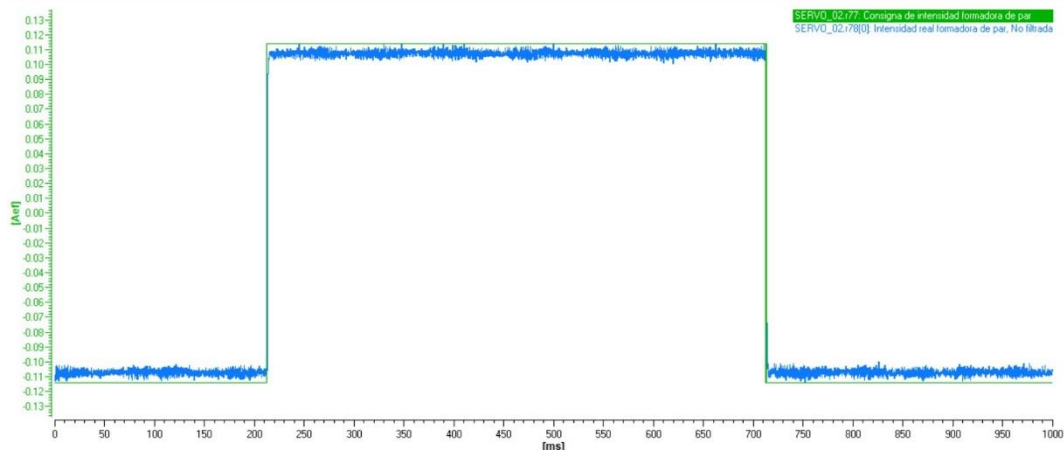


Figura 3.2 Respuesta del regulador de intensidad

3.1.2. Calibración del regulador de velocidad

Este regulador también es de tipo PI y sus parámetros fueron obtenidos por medio de la función de ajuste automático del software del accionamiento, se deben seguir los pasos de ajuste automático dejando que el motor se mueva

porque estos parámetros dependen mucho del tipo de carga acoplada al motor. Los valores obtenidos por el ajuste se muestran en la figura 3.3.

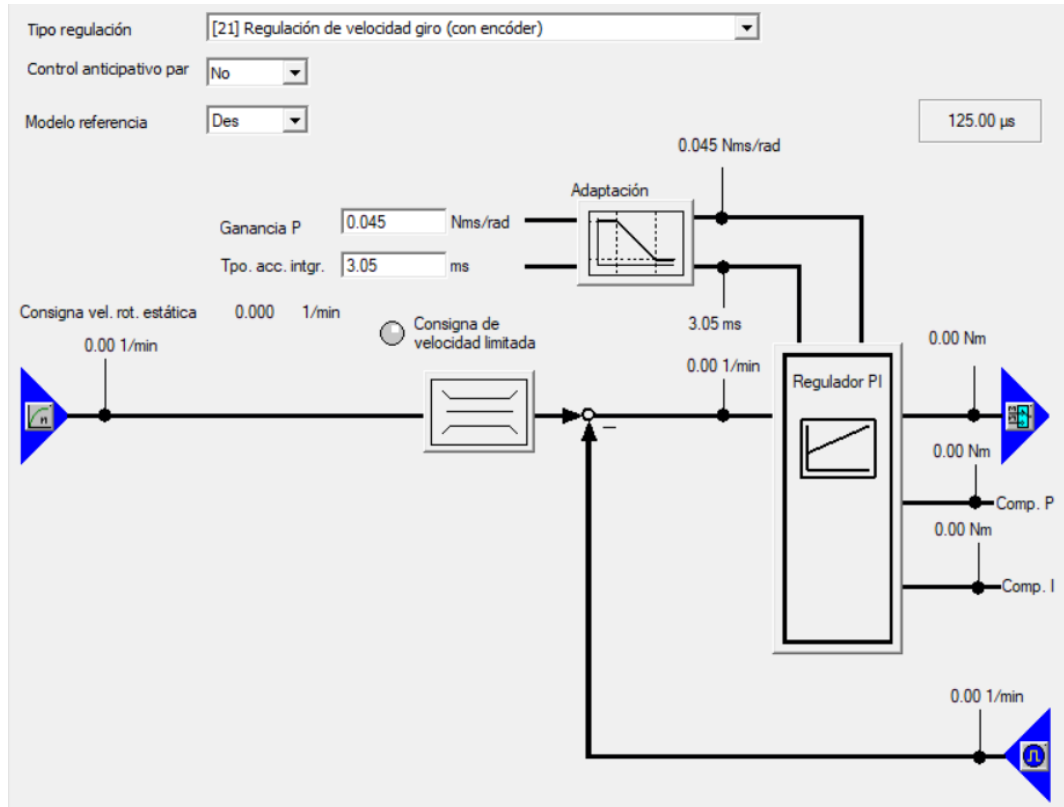


Figura 3.3 Diagrama y parámetros del regulador de velocidad

La respuesta del regulador con estos parámetros se muestra en la figura 3.4. Esta respuesta se estabiliza en 13,9 ms, provocando un error del 36% y un sobre pico inicial del 57%. Estos valores no son muy adecuados para el sistema, pero incluso pequeñas modificaciones en la ganancia provocaban mayores oscilaciones y un aumento del error de estado estable. Finalmente se decidió utilizar los valores del ajuste automático y concentrarnos en reducir el sobre pico y el error de la velocidad en el regulador de posición.

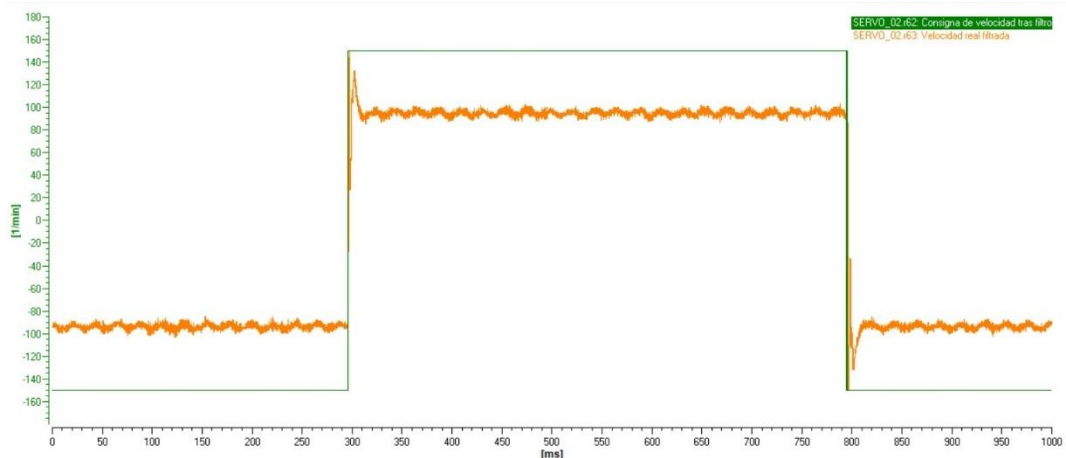


Figura 3.4 Respuesta del regulador de velocidad

3.2. Resultados y calibración del regulador de posición

Este regulador engloba todo el sistema y está ubicado en el PLC, su parametrización se realiza a través del objeto tecnológico configurado en el software TIA Portal. La ganancia normalmente no puede ser muy alta debido a que provocaría esfuerzos y oscilaciones exageradas como con los lazos internos de intensidad y velocidad, pero si se utiliza la tecnología de servo control dinámico, se puede aumentar esta ganancia para reducir el error de posición sin perjudicar la respuesta dinámica.

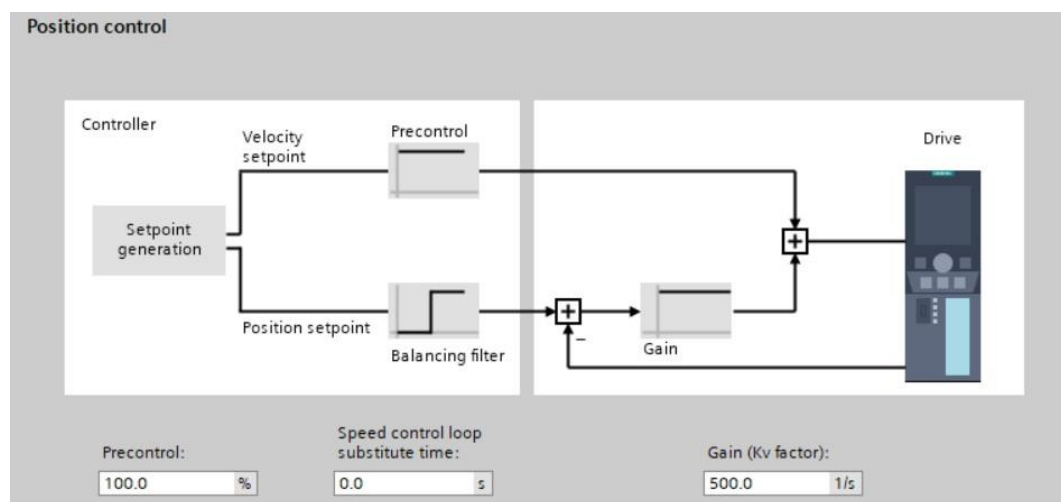


Figura 3.5 Diagrama y parametrización del regulador de posición

La figura 3.5 muestra los parámetros del regulador utilizados junto con el diagrama presente en el software. Al inicio se planteaba una ganancia de 5, pero se obtenía un error de estado estable de 0,03 mm. Se aumentó la ganancia hasta que el error

desapareciera y el resultado fue de 500. En la figura 3.6 se puede observar la gráfica obtenida en la prueba con una ganancia de 500, mientras que en la figura 3.7 se puede corroborar el error tanto en vacío como con una carga de 6 Kg. Los parámetros de precontrol y tiempo de sustitución se dejaron con los valores por defecto, cambiar estos parámetros producía un retardo en las respuestas lo cual era inadecuado para la aplicación.

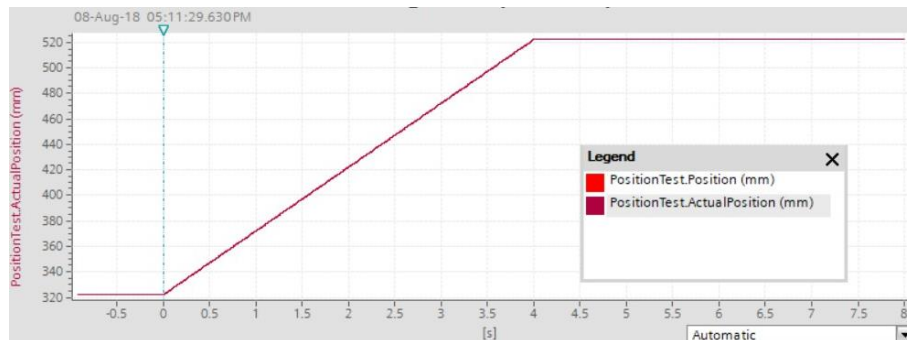


Figura 3.6 Respuesta del regulador de posición

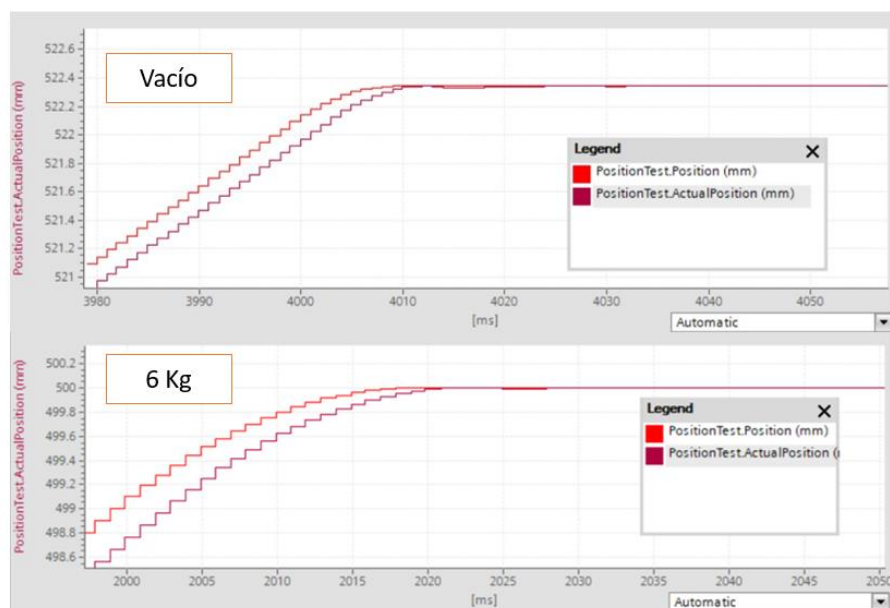


Figura 3.7 Comparación de respuesta en vacío y con carga

El error de seguimiento que se puede notar en la figura 3.7 se deben a un retardo del PLC en enviar la consigna al accionamiento, este retardo varía alrededor de 4 ms y provoca un error de seguimiento variable que no supera los 0.3 mm. Este

comportamiento es normal en todo tipo de sistemas y lo importante es asegurar que el error de estado estable sea cero, la figura 3.8 muestra este retardo.

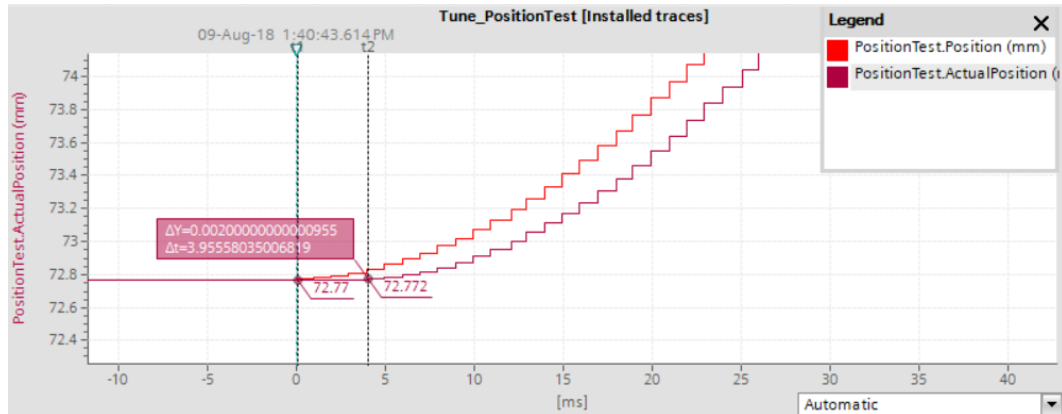


Figura 3.8 Retardo en el envío de la consigna

En cuanto al análisis de la velocidad, adquiriendo las mediciones desde el controlador obtenemos la respuesta mostrada en la figura 3.9. Aquí vemos como la dinámica mejoró con respecto a la figura 3.4 reduciendo considerablemente el sobre pico. En cuanto al error de estado estable, para la prueba se configuró una velocidad de 100 mm/s y el controlador, al medir el error existente en el accionamiento, aumentó la consigna para que el valor real sea el configurado y así eliminar el error del regulador de velocidad, obteniendo una respuesta exacta a la pedida.

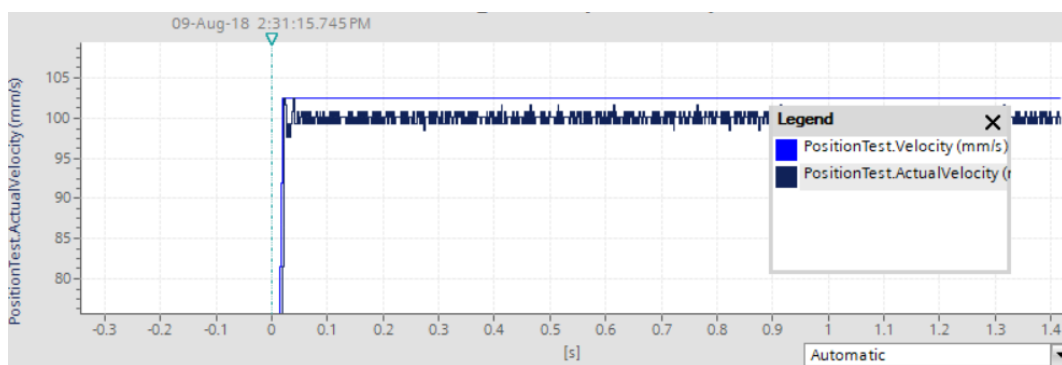


Figura 3.9 Respuesta de velocidad desde el PLC

3.3. Análisis de precisión del sistema

Aunque las pruebas del controlador muestren un error nulo en el sistema, se realizaron algunos cálculos extra para tratar de confirmar estos valores. Basándonos en la resolución del encoder y el paso del tornillo, la medición directa del sistema tiene una resolución de 0.003 mm.

Se realizó una serie de mediciones con la ayuda del sensor laser presentado en el capítulo 2 con los cuales se realizó la gráfica de la figura 3.10. En esta gráfica se representa el valor analógico del sensor versus el desplazamiento del tornillo en mm, los dos medidos por el PLC. Por el rango del sensor, que cada milímetro de desplazamiento del tornillo debería reflejar un valor de 1382,4. La gráfica de la figura 3.10 muestra una pendiente de 1379,7 dando un error de aproximadamente 0.002 mm en el sistema. Por lo que se decidió establecer en el sistema una precisión de 0.005 mm mientras que la mínima resolución que el usuario puede ingresar es de 0.01 mm.

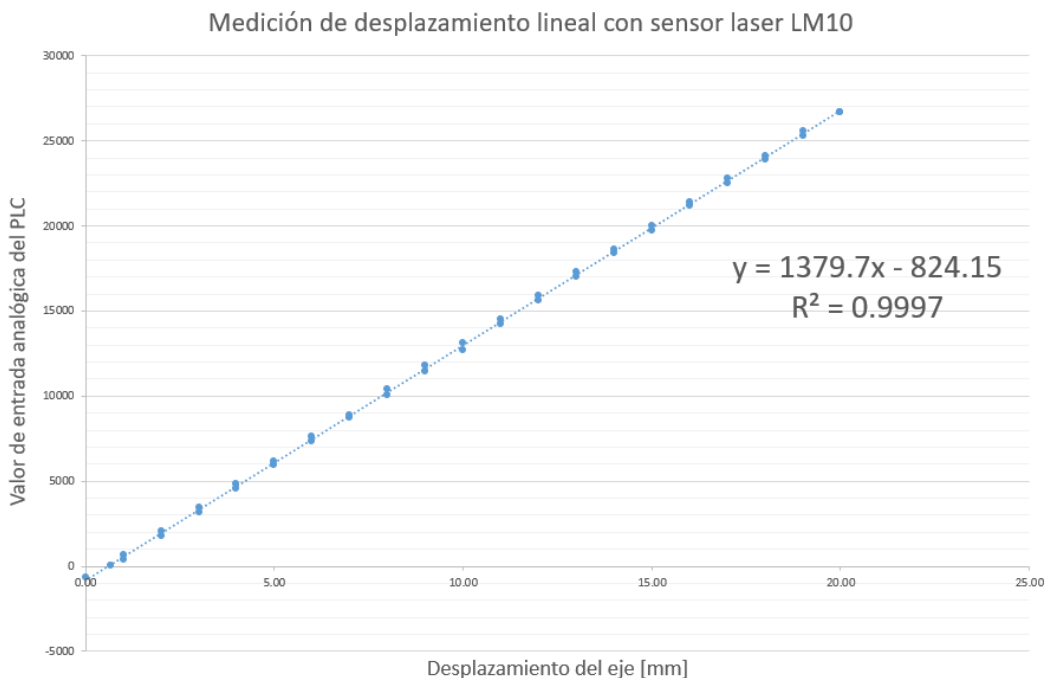


Figura 3.10 Medición del desplazamiento lineal con sensor láser

Con estos datos se asegura la precisión del sistema, pero otro factor importante que se midió fue el retardo en la programación del usuario. En la figura 3.11 se puede

observar el retardo que existe entre una orden enviada por el usuario en el modo manual, mientras que en la figura 3.12 se muestra el retardo del algoritmo en el modo automático. En ambos casos el retardo es de máximo 27 ms, lo cual es imperceptible para el ser humano.

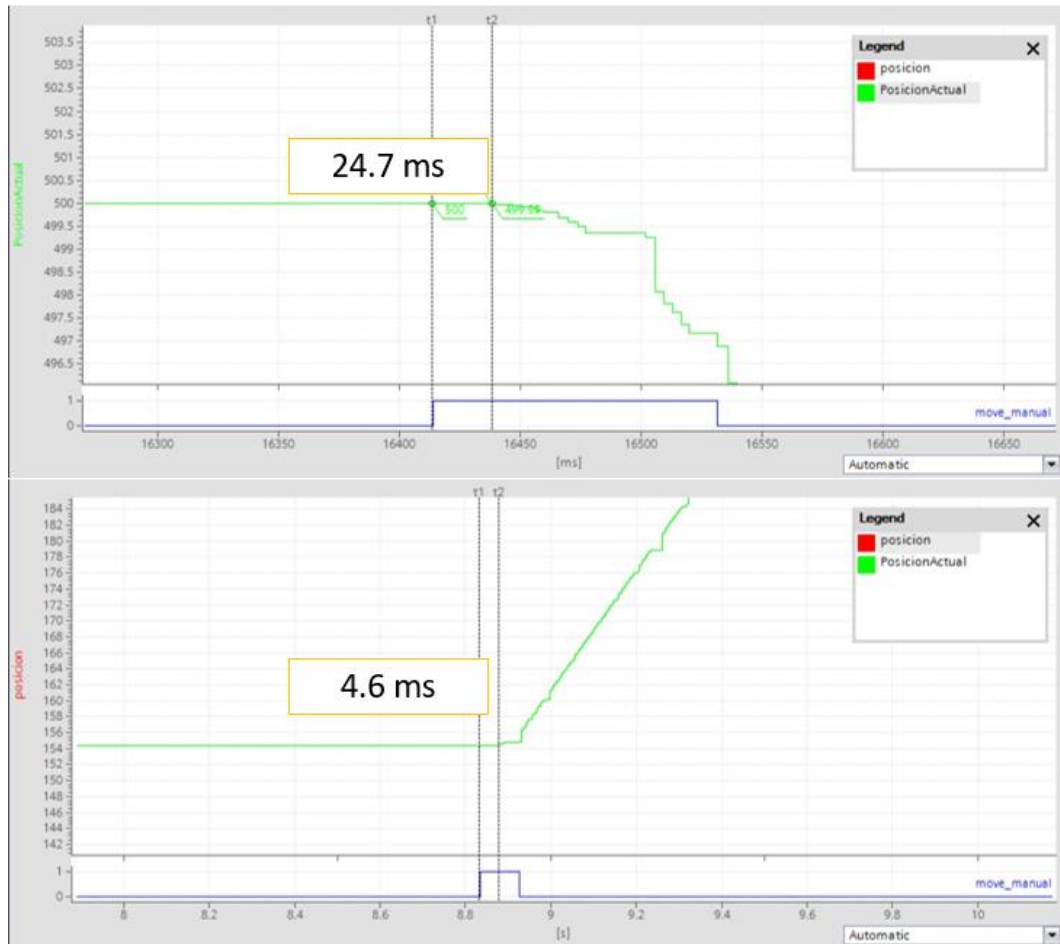


Figura 3.11 Retardo de acción del modo manual

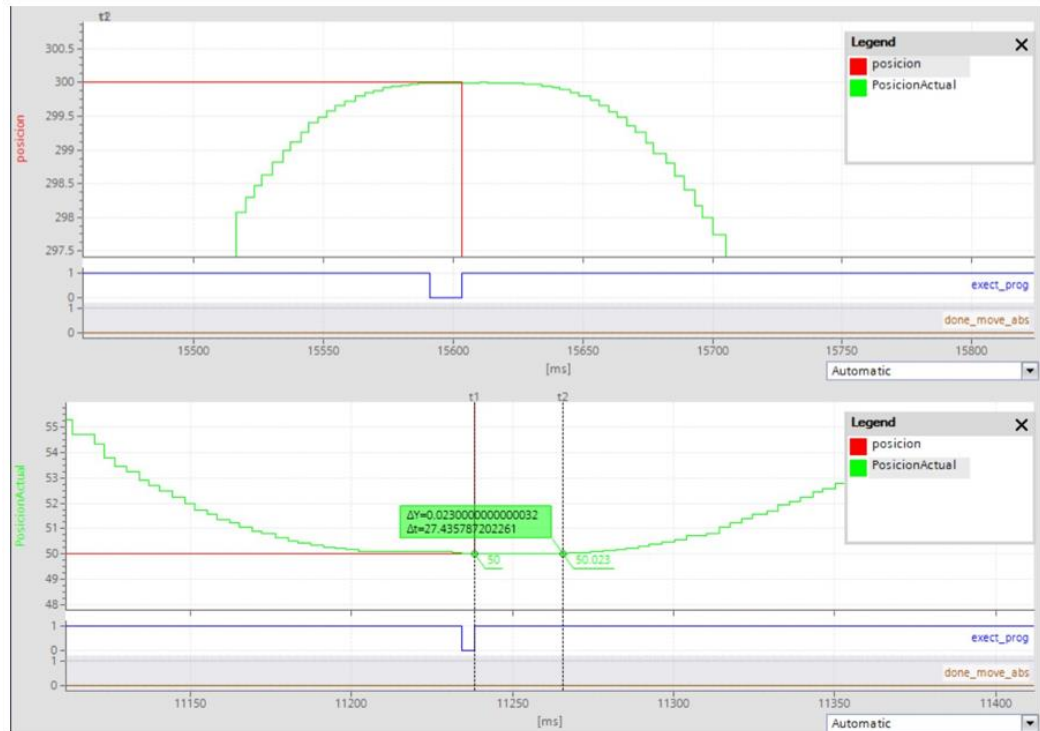


Figura 3.12 Retardo de acción del modo automático

3.4. Análisis de versatilidad del sistema

Con la interfaz fluida se obtuvo retroalimentación positiva por parte de los usuarios, mostrando interés en los colores, formas, imágenes y movimiento que presentaba la pantalla del panel de mando. Observaron que era fácil de utilizar, aunque necesitaron un tiempo para reconocer los botones y que se les respondieran algunas preguntas que tenían. La pantalla del panel de mando y la de inicio se pueden observar en la figura 3.13.



Figura 3.13 Pantallas de inicio y panel de mando

En cuanto a la forma de programación en la pantalla mostrada en la figura 3.14, se les dificultó su uso, por lo que esta pantalla necesita desarrollarse aún más. Sin embargo, reconocieron que la forma de descargar y cargar los programas desde una computadora externa por medio del servidor web del PLC era sencilla y más práctica para ellos. La página de los programas en el servidor web puede observarse en la figura 3.15, mientras que los programas descargados en la computadora en formato .CSV pueden editarse en un block de notas o en Excel como se muestra en la figura 3.16.



Figura 3.14 Pantalla de programación

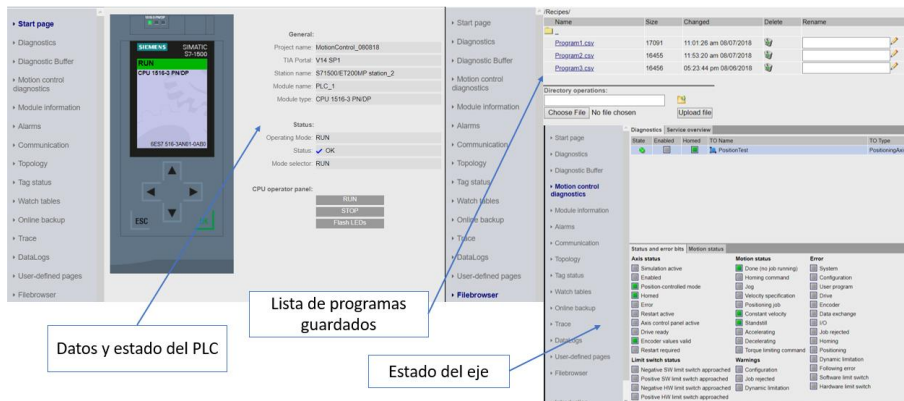


Figura 3.15 Servidor Web del PLC

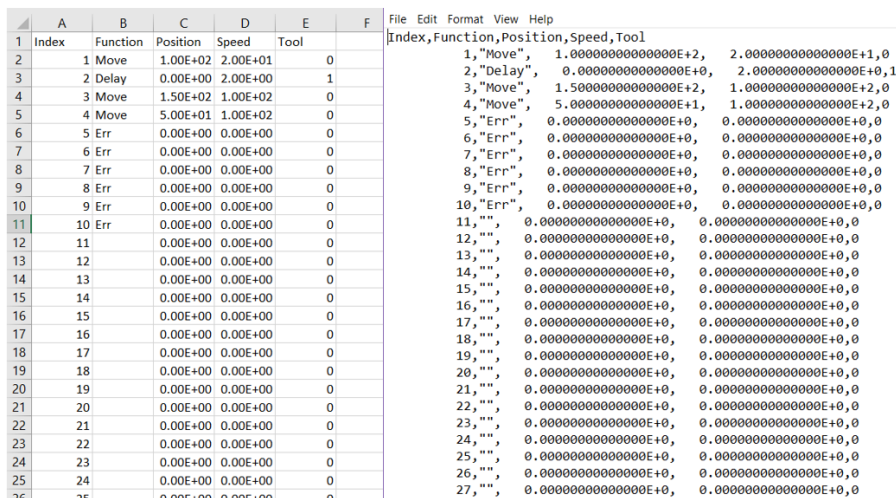


Figura 3.16 Edición de archivos .csv

3.5. Análisis de consumo de energía y factor de potencia

Uno de los criterios de selección del motor fue su bajo consumo de energía y alto factor de potencia en comparación con motores que realizan tareas similares. Mediante un Medidor de Energía (*FLUKE 1735 POWER LOGGER ANALYST*) se realizaron pruebas de medición de consumo y factor de potencia, en la figura 3.17 se puede observar cómo se realizaron las mediciones.

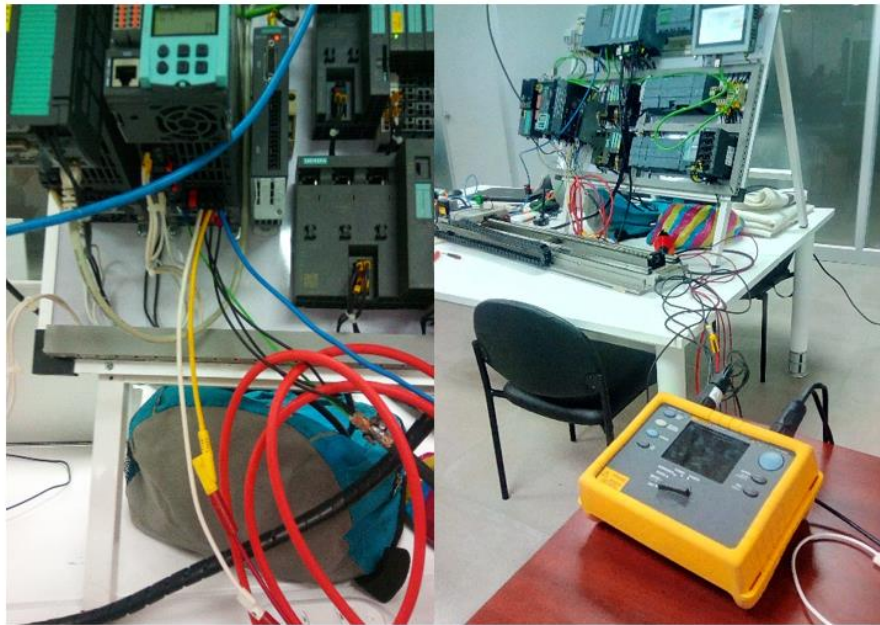


Figura 3.17 Equipo para la medición de energía

La corriente de consumo del motor es del orden de los miliamperios lo que dificultaba la medición de su magnitud y elaboración de gráficas debido a que el rango de medición de corriente del analizador de energía está en el orden de los amperios. Se realizaron dos mediciones, una a velocidad media de 100 mm/s y una a máxima velocidad de 250 mm/s. Los datos mostrados en el medidor de energía se pueden observar en las figuras 3.18 y 3.19.

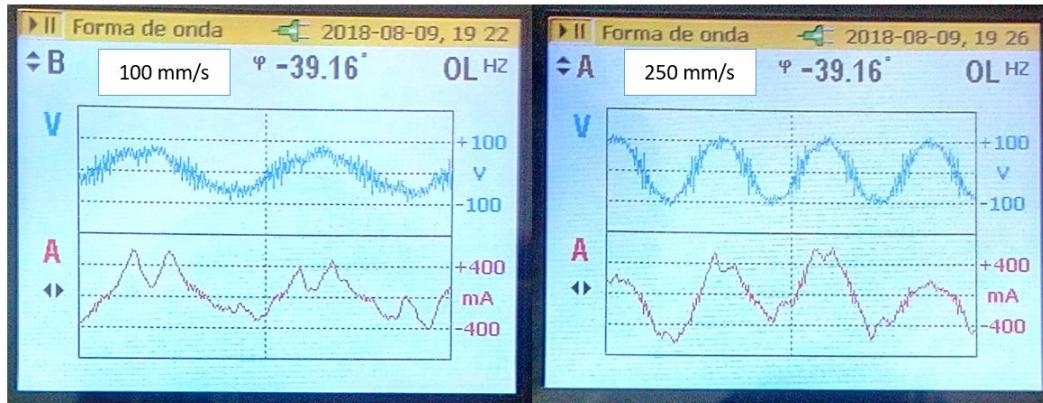


Figura 3.18 Mediciones de corriente

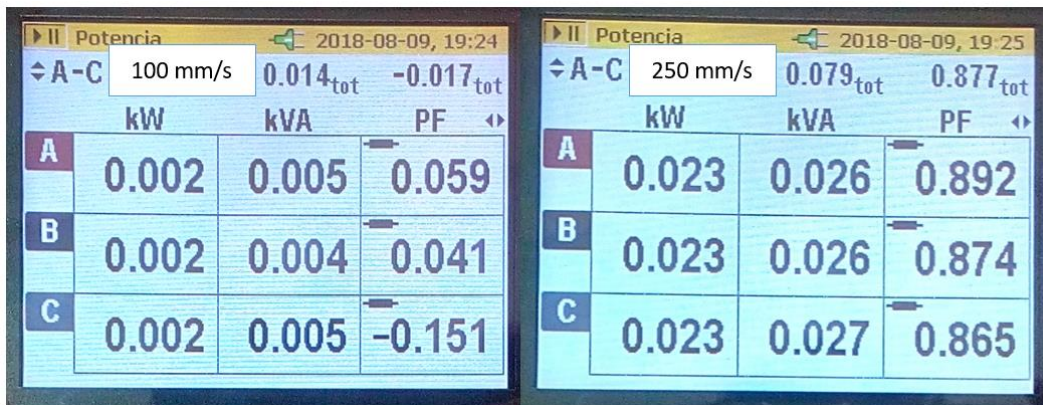


Figura 3.19 Mediciones de energía

La corriente máxima de consumo en vacío es de 350 mA y la potencia máxima es de 2.3 W. El valor mostrado en la figura 3.19 correspondiente al factor de potencia no representa una medida real debido a las incongruencias en la medición de corriente explicadas en párrafos anteriores. Además, gráficamente parece no haber desfase entre voltaje y corriente por lo que el factor de potencia es muy cercano a la unidad, tal y como lo dice la teoría.

3.6. Costo monetario de la implementación del prototipo

En este apartado se detalla los costos referentes tanto en materiales como en mano de obra, involucrados en el desarrollo del proyecto. Los precios de algunos equipos fueron obtenidos de [22].

Tabla 3.1 Costo de equipos y materiales

LISTA DE MATERIALES					
ITEM	Cantidad	Descripción	Unidad	V. Unit	V. Total
1	2	Disyuntor termomagnético 2P-16	u	\$ 22.00	\$ 44.00
2	4	Caja portafusible 10x38 1P-32 A	u	\$ 2.38	\$ 9.52
3	2	Fusible 10x38 2 A	u	\$ 0.97	\$ 1.94
4	2	Fusible 10x38 4 A	u	\$ 0.97	\$ 1.94
5	1	Fuente DC PM 190 W 120/230 VAC	u	\$ 360.00	\$ 360.00
6	1	Controlador Simatic S7-1500	u	\$ 3,500.00	\$ 3,500.00
7	1	Simatic Memory Card 24 MB	u	\$ 667.00	\$ 667.00
8	1	Módulo de 32 salidas digitales 24 VDC	u	\$ 1,069.00	\$ 1,069.00
9	1	Módulo de 8 entradas analógicas	u	\$ 1,449.00	\$ 1,449.00
10	1	Módulo de 4 salidas analógicas	u	\$ 1,223.00	\$ 1,223.00
11	1	Módulo de 32 entradas digitales	u	\$ 750.00	\$ 750.00
12	1	Simatic HMI Comfort Panel 7"	u	\$ 2,505.00	\$ 2,505.00
13	1	Software TIA PORTAL STEP 7 Profesional V14 SP1	u	\$ 4,880.00	\$ 4,880.00
14	1	Licencia para actualizar Tia Portal STEP 7 V14 SP1	u	\$ 1,875.00	\$ 1,875.00
15	1	Accionamiento motor Sinamics S120	u	\$ 816.00	\$ 816.00
16	1	Motor Síncrono Simotics 1FK7033	u	\$ 1,628.00	\$ 1,628.00
17	1	Cable de comunicación y fuerza motor	u	\$ 150.00	\$ 150.00
18	1	Computador	u	\$ 960.00	\$ 960.00
19	1	Juego de tornillo sin fin, rieles y bandeja transportadora	u	\$ 425.00	\$ 425.00
20	1	Base de acero inoxidable para sin fin	u	\$ 390.00	\$ 390.00
21	1	Soportería para montaje de motor	u	\$ 50.00	\$ 50.00
22	3	Sensor fin de carrera para detección de posición	u	\$ 1.00	\$ 3.00
23	1	Sensor Láser de posición LM-10	u	\$ 417.00	\$ 417.00
24	1	Controlador NAIS para sensor láser	u	\$ 537.00	\$ 537.00
25	1	Luz piloto 24 VDC	u	\$ 2.00	\$ 2.00
26	120	Cadena portacables 10x25	cm	\$ 0.20	\$ 24.00
27	5	Cable Profinet CAT 5 apantallado	m	\$ 6.00	\$ 30.00
28	6	Conectores blindados RJ45 industriales	m	\$ 60.00	\$ 360.00
29	1	Palanca de mando	u	\$ 50.00	\$ 50.00
30	global	Canaleta, cable, pernería, etc	u	\$ 400.00	\$ 400.00
		SUBTOTAL DE MATERIALES			\$ 24,577.40

Tabla 3.2 Costos de la mano de obra

MANO DE OBRA		
ITEM	Descripción	V. Total
1	Montaje de tablero de control	\$ 150.00
2	Instalación eléctrica de tablero de control	\$ 150.00
3	Montaje de sensores y actuadores	\$ 200.00
4	Instalación eléctrica de sensores y actuadores	\$ 90.00
5	Configuración y puesta en marcha de accionamiento de motor	\$ 300.00
6	Configuración de comunicación de controlador	\$ 100.00
7	Programación de algoritmo de control de PLC	\$ 600.00
8	Programación de interfaz en HMI	\$ 600.00
9	Pruebas de precisión y calibración de equipos	\$ 150.00
	SUBTOTAL DE MANO DE OBRA	\$2,340.00

Tabla 3.3 Costos totales del prototipo

PRESUPUESTO REFERENCIAL DEL PROTOTIPO DE MÁQUINA INDUSTRIAL	
MATERIAL	\$ 24,577.40
MANO DE OBRA	\$ 2,340.00
TRANSPORTE DE MATERIALES	\$ 300.00
TOTAL DE PROTOTIPO	\$ 27,217.40

3.7. Análisis de costos del proyecto escalado

El punto de partida para el escalamiento del prototipo es la selección de una máquina industrial comercial que cumple con el mismo principio de operatividad del prototipo. La máquina seleccionada se trata de un Robot Cartesiano de 3 ejes con aplicaciones de inyección y moldeo mostrada en la figura 3.20. Las características se observan en la tabla 3.4 y 3.5.



Figura 3.20 Robot Reis

Tabla 3.4 Parámetros generales de la máquina

Item	Parámetro	Descripción
1	Fabricante	Reis Robotics
2	Tipo	Cartesiano
3	Número de ejes	3
4	Aplicación	Inyección de moldeado
5	Carga útil	16 kg
6	Repetitividad	0.1 mm
7	Peso	300 kg
8	Costo	\$ 29 999.00

Tabla 3.5 Parámetros eléctricos de la máquina

Item	Parámetro	Descripción
1	Potencia nominal	5 kW
2	Corriente nominal	16 A
3	Voltaje de operación	480 V o 110 V
4	Voltaje de control	24 V

En base a estas características se dimensionan los detalles y materiales de la tabla 3.6.

Tabla 3.6 Costo de equipos y materiales del sistema escalado

LISTA DE MATERIALES					
ITEM	Cantidad	Descripción	Unidad	V. Unit	V. Total
1	1	Fuente DC PM 190 W 120/230 VAC	u	\$ 360.00	\$ 360.00
2	1	Controlador Simatic S7-1500	u	\$ 3,500.00	\$ 3,500.00
3	1	Simatic Memory Card 24 MB	u	\$ 667.00	\$ 667.00
4	1	Módulo de 32 salidas digitales 24 VDC	u	\$ 1,069.00	\$ 1,069.00
5	1	Módulo de 8 entradas analógicas	u	\$ 1,449.00	\$ 1,449.00
6	1	Módulo de 32 entradas digitales	u	\$ 750.00	\$ 750.00
7	1	Simatic HMI Comfort Panel 7"	u	\$ 2,505.00	\$ 2,505.00
8	1	Software TIA PORTAL STEP 7 Profesional V14 SP1	u	\$ 4,880.00	\$ 4,880.00
9	1	Licencia para actualizar Tia Portal STEP 7 V14 SP1	u	\$ 1,875.00	\$ 1,875.00
10	1	Accionamiento motor Sinamics S120 tipo bookSize para 3 ejes	u	\$ 3,700.00	\$ 3,700.00
11	3	Servomotores Simotics S 5 kW	u	\$ 1,628.00	\$ 4,884.00
12	3	Cable de comunicación y fuerza motor	u	\$ 150.00	\$ 450.00
13	1	Computador	u	\$ 960.00	\$ 960.00
14	6	Sensor fin de carrera para detección de posición	u	\$ 1.00	\$ 6.00
15	1	Estructura metálica de máquina	u	\$ 2,000.00	\$ 2,000.00
16	1	Sistema neumático	u	\$ 500.00	\$ 500.00
17	120	Cadena portacables 30x75	cm	\$ 1.00	\$ 120.00
18	15	Cable Profinet CAT 5 apantallado	m	\$ 6.00	\$ 90.00
19	6	Conectores blindados RJ45 industriales	m	\$ 60.00	\$ 360.00
20	global	Canaleta, cable, pernería, etc	u	\$ 400.00	\$ 400.00
		SUBTOTAL DE MATERIALES			\$ 30,525.40

Tabla 3.7 Costos de mano de obra para sistema escalado

MANO DE OBRA		
ITEM	Descripción	V. Total
1	Montaje de sensores y actuadores	\$ 200.00
2	Instalación eléctrica de sensores y actuadores	\$ 90.00
3	Configuración y puesta en marcha de accionamiento de motor	\$ 300.00
4	Configuración de comunicación de controlador	\$ 100.00
5	Programación de algoritmo de control de PLC	\$ 600.00
6	Programación de interfaz en HMI	\$ 600.00
7	Pruebas de precisión y calibración de equipos	\$ 150.00
	SUBTOTAL DE MANO DE OBRA	\$2,040.00

Tabla 3.8 Costo total del sistema escalado

PRESUPUESTO REFERENCIAL DEL PROTOTIPO ESCALADO	
MATERIAL	\$ 30,525.00
MANO DE OBRA	\$ 2,040.00
TRANSPORTE DE MATERIALES	\$ 300.00
TOTAL DE PROTOTIPO	\$ 32,865.00

3.8. Factibilidad y rentabilidad

El proyecto realizado es factible técnicamente, se pueden alcanzar precisiones muy altas y con una interfaz amigable con los equipos disponibles en el mercado. Una vez realizado el estudio del sistema, su replicación es mucho más sencilla. La programación es complicada y las mejoras tomarán tiempo por las continuas retroalimentaciones por parte del usuario que se necesitan, pero no son imposibles de realizar.

Acerca de la rentabilidad del proyecto se concluye que los costos de elaboración de máquinas escaladas a tamaños y potencia que demande el cliente están muy a la par de los costos de las máquinas comerciales, con el plus de la versatilidad e interfaz incluidas resultan en una inversión de mayor provecho que se reflejará en una producción más dinámica, menos pérdidas energéticas y paradas de producción debido a reprogramación y mantenimiento. Por lo tanto, el proyecto es rentable hasta cierto punto de diseño y operabilidad.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

En este capítulo se mencionarán las fortalezas del proyecto en base a los resultados obtenidos, qué mejoras deben hacerse y se mencionarán los trabajos futuros que se pueden realizar sobre el tema.

4.1. Conclusiones

La tecnología utilizada para el sistema de control de movimiento probó ser de los más eficiente que se pudo escoger para la aplicación. Aunque los equipos son costosos, su precio es justificado por todos los resultados y opciones de escalabilidad que brinda. Se logró montar el prototipo relativamente rápido, el fabricante ofrece todas las facilidades técnicas para la configuración y puesta en marcha de los equipos y la tecnología de comunicación demostró ser eficiente y confiable. Algunas veces se piensa que se sobrevalora los equipos industriales por la marca, pero se debe tomar en cuenta todo el trabajo de ingeniería que se ha realizado para que estas soluciones puedan implementarse. El sistema tiene una precisión que se puede igualar a las máquinas en el mercado actual, tanto en posición como en velocidad. Tiene la ventaja de usar un motor síncrono que le permite moverse tan rápido para reducir tiempos de producción como tan lento para procesos delicados, sin perder fuerza o precisión sobre sus acciones. Con las velocidades altas de 200 mm/s se puede realizar tareas de pintura o transporte mientras que con las velocidades bajas de 10 $\mu\text{m/s}$ se puede realizar tareas de colocación de sustancias de una manera que sería imposible de lograr por una persona, esto lo diferencia de los sistemas que utilizan motores DC o Stepper ya que estos no pueden moverse tan lento sin perder su fuerza y su precisión, y por ello se justifica el uso de la tecnología de comunicación para controlar un servomotor síncrono que es mucho más complicada que para un motor de otro tipo. La comunicación por medio de Ethernet evita que existan interferencias y que los lazos de control se mantengan siempre estables, el diagnóstico y estado de los equipos involucrados en la topología se comparte y puede ser visualizada para identificar

fallos rápidamente, incluso si se requiere hacer un cambio de equipos basta con cargar la configuración desde la tarjeta SD para que el nuevo equipo tenga exactamente los mismos parámetros que el anterior. El prototipo no mostró reducción de precisión con la adición de peso en su plataforma, los 6 kg que se utilizaron bien podrían referirse a los demás ejes de la máquina completa y los valores de consumo de corriente tampoco aumentaron bruscamente. Esto se debe a la mecánica del tornillo sin fin, resulta imposible para una persona mover la plataforma directamente incluso con el motor desenergizado, solamente se puede mover ejerciendo fuerza directamente sobre el eje rotacional del motor; y cuando el prototipo está encendido y el motor está trabajando, la fuerza que ejerce hace que no sea afectado por perturbaciones del sistema, manteniendo una velocidad constante hacia su destino sin importar si se agrega o quita peso de la plataforma. Si ya se mencionó la precisión y estabilidad del sistema, tenemos que referirnos también de su dinámica. Las aceleraciones que puede soportar son demasiado altas gracias a que el motor está específicamente construido para aplicaciones de alta dinámica, permitiendo arranques y frenados inmediatos sin picos de corriente o vibraciones exageradas. Haciéndolo trabajar por varias horas, su temperatura se mantuvo a menos de 30°C demostrando que en realidad podría soportar más aceleraciones y peso sin problemas. En conclusión, para obtener un sistema preciso y confiable se debe de realizar la inversión correspondiente, existen sistemas más baratos, pero tienen características inferiores a las presentadas.

La programación del PLC permitió el fácil control del sistema por parte del usuario, aunque aún hay lugar para mejoras. La interfaz del panel de mando, junto con las animaciones de botones y textos hacen que el usuario pueda visualizar rápidamente la información que necesita conocer en ese momento. Existe una diferencia notable en colocar una pantalla llena de botones y cuadros de valores rectangulares a utilizar imágenes vectorizadas y colores para identificar el estado del sistema; el usuario aumenta su motivación para trabajar con la interfaz y es un tema que muchos proyectos pasan por alto. El desarrollador olvida que al final quien va a utilizar el equipo es una persona normal, que trabajará con él durante largas jornadas, y por lo tanto requiere un ambiente de trabajo cómodo, tanto físico como en sus herramientas de software. Otro aspecto importante que tomar en cuenta es el tipo

de programación que se utilizó, al utilizarse funciones tipo librería, se puede escalar el proyecto al número de ejes que se requiera sin tener que volver a plantear los algoritmos utilizados. Además, la programación en paralelo permite que la máquina pueda estar trabajando mientras el usuario edita en el mismo sistema el próximo programa a utilizar; existen máquinas comerciales que no permiten esta tarea, y se limitan a tener que editar los programas utilizando un computador externo, generando mayores costos por el tiempo que conlleva movilizarse a la estación de trabajo y cargar o descargar los programas para probar cada cambio realizado; o en el peor de los casos, la máquina debe de pararse para que el usuario pueda editar o cargar nuevos programas. Esto demuestra la complejidad de la programación y el valor agregado que se le da al producto final; podemos tener dos sistemas con la misma tecnología, consumo y precisión, pero con interfaces y utilidad totalmente diferentes, lo cual al final se puede ver reflejado en el precio del sistema. Es por ello por lo que, en conclusión, un ingeniero nunca debe de olvidar que su solución va a ser utilizada por un usuario final, y debe enfocar su diseño a facilitar el trabajo de ese usuario. Sucede algo similar con la integración del joystick en el sistema, se lo agregó pensando en que el usuario final no sabría qué valor de velocidad colocar en el programa y que pudiera utilizar el joystick para mover el eje hasta que encontrara la velocidad adecuada, la guardara, y la ingresara para que el programa automático la replique, haciendo una migración rápida de un trabajo manual humano a un trabajo automático preciso.

El rendimiento energético de este tipo de sistema se intentó medir, pero las condiciones no permitieron obtener resultados confiables. Las gráficas y valores mostrados de corriente y energía solo se pudieron adquirir del motor, y al final lo importante es tener una lectura de estas variables en la parte de alimentación general de los equipos. El consumo era muy pequeño para medirlo con los instrumentos que se tenían a mano, y las interferencias del resto de equipos en la red eléctrica del lugar de trabajo mostraban armónicos mayores a las corrientes de consumo. En cuanto al tiempo de edición de los programas se obtuvieron resultados positivos, pudiendo cambiar entre programas de una manera rápida. La utilización de botones de pausa y paro en el algoritmo de modo automático permitió tener mayor control sobre la ejecución de las tareas, a esto debemos agregar el retardo

minúsculo en la ejecución de los programas que depende del número de pasos que tenga el mismo, generando un retardo de 1 segundo cada 37 pasos aproximadamente. Tiempos de producción y consumo energético son rubros que deben tomarse en cuenta como costo por operación de un sistema, y aunque los costos de compra inicial entre este proyecto y los existentes en el mercado sean similares, este rubro al final decidirá el costo final total del sistema y cuál debe ser la elección del cliente. Para realizar una comparación exacta de los costos de operación se necesitaba contar con los equipos finales terminados, y por ser este proyecto un prototipo, no se pudo realizar dicha comparación.

Aunque este proyecto fue concebido como prototipo para la construcción de máquinas cartesianas, la tecnología de control de movimiento tiene aplicaciones en muchas otras áreas. Existen procesos que requieren controles de velocidad o par y no de posición, pero este proyecto también puede ser modificado para cumplir esos requerimientos específicos. Si bien existen soluciones comerciales más avanzadas y que son recomendadas por los propios fabricantes para trabajos de máquinas cartesianas, este sistema nos permitió aprender el funcionamiento del control de movimiento; sus ventajas y falencias fueron analizadas y permiten tener una mayor comprensión de este tipo de implementaciones. Muchas máquinas en la industria poseen varias partes interconectadas, y tal vez algunas de ellas son de control de movimiento, pero viven en conjunto y son controladas al unísono para conformar un todo. Es importante comprender que podemos utilizar este proyecto como prototipo de una máquina cartesiana que funciona enteramente con control de movimiento, o lo podemos utilizar como parte de un gran proceso, donde quizás existen varios ejes a lo largo de la línea de producción, cada uno independiente con tamaños o formas diferentes, pero para poder configurarlos y asegurar su precisión se requieren los mismos conocimientos. Incluso las definiciones y funciones de la interfaz y la versatilidad de este proyecto no solamente se pueden utilizar para máquinas cartesianas, sino que deben ser implementadas en cualquier proyecto de ingeniería que se realice, sea o no para la industria, siempre teniendo claro el usuario final de nuestra solución.

4.2. Recomendaciones

Acerca de la versatilidad e interfaz que se logró con la pantalla HMI, en la cual se muestra la manera más sencilla y ergonómica para realizar cambios en la rutina de máquina o realizar movimientos precisos de piezas especiales, se necesita mayor trabajo para que pueda considerarse una máquina industrial completa. Se debe agregar una forma de cargar variables de posición o velocidad para simplemente colocar el nombre de la variable en la rutina de trabajo, si posteriormente se necesita cambiar esa posición o velocidad bastará con cambiar el valor de la variable sin necesidad de editar de nuevo el programa. Se debe agregar también un creador de producción en el cual se pueda especificar que rutina se va a correr y cuantas veces queremos que lo haga, así se podría configurar por ejemplo que se fabriquen 100 piezas de un tipo y luego inmediatamente se comience la producción de 50 piezas de otro tipo o tamaño, colocando también un visualizador para saber cuántas piezas de cada tipo se van realizando y cuánto tiempo falta para la culminación de la producción. El prototipo posee un servidor web para la carga y descarga de programas, el proyecto solo se basó en la comunicación directa a este servidor por medio de un cable ethernet, pero se debe ampliar esta comunicación para que un computador, en una oficina lejos de la planta, pueda configurar y visualizar el diagnóstico del sistema; esta solución se puede implementar utilizando un cableado por medio de switches industriales o por internet utilizando routers y VPNs. En cuanto al joystick utilizado, se recomienda utilizar uno de tipo industrial con más botones para una reducción de zona muerta, mayor linealidad, y para no tener que acercarnos a la pantalla a cambiar la sensibilidad del modo manual continuo; también se pueden conectar un teclado y un ratón a la pantalla para más comodidad al momento de editar los programas. Una alternativa de menos costo es la utilización de un tablero de control con botoneras, pulsadores y potenciómetros para asignar las cuatro funciones básicas de un accionamiento de motor que son: arranque, freno, inversión de giro y variación de velocidad. Este caso particular puede suceder en aplicaciones de procesos sencillos o muy específicos diseñados para una sola labor, siempre y cuando no se pierdan los conceptos de control de movimiento y precisión del mecanismo. También existiría una reducción de costos por el accionamiento y la capacidad del controlador.

Existen equipos elaborados por el fabricante Siemens que prestan características exclusivas para el manejo de máquinas industriales y procesos de servo aplicaciones en el traslado y acción de un elemento terminal, estos equipos son derivados de la línea Sinumerik y Simotics que se destacan por su potente procesamiento numérico en labores de máquinas de taller y de máxima precisión. Además, Sinumerik cuenta con una interfaz de lenguaje G que le permite al usuario ingresar modelos de bocetos o dibujos de piezas que desea que la máquina fabrique, creados en software CAD, así como datos para instrucciones exclusivas para el accionamiento de maquinaria CNC. Esto cierra completamente la brecha entre la máquina y el diseñador, permitiendo pasar de un diseño a un prototipo en pocos minutos. Si incluimos también conectividad por internet, se obtendría un proyecto de industria 4.0, donde el diseñador gráfico crea un nuevo producto y lo envía inmediatamente al sistema para que lo fabrique con manufactura aditiva, reduciendo los tiempos de salida al mercado de productos. Estos equipos también facilitan la interpolación de trayectorias y sincronización de varios ejes, permitiendo una escalabilidad más sencilla. Aunque la programación realizada es escalable para mayor número de ejes, si el equipo brinda las herramientas para la creación de trayectorias complejas y contiene librerías específicas para estos trabajos, se deben aprovechar, por lo que se recomienda analizar un trabajo realizado con estos equipos más avanzados para identificar si se justifica su adquisición en comparación con el prototipo planteado en este proyecto.

Con el objetivo de medir y confirmar los desplazamientos que realiza la plataforma del eje se recomienda utilizar sensores láser de medición de desplazamiento de un alcance de por lo menos toda la longitud del espacio de trabajo. Las mediciones realizadas ocurrieron en los primeros milímetros del tornillo, pero si existen desviaciones o deformaciones en tramos más alejados no se pueden comprobar tan fácil. Estos sensores poseen un costo muy elevado por lo que no pudieron ser adquiridos para este proyecto, tienen una sensibilidad muy alta (menos de 10 micrómetros) y prestan mucha confiabilidad a la hora de verificar la precisión de movimiento de la herramienta terminal. Se intentó medir estos desplazamientos con reglas y cintas métricas junto a nonios semejantes a los calibradores vernier, pero

las reglas utilizadas tenían un error de milímetros entre sí, por lo que sus mediciones fracasaron y se descartaron. Estas mediciones servirán para confirmar la precisión del sistema, ya que el controlador mide desplazamientos indirectamente confiando en que el paso del tornillo es el adecuado y es constante en toda su longitud, por ello se necesita un análisis mayor de la mecánica del sistema.

Se recomienda ampliar las funciones del modo automático, permitiendo crear bucles de rutinas y otros tipos de control externo. Por ejemplo, se pueden utilizar las mediciones de corriente y par del motor para hacer un control de fuerza del eje y permitir al usuario crear un programa para mover el eje a una cierta velocidad hasta que la fuerza que ejerza sea de cierto valor límite para compactar bloques de cartón o presionar pernos. También, se pueden colocar protecciones de fuerza para que cuando la plataforma choque con un obstáculo o se tope con el cuerpo de un operador, el sistema detenga el motor e informe de la alarma producida. Estas funciones requieren mucha programación adicional, pero es factible su realización.

BIBLIOGRAFÍA

[1] R. Vaughn, "The difference between Cartesian, six-axis, and SCARA robots," *Mach. Des.*, 2013.

Disponible en: <https://search.proquest.com/docview/1662436355?accountid=171402>.

[2] S. Kendrick, "Understanding Your Options," *Plastics Technology*, vol. 60, (5), pp. 46-47,49,51, 2014.

Disponible en: <https://search.proquest.com/docview/1528145503?accountid=171402>.

[3] N. Elkmann *et al*, "Kinematics, sensors and control of the fully automated façade-cleaning robot SIRIUSc for the Fraunhofer headquarters building, Munich," *The Industrial Robot*, vol. 35, (3), pp. 224-227, 2008.

Disponible en: <https://search.proquest.com/docview/216996941?accountid=171402>.

DOI: <http://dx.doi.org/10.1108/01439910810868543>.

[4] W. You *et al*, "Control system design for heavy duty industrial robot," *The Industrial Robot*, vol. 39, (4), pp. 365-380, 2012.

Disponible en: <https://search.proquest.com/docview/1020672394?accountid=171402>.

DOI: <http://dx.doi.org/10.1108/01439911211227944>.

[5] J. Chinnasamy *et al*, "A workbench for motion control experiments using programmable automation controllers in industrial automation laboratory at Kongu Engineering College," *Computer Applications in Engineering Education*, vol. 26, (3), pp. 566-576, 2018.

Disponible en: <https://search.proquest.com/docview/2047351404?accountid=171402>.

DOI: <http://dx.doi.org/10.1002/cae.21908>.

[6] P. Pillay and R. Krishnan, "Modeling, simulation, and analysis of permanent-magnet motor drives. I. The permanent-magnet synchronous motor drive," en *IEEE Transactions on Industry Applications*, vol. 25, no. 2, pp. 265-273, Mar/Apr 1989. Disponible en: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=25541&isnumber=966>.

[7] B. A. Ishak and I. A K M Parvez, "Design and fabrication of a Cartesian painter robot for the construction industry," *International Journal of Mechanical Engineering Education*, vol. 34, (2), pp. 125-127,129-144, 2006.

Disponible en: <https://search.proquest.com/docview/200210224?accountid=171402>.

[8] Freescale Semiconductor, Inc. (2012). Permanent Magnet Synchronous Motor Control.

Disponible en:

<https://cache.freescale.com/files/industrial/doc/brochure/BBPRMMAGSYNART.pdf>.

[9] W. Labs, "PLCs and PACs: Know when and where to use them," *Food Engineering*, vol. 88, (12), pp. 55-58,60-61, 2016.

Disponible en: <https://search.proquest.com/docview/1865196769?accountid=171402>.

[10] T. Jensen, "Easy Motion System Integration: Practical Strategies, Tools," *Control Eng.*, 2012.

Disponible en: <https://search.proquest.com/docview/1023319449?accountid=171402>.

[11] S. V. Murphy and A. Atala, "3D bioprinting of tissues and organs," *Nat. Biotechnol.*, vol. 32, (8), pp. 773-85, 2014.

Disponible en: <https://search.proquest.com/docview/1551941052?accountid=171402>.

DOI: <http://dx.doi.org/10.1038/nbt.2958>.

[12] D. D. Camacho *et al*, "Applications of additive manufacturing in the construction industry – A prospective review," en 2017,

Disponible en: <https://search.proquest.com/docview/1943517631?accountid=171402>.

[13] B. Rooks, "Robot welding in shipbuilding," *The Industrial Robot*, vol. 24, (6), pp. 413-417, 1997.

Disponible en: <https://search.proquest.com/docview/217009405?accountid=171402>.

[14] Anonymous "Global Motion Control Market," *M2 Presswire*, 2017.

Disponible en: <https://search.proquest.com/docview/1857255567?accountid=171402>.

[15] Directorio de Empresas, Instituto Ecuatoriano de Estadística y Censo [online].

Disponible en:

http://produccion.ecuadorencifras.gob.ec/geoqlik/proxy/QvAJAXZfc/opendoc.htm?document=empresas_test.qvw&host=QVS%40virtualqv&anonymous=true.

[16] Siemens. (2018, Febrero). Configuration Manual - SIMOTICS Synchronous motors SIMOTICS S-1FK7 Generation 2 For SINAMICS S120.

Disponible en: <https://support.industry.siemens.com/cs/document/55379345/simotics-s-1fk7-g2-synchronous-motors?dti=0&pnid=13349&lc=en-WW>

[17] Siemens. (2018, Febrero). Equipment Manual - SINAMICS S120 AC Drive.

Disponible en: <https://support.industry.siemens.com/cs/document/109757057/sinamics-s120-ac-drive?dti=0&lc=en-WW>

[18] Siemens. (2017, Noviembre). Function Manual - SINAMICS S120 Drive Functions.

Disponible en: <https://support.industry.siemens.com/cs/document/109754299/sinamics-s120-function-manual-drive-functions?dti=0&lc=en-WW>

[19] Siemens. (2016, Octubre). SIMATIC S7-1500/ET 200MP Automation System in a nutshell. Disponible en:

<https://support.industry.siemens.com/cs/document/109481357/simatic-s7-1500-et-200mp-automation-system-in-a-nutshell?dti=0&lc=en-WW>

[20] Siemens. (2016, Septiembre). Manual de Funciones - SIMATIC S7-1500 Motion Control V3.0 en el TIA Portal V14. Disponible en:

https://cache.industry.siemens.com/dl/files/589/109739589/att_897265/v1/s71500_motion_control_function_manual_sp-SP_es-ES.pdf

[21] Panasonic. (1998). Micro Laser Displacement Sensor LM10 Catalog. Disponible en:

https://www3.panasonic.biz/ac/na_download/fasys/measurement/measurement/catalog/lm10_e_cata.pdf?f_cd=403765

[22] Siemens. (2018, Enero). "Lista de precios de productos eléctricos industriales".