



UNIVERSIDAD TÉCNICA DEL NORTE

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN MECATRÓNICA**

ARTÍCULO CIENTÍFICO

TEMA:

**SISTEMA MODULAR PARA REALIZAR PRÁCTICAS DE
SERVOMECANISMOS Y CONTROL MEDIANTE DESLIZADOR
LINEAL E INTERFAZ CON LABVIEW**

AUTOR: Ernesto Vladimir Palacios Merino

DIRECTOR: Ing. Diego Terán

Ibarra - Ecuador
2014

SISTEMA MODULAR PARA REALIZAR PRÁCTICAS DE SERVOMECANISMOS Y CONTROL MEDIANTE DESLIZADOR LINEAL E INTERFAZ CON LABVIEW

Palacios Merino, Ernesto Vladimir
mecatronica.mid@gmail.com
Universidad Técnica del Norte

Resumen—El presente trabajo describe el diseño e implementación de un deslizador lineal, el mismo que ha sido acoplado para su uso con el servomotor GSK 110SJT. Para llevar a cabo el control de este mecanismo se han creado librerías e interfaces en el lenguaje de programación gráfico de LabVIEW, dichas librerías han sido elaboradas con el ánimo de que brinden a los estudiantes de mecatrónica de la Universidad Técnica del Norte, facilidad de uso y escalabilidad en sus proyectos, animando a la creación de programas y mecanismos más elaborados. Una vez que se ha completado la parte mecánica, eléctrica y electrónica se cuenta con un sistema, en el cual se interrelacionan todos sus módulos y subsistemas, a través de una librería y protocolos de comunicación definidos, pudiendo de esta manera llevar a cabo prácticas de servomecanismos y control.

Índice de Términos—LabVIEW (Plataforma de desarrollo virtual gráfico). API (Application Programming Interface)

I. INTRODUCCIÓN

EL PRESENTE TRABAJO DE GRADO DESCRIBE EL DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA MODULAR PARA REALIZAR PRÁCTICAS DE SERVOMECANISMOS Y CONTROL. El objetivo del mismo es que tener mayor facilidad de uso de los para los servomotores GSK 110SJT, se han desarrollado cuatro prácticas en las cuales los estudiantes podrán familiarizarse con el sistema y con el control que se puede ejercer sobre la misma.

El recorrido del deslizador es de 80 centímetros dentro de los cuales se ha destinado dos sensores ópticos a cada extremo del deslizador los cuales sirven de advertencia y protección ante eventuales colisiones que puedan sucederse por una mala operación del equipo.

El sistema electrónico permite el control del servodrive a través de una comunicación serial RS-232, o mediante una red de datos TCP/IP. Estas dos formas de comunicación brindan gran flexibilidad en el manejo del dispositivo.

El sistema se subdivide en tres partes claramente diferenciadas, el subsistema mecánico, el subsistema electrónico y el subsistema informático. En la base del subsistema mecánico se encuentra el tornillo de bolas recirculantes, el cual junto a las guías y rodamientos permiten el

movimiento lineal del deslizador. El tornillo utilizado permite tener una precisión de 52um, el servomotor utilizado cuenta con 10 000 pasos por revolución, lo que se traduce a una precisión de 0.5um, la cual se ve limitada por la precisión del tornillo.

El segundo subsistema, el electrónico, cuenta como elemento principal el microcontrolador ARM mbed® el cual actúa como puente entre el subsistema informático y el subsistema mecánico, este elemento permite la ejecución de comandos enviados a través de sus interfaces seriales, o de su interfaz ethernet, permitiendo el control del deslizador de una manera sencilla.

El subsistema informático es aquel que se encuentra expuesto directamente al usuario, se ha creado para ello una interfaz para su programación (API), mediante la cual se puede interactuar con el servomotor y consecuentemente con el deslizador, esta API se ha desarrollado sobre el lenguaje de programación de LabVIEW, sin embargo debido a la característica de comunicación serial del dispositivo estas librerías pueden ser exportadas a otros lenguajes de programación.

II. DESARROLLO DE CONTENIDOS

A. Diseño del sistema

La transmisión de potencia desde el servomotor hacia el carro deslizador se lo realiza a través del tornillo central, el cual se ha determinado como un tornillo de bolas recirculantes el cual permitirá contar con la precisión necesaria para la implementación de este trabajo, además de minimizar el juego existente entre la tuerca y el tornillo.

Para facilitar el movimiento del carro deslizador se ha escogido rodamientos lineales, los cuales proveen de un bajo coeficiente de fricción además de soportar la fuerza ejercida por la carga. Estos rodamientos requieren de guías de acero rectificadas las cuales brindarán una larga vida útil a los rodamientos.

La estructura para soportar el mecanismo se ha elaborado en acero ASTM A-500, permitiendo soportar todas las fuerzas generadas y brindando la rigidez necesaria para el buen desempeño del deslizador para la carga propuesta la cual es de 30 Kg.

B. Características mecánicas.

El rodamiento lineal escogido es el SKF LBBR 20, el cual

soporta una carga estática de 800 Newtons. El tiempo de vida útil calculado para una carga de 588 Newtons es de 176 Km de recorrido de acuerdo a la siguiente ecuación:

$$L_{10} = \frac{l_s}{l_t} \frac{C}{P}^p$$

$$L_{10} = \frac{0.7}{1} \frac{800}{588}^3$$

$$L_{10} = 1.76 \cdot 10^5 (m)$$

Se ha utilizado dos rodamientos por guía, con un total de cuatro rodamientos. El tornillo tiene una longitud total de 1000 mm, un diámetro externo de 16 mm y una longitud de paso de 5mm, se cuenta con una eficiencia del 90% y un coeficiente de fricción de 0.05% con lo cual cumple con los requisitos necesarios para esta aplicación.

Para calcular la inercia del eje contamos con la siguiente fórmula:

$$J_{eje} = 7.57 \times 10^{-13} D^4 L \text{ kg} - m^2$$

Dónde.

J = Inercia en el eje

D = Diámetro del eje

L = Longitud del eje

Utilizando los valores de la hoja de datos del tornillo obtenemos:

$$J_{eje} = 7.57 \times 10^{-13} 16^4 1000 \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 4.96 \times 10^{-5} \text{ (kg.m}^2\text{)}$$

La inercia de la carga es la siguiente:

$$J_{eje} = 2.25 \times 10^{-8} \frac{m_{carga}}{P^2} \text{ kg} - m^2$$

Dónde.

J_{carga} = Inercia de la carga reflejada en kg.m²

m_{carga} = Carga útil en kg a:

P = paso del eje expresado en rev/mm

Utilizando los valores de este proyecto tenemos:

$$J_{eje} = 2.25 \times 10^{-8} \frac{30}{0.2^2} \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 1.6875 \times 10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 7.57 \times 10^{-13} 16^4 1000 \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 4.96 \times 10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{TOTAL} = 1.6875 + 4.96 \times 10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{TOTAL} = 6,6475 \text{ (kg.m}^2\text{)}$$

El torque para la aceleración, tomando en cuenta la carga

utilizada, la inercia en el motor y la eficiencia obtenemos el siguiente resultado.

$$T_{\alpha} = J_{motor} + \frac{J_{eje}}{e} + \frac{J_{carga}}{e} \alpha \text{ N.m}$$

Dónde:

e = Eficiencia

J_{motor} = Inercia del motor (típico 0.5)

$$T_{\alpha} = 0.5 + \frac{6.6475}{0.9} \times 10^{-5} 11310 \text{ (N.m)}$$

$$T_{\alpha} = 0,89 \text{ (N.m)}$$

El torque generado por el servomotor es de dos Newton por metro, con lo cual podrá funcionar sin ningún inconveniente.

C. Características electrónicas.

El subsistema electrónico tiene como base al microcontrolador LPC1768 el cual implementa la plataforma de desarrollo ARM mbed. Este microcontrolador interactúa con el servodrives mediante señales eléctricas, el servodrives es el elemento que controla al servomotor. El microcontrolador está conectado al encoder y a los sensores de proximidad del deslizador.

Los sensores de proximidad están colocados de tal manera que el carro deslizador no golpee los extremos del deslizador y evite que sea dañado por una mala manipulación. Se ha implementado un sensor de advertencia y uno de error para cada extremo del deslizador, esto permite al programador tomar acciones correctivas y de no hacerlo el sensor de error actúa y detiene al servomotor.

El encoder implementado es un encoder absoluto de cuadratura HOHNER, el cual cuenta con una resolución de 360 pulsos por revolución, al ser conectado al microcontrolador puede multiplicar la resolución por cuatro, esto se puede lograr debido a los dos canales, A y B del encoder dando una resolución de 1440 pulsos por revolución, lo cual se traduce a una precisión de 3.472 micrones.

El subsistema electrónico tiene dos fuentes de energía una de 24 VDC y otra de 5 VDC, la fuente de 24 VDC se utiliza para la alimentación del encoder y la generación de una señal analógica de voltaje para el control de velocidad del servomotor, la fuente de 5 VDC es utilizada para la alimentar el microcontrolador.

El microcontrolador implementa para la comunicación dos puertos seriales, uno a través de USB y otro a través de un conector DB-9, también se implementa un puerto de comunicaciones ethernet, con el cual se puede conectar el microcontrolador a una red de datos cableada.

D. Características de software

Las opciones de comunicación implementadas por el microcontrolador simplifican en gran medida el control del servomotor, tomando ventaja de este hecho se ha diseñado un listado de comandos que el microcontrolador puede ejecutar para el control del servomotor.

TABLA I
LISTADO DE COMANDOS PARA LA COMUNICACIÓN CON EL MICROCONTROLADOR ARM MBED

| Comando | Valores aceptables | Detalle |
|---------|--------------------|--|
| S | 0 a 1 | Apagar / Encender el servomotor |
| H | 0 a 500000 | Frecuencia de salida del tren de pulsos, en hertzios. |
| K | 0 a 500 | Frecuencia de salida del tren de pulsos, en kilohertzios. |
| D | 0 a 1 | Sentido de giro: (CW = 1 – Hacia el motor). (CCW = 0 – Hacia el encoder). |
| A | -10000 a +10000 | Voltaje de salida de +-10V con una resolución de -4.8mV. |
| P | 0 a 2 147 483 647 | Número predefinido de pulsos |
| M | 0 a 2 147 483 647 | Generar un número predefinido de pulsos, multiplica el valor del comando por mil |
| N | 0 a 2 147 483 647 | Generar un número predefinido de pulsos, multiplica el valor del comando por un millón |
| V | 10 - 500000 | La velocidad en tren de pulsos a la cual se va a llegar al ángulo deseado para el eje del motor. |
| I | 0 a 2 147 483 647 | Lleva el carro al inicio del recorrido. El valor numérico determina la frecuencia del tren de pulsos |
| E | 0 a 1 | 0 Leer posición actual. 1 Leer velocidad actual. |
| Z | 0 | Limpia el contador del encoder |

El microcontrolador ejecutará estos comandos siempre y cuando lleguen en el formato correcto.

[int32]-[Comando][Enter]

[int32] = Cualquier número entero de 32bits.
 [Comando] = Letra mayúscula que designa una acción a realizar.
 [Enter] = Carácter Retorno de carro para enviar el comando.

Ejemplo de comandos aceptables:
 1500-H
 -125000000-P
 0-E
 -1-R

Se ha implementado una librería en el lenguaje de programación LabVIEW que permite a los estudiantes la interacción con el servomotor, esta librería actúa como una Interfaz de programación sobre la cual los estudiantes pueden desarrollar sus proyectos.

Esta librería se ha desarrollado en dos versiones, una para la comunicación serial y una para la comunicación ethernet, lo que permite esta librería es tener bloques de programación sencillos con los cuales se puede elaborar programas más complejos.

III. RESULTADOS OBTENIDOS

Mediante la aplicación de todo lo hasta aquí expuesto se tiene el resultado final el cual se muestra a continuación.

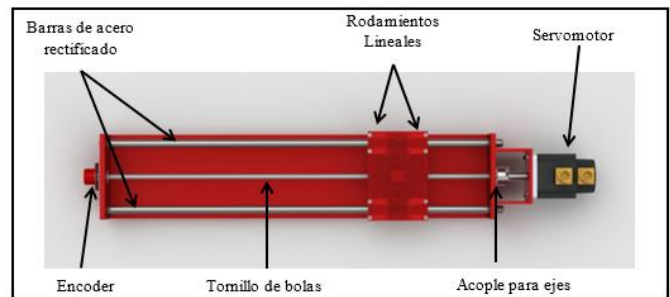


Fig. 1. Subsistema Mecánico.

El subsistema mecánico ensamblado, el servomotor es sujetado a la estructura del deslizador y unido al tornillo central mediante el uso de un matrimonio mecánico, de esta manera el tornillo central girará conjuntamente con el eje del servomotor. El carro deslizador se encuentra sujeto a la nuez del tornillo transmitiendo el torque rotacional del servomotor en movimiento, el peso que se aplica sobre el carro deslizador mismo es soportado por las guías laterales sobre las cuales se montan los rodamientos lineales facilitando su movimiento. Al final del tornillo de bolas se ubica el encoder, el cual gira junto con el eje del tornillo y del servomotor sirviendo de esta manera como sensor de retroalimentación al sistema.

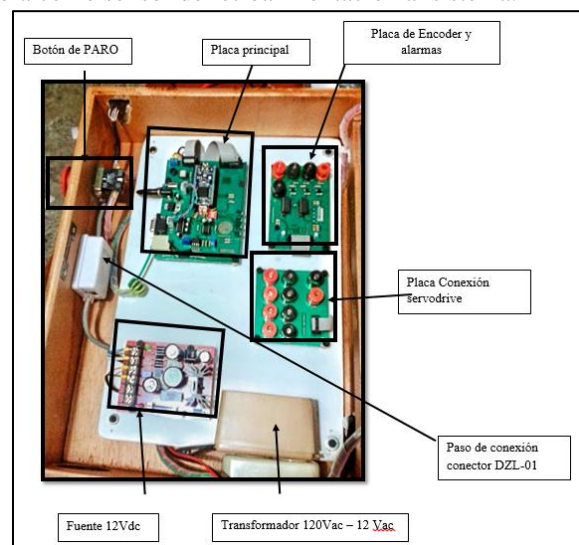


Fig. 2. Subsistema Electrónico

La implementación del subsistema electrónico cuenta con tres placas electrónicas, cada una desempeñando una función específica. La primera es la placa de conexión con el servodrive, la cual sirve para realizar las conexiones eléctricas con el servodrive. La segunda es la placa de encoder y alarmas, esta tiene la función de conectar el encoder y los sensores de proximidad ópticos, adapta su voltaje e implementa un circuito de protección independiente al microcontrolador principal para los casos en los que el carro del deslizador se acerque demasiado a los extremos del servomotor. Finalmente tenemos nuestra placa principal que es aquella en la que está montado el microcontrolador, en esta se tiene los puertos de comunicación con los dispositivos exteriores y en la cual se configura el modo de operación del subsistema.

El subsistema cuenta con dos modos de comunicación, serial y ethernet, sin embargo el microcontrolador solo puede trabajar con uno de ellos a la vez, por tal motivo es necesario configurar el puerto con el cual se va a trabajar. Para el modo de comunicación serial tenemos dos puertos, USB y conector DB-9, el puerto USB puede ser conectado directamente a un computador, sin embargo la longitud del cable queda restringida al estándar USB, para superar este problema se encuentra implementado el puerto DB-9, el cual cumple con los estándares RS-232 y puede ser extendido hasta una longitud de varios metros, dependiendo de la protección del cable utilizado. En el modo de comunicación ethernet se tiene un conector RJ-45 el cual permite al dispositivo ser conectado a un ruteador con el protocolo DHCP habilitado para que de esta manera el dispositivo pueda obtener una dirección IP válida y se pueda realizar la comunicación con el microcontrolador.

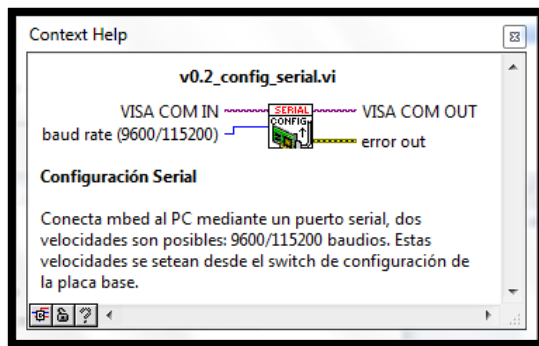


Fig. 3. Subsistema de Control

Finalmente tenemos el subsistema de control, implementado a través de la API creada para este efecto, la misma que ha sido desarrollada para el lenguaje LabVIEW. Esta librería permite controlar todos los aspectos fundamentales del dispositivo y ha sido escrita en dos versiones, una para cada modo de comunicación, tenemos la librería de comunicación serial y la librería de comunicación ethernet. Ambas librerías son simétricas, es decir que por cada función implementada en la librería serial se tiene su

contraparte en la librería ethernet.

Además del control básico de los comandos enviados al servomotor estas librerías poseen funciones de reconocimiento de errores y generación de alarmas, lo cual facilita al estudiante el tomar acciones correctivas de acuerdo a los eventos generados.

Una vez que sistema modular se encuentra operativo se realizó algunas pruebas para comprobar el correcto funcionamiento de todos los subsistemas, encontrándose los siguientes resultados.

TABLA II
PRUEBAS DE POSICIÓN

| N° Prueba | Carga (Kg) | Error medido (mm) |
|-----------|------------|-------------------|
| 1 | 0 | > 0.01 |
| 2 | 5 | > 0.01 |
| 3 | 10 | > 0.01 |
| 4 | 15 | > 0.01 |
| 5 | 20 | > 0.01 |
| 6 | 25 | > 0.01 |
| 7 | 30 | > 0.01 |

El servomotor cuenta con dos modos de funcionamiento, uno de control de posición el cual es muy preciso y un modo de control de velocidad, para aplicaciones en las que se requiere de un torque constante.

En la Tabla 1 se analiza el error medido cuando el deslizador es sometido a diferentes cargas, obteniendo un resultado mejor al requerido por el alcance de nuestra aplicación. Esto es completamente esperado debido a la precisión tanto del servomotor como al juego casi inexistente en el tornillo.

TABLA III
PRUEBAS DE VELOCIDAD

| Voltaje aplicado (Vdc) | Velocidad esperada | Velocidad Medida (RPM) |
|------------------------|--------------------|------------------------|
| 10.00 | 2000 | 2000 |
| 5.00 | 1000 | 1150 |
| 2.00 | 400 | 452 |
| 1.00 | 200 | 218 |
| 0.00 | 0.00 | 0 |
| -1.00 | -200 | -217 |
| -2.00 | -400 | -450 |
| -5.00 | -1000 | -1150 |
| -10.00 | -2000 | -2000 |

En el modo de control de velocidad el servomotor se controla por medio de una salida de voltaje, para las mediciones se ocupó un multímetro FLUKE 83V calibrado con una precisión de cuatro cifras significativas, la medición de la velocidad fue tomada mediante el sensor del servodrive y se desconectó el carro deslizador del tornillo para tomar varias mediciones sin que el carro golpee el final del deslizador. En este caso se puede apreciar que la velocidad medida no corresponde exactamente con el valor esperado, esto puede deberse a que el sensor de voltaje en el servodrive se encuentra descalibrado. Este error inherente en el servomotor abre la posibilidad de implementar métodos de control que permitan estudiar, minimizar y eliminar el error con el control de velocidad del servomotor.

IV. PRACTICAS

El objetivo de este proyecto es la utilización del sistema modular por parte de los estudiantes, por tal motivo se ha desarrollado cuatro prácticas con las cuales se espera que el estudiante se familiarice con el manejo y programación del módulo. Estas prácticas exponen al estudiante con todas las posibilidades brindadas por el sistema y alientan a su modificación y observación de resultados. Todas las prácticas se ofrecen a través del modo de comunicación serial.

A. Práctica 01. Generación de frecuencia

En la primera práctica se presenta al estudiante con el control del servomotor en su modo de posición. En este modo de control el subsistema electrónico genera un tren de pulsos, la frecuencia de este tren de pulsos determina la velocidad del servomotor, la dirección del giro es controlada por una salida digital.

El estudiante es presentado con una interfaz de fácil manejo en la cual controlan el encendido y apagado del servomotor, la frecuencia del tren de pulsos generados, y la dirección de giro del servomotor.

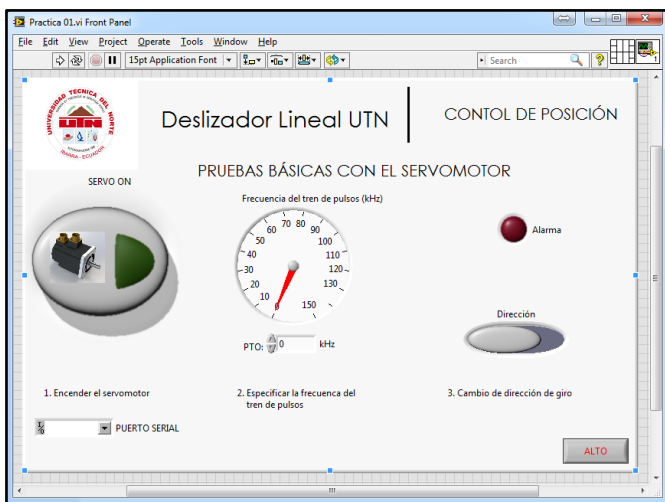


Fig. 4. Panel Frontal – Control de Posición.

Para el correcto funcionamiento de la práctica se requiere

que el estudiante configure de manera adecuada los parámetros del servomotor, la comunicación serial y las conexiones eléctricas del servodrive, para completar estos pasos es necesario que el estudiante conozca estos aspectos básicos del sistema con lo cual se tiene una introducción a los conceptos básicos del manejo del servomotor y del sistema que se ha diseñado para su control.

B. Práctica 02. Generación de voltaje

Esta práctica es similar a la primera práctica, sin embargo el modo de funcionamiento del servomotor es control de velocidad, en esta práctica el subsistema electrónico genera una señal de voltaje variable que controla la velocidad rotacional del servomotor.



Fig. 5. Panel Frontal – Control de velocidad.

Este tipo de control se utiliza para alcanzar altas velocidades con el servomotor, el voltaje generado puede llegar a tener un valor máximo de +/- 10 Vdc. En esta práctica se tiene un control de lazo abierto, se tienen un deslizador para controlar el voltaje de salida y un tacómetro el cual indica la velocidad medida por el encoder en revoluciones por minuto.

C. Práctica 03. Control de Posición

La práctica número tres es una práctica en la cual se busca posicionar el carro deslizador de una manera precisa y dinámica en un punto determinado del recorrido deslizador. Para ello se hace uso del generador de frecuencia del microcontrolador para generar un número determinado de pulsos y de esta manera desplazar el carro una distancia definida.

El mecanismo utilizado para la generación de pulsos es mediante un cronómetro el cual lleva la cuenta del tiempo que se han generado pulsos y calcula la posición en la cual se encuentra el carro. Este es un proceso de lazo abierto, el programa en LabVIEW asume que el carro se ha desplazado una determinada distancia dependiendo del tiempo que ha transcurrido en su cronómetro interno.



Fig. 6. Panel Frontal – Control de Posición.

En esta practica el control de posición es dinámico, es decir que la posición final del carro puede cambiar de manera aleatoria, esto se ha tomado en cuenta en la programación de la practica, cuando la posición de llegada del carro cambia el programa calcula cuanto tiempo ha transcurrido desde la última vez que se llegó a la posición final y dependiendo de la frecuencia del tren de pulsos calcula la posición actual y cuanto tiempo debe transcurrir hasta llegar a la nueva posición final. Mediante la aplicación de este algoritmo se incurre en un error cada vez que se calcula una nueva posición, ya que el algoritmo no toma en cuenta el tiempo transcurrido mientras se realiza el cálculo. Este error, aunque es mínimo debido a la velocidad de procesamiento de los computadores actuales, es acumulativo y cuando se generan altas frecuencias puede llegar a afectar notablemente la precisión del deslizador.

Existen varios métodos para corregir este error, y se alienta al estudiante a explorar las opciones y experimentar con los resultados obtenidos.

D. Práctica 04 – Control de velocidad

La última práctica tiene como objetivo demostrar las capacidades de control en lazo cerrado del sistema, utilizando como sensor de retroalimentación al encoder. Este se utiliza para medir la velocidad rotacional del servomotor y realizar los ajustes necesarios para obtener la velocidad seteada.

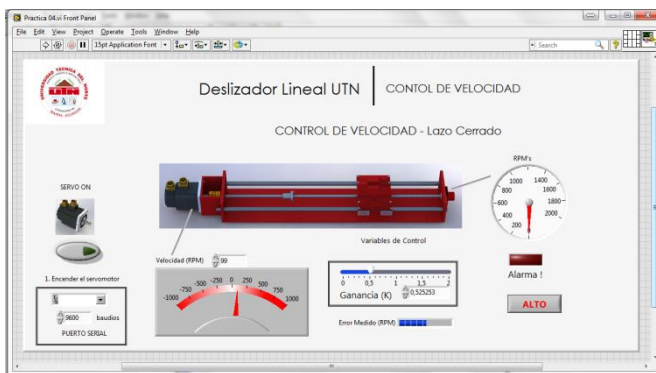


Fig. 7. Panel Frontal – Control de Velocidad

Para el lazo de cerrado se utiliza un control proporcional el cual puede ser modificado durante la ejecución del programa permitiendo experimentar con este tipo de control y observar de manera experimental los efectos que tiene este control sobre la salida de nuestro sistema.

V. CONCLUSIONES

- Se ha determinado que la velocidad máxima soportable por el mecanismo es de 1500 RPM, una velocidad superior por un periodo prolongado de tiempo puede provocar daños a los rodamientos del deslizador.
- Se diseñó un sistema que puede ser usado por una amplia gama de estudiantes, no es estrictamente necesario tener conocimientos avanzados en servomecanismos.
- Las librerías creadas en el entorno de desarrollo permiten una programación del deslizador que se acople a las diferentes prácticas del deslizador.
- Se logró un diseño robusto adecuado para soportar posibles abusos por parte de los estudiantes en sus experimentos.
- El módulo se implementó con medidas de seguridad que permiten la colisión de la mesa de trabajo con los extremos del deslizador. Además de un botón de paro de emergencia en caso de ser necesario.

VI. SUGERENCIAS

- Se deben observar todas las precauciones de seguridad al momento de operar el módulo para asegurar una larga vida útil del mismo.
- Se sugiere la implementación de puesta tierra para todos los equipos del laboratorio de electrónica y mecatrónica.
- La programación tanto de la librería de funciones en LabVIEW como la del microcontrolador son de código abierto, se sugiere su estudio y mejoramiento de los algoritmos implementados en el módulo.
- El módulo puede ser controlado desde Matlab, se sugiere la implementación de lazos de control en este programa como un ejemplo práctico de sus capacidades de programación.
- La mesa del deslizador tiene la capacidad de atornillar diferentes tipos de mecanismos adicionales, se sugiere el montaje de un brazo robótico que permita la manipulación de objetos.

REFERENCIAS

- [1] Beer, F., & Johnston, R. (2010). MECÁNICA VECTORIAL PARA INGENIEROS. México: McGraw Hill.
- [2] Sharp, R. (2008). Principles of Protocol Design. Kongens Lyngby: Springer.
- [3] Blanding, D. L. (1995). Machine Design Using Kinematics Processing. New York: The American Society of Mechanical Engineers.
- [4] NSK Corporation. (2005). What is a ball screw? Franklin.

- [5] Solum, A. H. (1992). Precision machine design. New Jersey: Prentice-Hall, Inc.
- [6] GSK CNC Equipment Co. (2007). DA98D Digital AC Servo Drive Unit. Guanzhou.
- [7] Hihglights. (2011). CONVERSIÓN DE SEÑAL DE ENCODER DE CUADRATURA . Cuenca, Ecuador.
- [8] León Aguirre, A., & Tapia Vaca, J. (2009). Sincronismo y supervisión de posición y velocidad de un motor trifásico asíncrono con un sistema de encoders maestro-esclavo para aplicaciones de control de ejes. Latacunga: Escuela Politécnica del Ejército.
- [9] SKF Group. (20 de Enero de 2014). www.skf.com. Obtenido de http://www.skf.com/binary/12-55835/SKF4182_EN_Linear-ball-bearings.pdf
- [10] THK. (2010). Guide Ball Bushing / Linear Bushing. Agoura Hills: THK.
- [11] Hohner Corporation. (20 de Enero de 2014). www.encoderonline.com. Obtenido de <http://www.encoderonline.com/Devicenet/Data-Sheets/Incremental/Data-33.htm>
- [12] Márquez Días, J., Pardo Sánchez, K., & Pizarro Valencia, S. (10 de Enero de 2014). <http://rcientificas.uninorte.edu.co/>. Obtenido de <http://rcientificas.uninorte.edu.co/index.php/ingenieria/article/view/2272/1484>
- [13] Microsoft. (24 de 11 de 2013). <http://technet.microsoft.com/>. Obtenido de [http://technet.microsoft.com/en-us/library/cc759499\(v=ws.10\)](http://technet.microsoft.com/en-us/library/cc759499(v=ws.10))
- [14] Zuñiga Tufiño, M. A. (2008). ESTUDIO DISEÑO Y SIMULACIÓN DE UN SISTEMA DE. Quito: ESCUELA POLITECNICA NACIONAL.
- [15] Aerotech. (20 de Enero de 2014). www.aerotech.com. Obtenido de http://www.aerotech.com/media/247221/section16_motor%20selection%20and%20sizing.pdf

Autor

Ernesto Palacios Merino, estudiante de mecatrónica en la Universidad Técnica del Norte, oriundo de la ciudad de Loja. Actualmente se encuentra trabajando en la empresa Kradac, en el área de robótica.



UNIVERSIDAD TÉCNICA DEL NORTE

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN MECATRÓNICA**

SCIENTIFIC ARTICLE

TOPIC:

**MODULAR SYSTEM FOR CONTROL AND SERVOMECHANISMS
PRACTICES USING A LINEAR SLIDER WITH A LABVIEW
INTERFACE**

AUTHOR: Ernesto Vladimir Palacios Merino

DIRECTOR: Ing. Diego Terán

**Ibarra - Ecuador
2014**

MODULAR SYSTEM FOR CONTROL AND SERVOMECHANISMS PRACTICES USING A LINEAR SLIDER WITH A LABVIEW INTERFACE

Palacios Merino, Ernesto Vladimir
mecatronica.mid@gmail.com
Universidad Técnica del Norte

Abstract— This paper describes the design and implementation of a linear slider; it has been design for use with GSK 110SJT servomotor. To implement the control for this mechanism have been established libraries and interfaces in the graphical programming language LabVIEW, these libraries have been produced with the intention to provide students of Mechatronics in the "Universidad Técnica del Norte" usability and scalability in their projects, by encouraging the creation of programs and elaborated mechanisms. Once the mechanical, electrical and electronics parts have been completed the students will have a system in which all these modules and subsystems are interrelated through a software library and defined communication protocols, and can thus carry out practices and servo control..

Glossary — *LabVIEW (Graphic visual development platform). API (Application Programming Interface)*

VII. INTRODUCTION

THIS PAPER DESCRIBES THE DESIGN AND IMPLEMENTATION OF A MODULAR SYSTEM FOR SERVO AND CONTROL PRACTICES. The objective is to have greater ease of use for GSK 110SJT servo motor, four practices have been developed in which students can become familiar with the system and the control that can be exerted on it.

The slider travel is 80 centimeters, inside of which there has been allocated two optical sensors at each end of the slider which serve as warning and protection against possible collisions that could be caused by poor equipment operation.

The electronic system allows control over the servo drive via a serial RS-232, or via a TCP / IP connection. These two forms of communication offer great flexibility for handling the device.

The system is divided into three distinct parts, the mechanical subsystem, the electronic subsystem and the computer subsystem. On the basis of the mechanical subsystem is a ballscrew, which together with the linear guides and bearings allow linear movement of the slider. The ballscrew used has an accuracy of 52um, the servo used has 10,000 steps per revolution, which translates to a precision of 0.5um, this precision is limited by the accuracy of the ballscrew.

The second subsystem, has as its main element an ARM mbed® microcontroller which acts as a bridge between the computer subsystem and the mechanical subsystem, this element allows the execution of commands via their serial interfaces, or the ethernet interface, allowing control of the slider in a simple manner.

The computer subsystem is the one that is exposed directly to the user, for this an application programming interface (API) was created, through which you can interact with the servo and consequently with the slider, this API was developed on the LabVIEW programming language, however due to the serial communication characteristic of the device such libraries can be exported to other programming languages.

VIII. CONTENT DEVELOPMENT

A. System Design

The transmittion of power from the servomotor to the slider is performed through the central ballscrew, which has been determined as a ballscrew with re-circulating balls which will allow for the precision necessary in the implementation of this work, in addition to minimize the clearance between the nut and screw, referred as backlash.

To facilitate movement of the slider carriage linear bearings have been chosen, which provide a low friction coefficient in addition to support for the force exerted by the load. These bearings require rectified steel guides which provide long life to the bearings.

The structure for supporting the mechanism is made of steel ASTM A-500, allowing withstand all the forces generated and providing the necessary rigidity for the proper performance of the slider to the weight proposed which is 30 Kg.

B. Características mecánicas.

The linear Bearing chosen is SKF LBBR 20, which supports a static load of 800 Newtons. The estimated lifetime for a load of 588 Newtons is 176 km in length according to the following equation:

$$L_{10} = \frac{l_s}{l_t} \frac{C}{P}^p$$

$$L_{10} = \frac{0.7}{1} \frac{800}{588}^3$$

$$L_{10} = 1.76 \cdot 10^5 (m)$$

We used two bearings for each guide, with a total of four bearings. The screw has an overall length of 1000 mm, an outer diameter of 16 mm and a pitch length of 5 mm, the efficiency is 90% and a coefficient of friction of 0.05% thus meets the requirements for this application. To calculate the inertia on the shaft following formula has been used:

$$J_{eje} = 7.57x10^{-13} D^4 L \text{ kg} - m^2$$

Where,

J = Axis Inertia

D = Axis diameter

L = Axis length

Using the values provided by the datasheet we get:

$$J_{eje} = 7.57x10^{-13} 16^4 1000 \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 4.96x10^{-5} \text{ (kg.m}^2\text{)}$$

The inertia on the load is:

$$J_{eje} = 2.25x10^{-8} \frac{m_{carga}}{P^2} \text{ kg} - m^2$$

Where.

J_{carga} = Inertia on the load in kg.m²

m_{carga} = Useful load in kg a:

P = ballscrew step in rev/mm

Using the data of the Project we get:

$$J_{eje} = 2.25x10^{-8} \frac{30}{0.2^2} \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 1.6875 x 10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 7.57x10^{-13} 16^4 1000 \text{ (kg.m}^2\text{)}$$

$$J_{eje} = 4.96x10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{TOTAL} = 1.6875 + 4.96 x 10^{-5} \text{ (kg.m}^2\text{)}$$

$$J_{TOTAL} = 6,6475 \text{ (kg.m}^2\text{)}$$

The initial torque necessary for acceleration, calculated using the load, the servomotor inertia and the efficiency of the system we obtain the following result.

$$T_{\alpha} = J_{motor} + \frac{J_{eje}}{e} + \frac{J_{carga}}{e} \alpha \text{ N.m}$$

Where:

e = Efficiency

J_{motor} = Servomotor Inertia (typical 0.5)

$$T_{\alpha} = 0.5 + \frac{6,6475}{0.9} x 10^{-5} 11310 \text{ (N.m)}$$

$$T_{\alpha} = 0,89 \text{ (N.m)}$$

The torque generated by the servomotor is two Newton by meter, which is superior to the torque needed for the system to work.

C. Electronic characteristics.

The electronic subsystem is based on the LPC1768 microcontroller which implements the ARM mbed development platform. This microcontroller interacts with the servo drive using electrical signals; the servo drive is the element that controls the servomotor. The microcontroller is connected to the encoder and proximity sensors.

Proximity sensors are positioned so that the car does not hit the slider end and prevent it from being damaged by rough handling. We have implemented one warning sensor and one error sensor for each end of the slider; this allows the programmer to take corrective action, if none is taken the error sensor acts and stops the servomotor.

The implemented encoder is a HOHNER absolute quadrature encoder, which has a resolution of 360 pulses per revolution, it's connected to the microcontroller and the resolution can be multiplied by four, this can be achieved due to the two channels, A and B of the encoder giving a resolution of 1440 pulses per resolution, which translates to a precision of 3.472 microns.

The electronic subsystem has two power sources, one of 24 VDC and the other of 5 VDC, 24 VDC supply is used to power up the encoder and generating an analog voltage signal for the speed control of the servo motor, the source of 5 VDC is used to power the microcontroller.

The microcontroller implements two serial ports, one through USB and one through a DB-9 connector, there is another means of communication, an ethernet communications port through which the microcontroller can be connected to a wired network.

D. Software characteristics

Communication options implemented by the microcontroller greatly simplify servo control, taking advantage of this fact a list of commands have been designed so that the microcontroller can run them for servo motor control.

The microcontroller will execute these commands provided that they arrive with the correct format.

This library has been developed in two versions, one for serial communication and one for Ethernet communication, which allows this library to have simple programming blocks which can develop more complex programs.

A LabVIEW programming library has been design that allows students to interact with the servomotor, this library acts as a programming interface on which students can develop their projects.

TABLE I
LIST OF THE COMMANDS FOR THE COMMUNICATION WITH THE ARM MBED
MICROCONTROLLER

| Command | Acceptable Values | Detail |
|---------|-------------------|---|
| S | 0 to 1 | On/Off for the servomotor |
| H | 0 to 500000 | Pulse Train Output in Hertz |
| K | 0 to 500 | Pulse Train Output in Kilo Hertz |
| D | 0 to 1 | Direction: (1 = Clock wise). (0=Counter Clock wise). |
| A | -10000 to +10000 | Output voltage +-10V with a resolution of ~4.8mV. |
| P | 0 to 2147483647 | Generate a defined number of pulses |
| M | 0 to 2147483647 | Generate a defined number of thousands of pulses |
| N | 0 to 2147483647 | Generate a defined number of million of pulses. |
| V | 10 to 500000 | Frequency of the pulse train at for the defined pulses |
| I | 0 to 2147483647 | Takes the carriage to the beginning of the slide, the number defines the frequency of the pulse train. 0 = Reads current position. 1 Reads current speed (RPM). |
| E | 0 to 1 | |
| Z | 0 | Clears encoder counter. |

Format of the commands sent to the microcontroller.

[int32]-[Command][Return]

[int32] = integer number 32bits.
 [Command] = A letter describing the command to be implemented.
 [Return] = Return Carriage (0x13)

Example of valid commands:
 1500-H
 -125000000-P
 0-E
 -1-R

IX. RESULTADOS OBTENIDOS

Once all of this has been applied, the result is as follows.

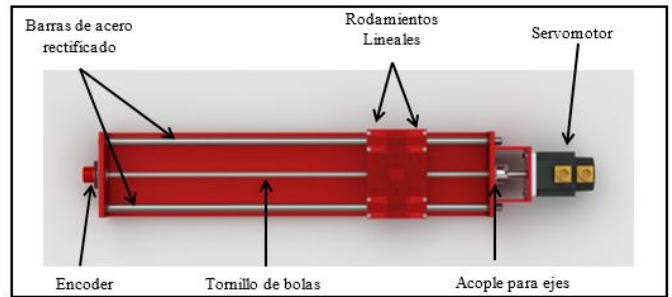


Fig. 1. Mechanical Subsystem.

The mechanical subsystem assembly, the actuator is attached to the sled frame and connected to the ballscrew through the use of a mechanical coupling, so the central ballscrew rotates together with the shaft of the servomotor. The slider carriage is subject to the nut of the screw transmitting rotational torque of the servomotor moving weight that is applied to the same slide carriage is supported by the side rails on which the bearings are mounted facilitating their movement. At the end of the ball screw there is an encoder, which rotates together with the screw shaft of the servomotor and thus serving as sensor feedback system is located.

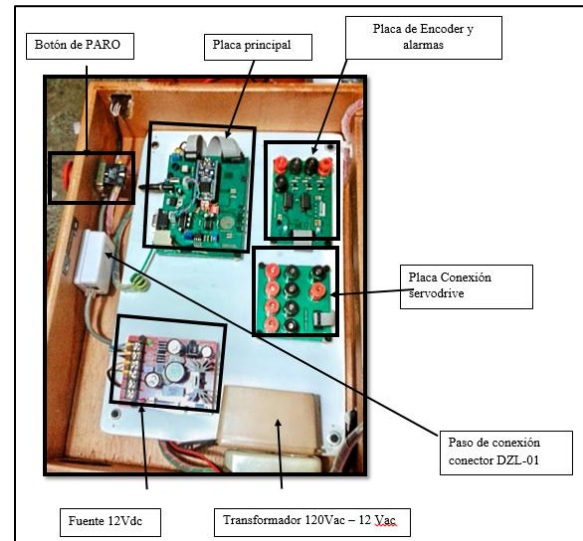


Fig. 2. Electronic subsystem

The implementation of the electronic subsystem has three electronic boards, each performing a specific function. The first board is connected to the servo drive, which is used to make electrical connections to the servo drive. The second is the encoder and alarms board, this serves to connect the encoder and optical proximity sensors, adjusts its voltage and implements an independent circuit protection to the main microcontroller for cases where the slider carriage moves too close to the ends of the actuator. Finally we have our main

board the one in which the microcontroller is located, this board has the ports for communication with external devices and in this board is located a switch to configure the operation mode of the microcontroller.

The subsystem has two communication modes, serial and ethernet, though the microcontroller can only work with one at a time, for this reason it is necessary to configure the port with which it will work. For the serial communication mode there are two ports, a USB and a DB-9 connector, the USB port can be connected directly to a computer, but the cable length is restricted to the USB standard, to overcome this problem a DB-9 port has been implemented, which meets RS-232 standards and can be extended to a length of several meters, depending on the protection of the cable used. In Ethernet mode of communication has an RJ-45 connector which allows the device to be connected to a DHCP enabled router so that the device can obtain a valid IP address and provide communication with the microcontroller.

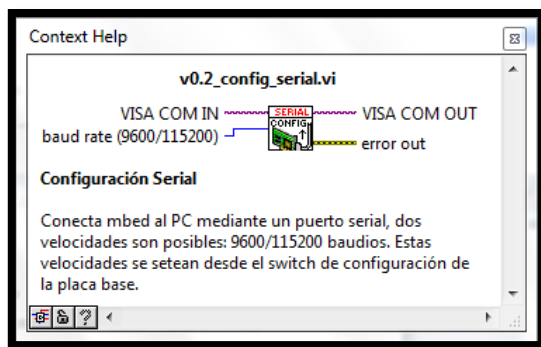


Fig. 3. Subsistema de Control

Finally we have the control subsystem, implemented through the API created for this purpose, the same that has been developed for the LabVIEW language. This library allows control of all key aspects of the device and has been written in two versions, one for each communication mode, the library for serial communication and the library for Ethernet communication. Both libraries are symmetrical, ie that for each function implemented in serial library there is a counterpart in the ethernet library.

Besides the basic control commands sent to the servomotor these libraries have functions for error detection and alarm generation, which helps students taking corrective actions according to events generated.

Once the modular system has been operative some tests were performed to verify the correct operation of all subsystems, and following results were found.

The servomotor has two modes of operation, a position control which is very precise and speed control mode, for applications where it is required that the servomotor maintains a constant torque.

Table 1 shows the measured error analyzed when the slider is subjected to different loads, obtaining a better than required precision by the scope of our application. This is entirely

expected because both precision of the servomotor and the almost nonexistent backlash on the ballscrew.

TABLE II
POSITION TEST

| N° Trial | Load (Kg) | Measured error (mm) |
|----------|-----------|---------------------|
| 1 | 0 | > 0.01 |
| 2 | 5 | > 0.01 |
| 3 | 10 | > 0.01 |
| 4 | 15 | > 0.01 |
| 5 | 20 | > 0.01 |
| 6 | 25 | > 0.01 |
| 7 | 30 | > 0.01 |

TABLE III
SPEED TEST

| Applied voltage (Vdc) | Expected Speed (RPM) | Measured Speed (RPM) |
|-----------------------|----------------------|----------------------|
| 10.00 | 2000 | 2000 |
| 5.00 | 1000 | 1150 |
| 2.00 | 400 | 452 |
| 1.00 | 200 | 218 |
| 0.00 | 0.00 | 0 |
| -1.00 | -200 | -217 |
| -2.00 | -400 | -450 |
| -5.00 | -1000 | -1150 |
| -10.00 | -2000 | -2000 |

The control mode for servo motor speed is controlled by means of a voltage output. This voltage was measurement using a FLUKE 83V multimeter calibrated to four significant figures addressed, the speed measurement was taken using the sensor of the servo drive and the slider carriage was disconnected from the ballscrew, this is necessary in order to take multiple measurements without the car hitting the end of the slider. In this case it can be seen that the measured speed does not correspond exactly to the expected value, this may be because the voltage sensor in the servo drive is un-calibrated. This inherent error in the servomotor opens the possibility of implementing control methods to study its results in the system, to minimize and eliminate the error with the speed control servomotor.

X. PRACTICAS

The objective of this project is the use of the modular system by students, for this reason four practices has been developed which are expected to familiarize students with the operation and programming of the module. These practices expose the student with all the possibilities offered by the system and encourage modification and observation results. All practices are offered through serial communication mode.

A. Practice 01. Frequency generation

In the first practice the student is presented with position control of the servomotor. In this control mode the electronic subsystem generates a pulse train, the frequency of this pulse train determines the motor speed, and direction of rotation is controlled by a digital output.

The student is presented with a user-friendly interface in which control the switching on and off of the servomotor, the frequency of the pulse train generated, and the direction of rotation of the servomotor.

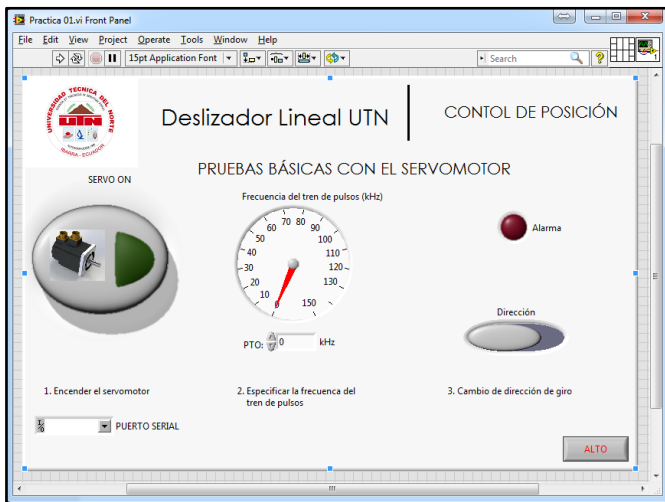


Fig. 4. Panel Frontal – Control de Posición.

For proper operation of the practice it is required the student properly configure the parameters of the servo motor, serial communication and servo electrical connections, to complete these steps is necessary for the student to know these basic aspects of the system thereby is an introduction to the basics of servo motor management.

B. Practice 02. Voltage generation

This practice is similar to the first one, however the operation mode is speed control, in this practice the electronic subsystem generates a variable voltage signal that controls the rotational speed of the servomotor.



Fig. 5. Frontal Panel – Speed Control

This type of control is used to achieve high-speed, the voltage generated can have a maximum of +/- 10 Vdc. This practice has an open-loop control, it has a slider to control the output voltage and a tachometer which indicates the speed measured by the encoder in RPM.

C. Practice 03. Position Control

The practice number three is a practice which seeks to position the slider carriage accurately and dynamically at a given point of the slider travel. This makes use of the microcontroller frequency generator, it is used to generate a number of pulses and thus move the carriage a defined distance.

The mechanism for generating pulses is by a timer which keeps track of time pulses that have been generated and calculates the position in which the carriage is. This is an open-loop process, the LabVIEW program assumes that the car has traveled a certain distance depending on the time that has elapsed in its internal timer.



Fig. 6. Frontal Panel – Position Control.

In this practice the position control is dynamic, meaning that the final position can change at random, this was taken into account when programming the practice, when the arrival position of the carriage changes the program calculates how long it has elapsed since the last time they reached the final position and depending on the frequency of the pulse train calculates the current position and the time span until the new end position. By applying this algorithm incurs an error every time a new position is calculated, since the algorithm does not take into account the time elapsed while the calculation is in

process. This error, although it is minimal due to the processing speed of today's computers, is cumulative and when high frequencies are generated, will notably affect the accuracy of the slider.

There are several methods to correct this error, and the student is encouraged to explore the options and experiment with the results.

D. Practice 04 – Speed control

The latter practice is intended to demonstrate the capabilities of closed loop control in the system, using as feedback the encoder sensor. This is used to measure the rotational speed of the servomotor and make the necessary adjustments to get the set speed.



Fig. 7. Frontal Panel – Speed Control

For the closed loop proportional control has been established which can be modified during program execution allowing to experiment with this type of control and experimentally observe the effects of this control over the output of our system is used.

XI. CONCLUSIONS

- It has been determined that the maximum speed under load is 1500 RPM, a higher speed for a prolonged period of time can cause damage to the bearings of the slider.
- A system has been design that can be used by a wide range of students, it is not strictly necessary to have advanced knowledge in servomechanisms.
- Libraries were created in the development environment that allows an easy programming of the slider so that it adjust to the different practices of the slider.
- A robust design was achieved suitable to withstand potential abuse by students in their experiments.
- The module has implemented security measures that prevent the collision of the work table with the ends of the slider. In addition to a emergency stop button if needed.

XII. SUGERENCIAS

- Observe all safety precautions when operating the module to ensure long service life.
- Implementation of grounding for all lab computers electronics and mechatronics is suggested.
- Programming of both the library functions in LabVIEW as the microcontroller firmware are open source, study and improvement of algorithms implemented in the module is suggested.
- The module can be controlled from Matlab, implement control loops in this program is suggested as a practical example of its programmability.
- The slide table is capable of driving different types of additional mechanisms, the assembly of a robotic arm that allows the manipulation of objects is suggested.

REFERENCES

[16] Beer, F., & Johnston, R. (2010). MECÁNICA VECTORIAL PARA INGENIEROS. México: McGraw Hill.

[17] Sharp, R. (2008). Principles of Protocol Design. Kongens Lyngby: Springer.

[18] Blanding, D. L. (1995). Machine Design Using Kinematics Processing. New York: The American Society of Mechanical Engineers.

[19] NSK Corporation. (2005). What is a ball screw? Franklin.

[20] Solum, A. H. (1992). Precision machine design. New Jersey: Prentice-Hall, Inc.

[21] GSK CNC Equipment Co. (2007). DA98D Digital AC Servo Drive Unit. Guanzhou.

[22] Hihglights. (2011). CONVERSIÓN DE SEÑAL DE ENCODER DE CUADRATURA . Cuenca, Ecuador.

[23] León Aguirre, A., & Tapia Vaca, J. (2009). Sincronismo y supervisión de posición y velocidad de un motor trifásico asíncrono con un sistema de encoders maestro-esclavo para aplicaciones de control de ejes. Latacunga: Escuela Politécnica del Ejercito.

[24] SKF Group. (20 de Enero de 2014). www.skf.com. Obtenido de http://www.skf.com/binary/12-55835/SKF4182_EN_Linear-ball-bearings.pdf

[25] THK. (2010). Guide Ball Bushing / Linear Bushing. Agoura Hills: THK.

[26] Hohner Corporation. (20 de Enero de 2014). www.encoderonline.com. Obtenido de <http://www.encoderonline.com/Devicenet/Data-Sheets/Incremental/Data-33.htm>

[27] Márquez Díaz, J., Pardo Sánchez, K., & Pizarro Valencia, S. (10 de Enero de 2014). <http://rcientificas.uninorte.edu.co/>. Obtenido de <http://rcientificas.uninorte.edu.co/index.php/ingenieria/articicle/view/2272/1484>

[28] Microsoft. (24 de 11 de 2013). <http://technet.microsoft.com/>. Obtenido de

[http://technet.microsoft.com/en-us/library/cc759499\(v=ws.10\)](http://technet.microsoft.com/en-us/library/cc759499(v=ws.10))

[29] Zuñiga Tufiño, M. A. (2008). ESTUDIO DISEÑO Y SIMULACIÓN DE UN SISTEMA DE. Quito: ESCUELA POLITECNICA NACIONAL.

[30] Aerotech. (20 de Enero de 2014). www.aerotech.com.
Obtenido de
http://www.aerotech.com/media/247221/section16_motor%20selection%20and%20sizing.pdf

Autor

Ernesto Palacios Merino,
Born in Ecuador, in the city of Loja, he is a mechatronics student in the “Universidad Técnica del Norte” university. He is currently employed in Kradac, in the robotics department.