

Implementación en DSPIC de un algoritmo óptimo para la estimación de velocidad de giro

Implementation of an optimal algorithm for the estimation of angular speed in DSPICS

José Danilo Rairán-Antolines

Ph.D. en Ingeniería – Sistemas y computación
Profesor titular Universidad Distrital
Francisco José de caldas UDFJC
Bogotá, Colombia
drairan@udistrital.edu.co

José Miguel Fonseca-Gómez

Ingeniero en Electricidad
Universidad Distrital Francisco José de Caldas UDFJC
Bogotá, Colombia
jmfonseca@gmail.com

Resumen— En este artículo se presenta la implementación de un método de medición de velocidad angular nuevo, cuando la información proviene de un encoder incremental. Parte del método es implementado en un dsPIC, mientras que el resto se programa en Simulink. La ventaja fundamental del método de medición es que produce una única aproximación cuando la velocidad es constante, en contraposición con los métodos convencionales, conocidos como a tiempo fijo y a espacio fijo, los cuales oscilan entre dos valores. Se presenta el control de velocidad de un motor, como ejemplo de aplicación, y se concluye que el método nuevo puede aplicarse para la aproximación de la razón de cambio, incluso cuando los datos no provienen de un encoder.

Palabras clave— dsPIC, encoder, espacio fijo, medición de velocidad, tiempo fijo.

Abstract— In this paper we present the implementation of a new method to measure angular speed when the data come from an incremental encoder. Part of the method is implemented in a dsPIC, whereas the remaining part was programmed in Simulink. The main advantage of the new method consists of having just one approximation at constant speed, whereas traditional methods oscillate between two values. The speed control of a motor serves as an application of the new method, which allows concluding that the proposed method can be used to estimate the rate of change of a variable even when the data do not come from an incremental encoder.

Keywords— dsPIC, Encoder, Fixed space, Fixed time, Speed measured.

1. INTRODUCCIÓN

En la industria se encuentra que los encoders incrementales son la opción más popular para medir la velocidad, dado su costo y su simplicidad. De ahí que cualquier propuesta para mejorar el

sensor tendrá un impacto positivo en la industria. Por tal razón, en este trabajo se implementa un método nuevo para la medición de velocidad, el cual minimiza el error absoluto en la aproximación, como se mostrará adelante.

En la literatura pueden identificarse tres grupos de trabajos alrededor de la propuesta en este artículo. El primero se trata de reducir los errores causados por la construcción misma del sensor; el segundo busca reducir el error en la aproximación, por el tratamiento de las señales, como se hace en este artículo; finalmente, se encuentran aplicaciones nuevas de los algoritmos conocidos. Correspondiente al primer grupo, en [1] se busca eliminar la naturaleza discreta del sensor, por medio de la estimación de la posición angular, usando la superposición de medidas distanciadas cada delta de tiempo. En el grupo dos, en cuanto a la reducción del error; un ejemplo lo constituye la generación de algoritmos para mejorar la calidad de la estimación, como en [2] que se presenta un algoritmo para mejorar el resultado alcanzado con los algoritmos tradicionales, por medio del cambio del tiempo de discretización. Dentro del campo de las aplicaciones está la estimación de la aceleración, y de ahí torque, lo cual puede ser utilizado en el control de motores, como se muestra en [3].

El encoder incremental es un dispositivo electromecánico, el cual permite aproximar la velocidad angular de un eje. La parte mecánica consiste en un disco, ranurado, a espacios que se supondrán iguales. La parte electrónica se encarga de generar una luz, la cual es interpretada por un re-

ceptor, como uno cuando la luz pasa la ranura y llega al receptor y como cero en caso contrario.

La aproximación de la velocidad requiere de dos mediciones: una de posición y una de tiempo. El error en la aproximación tiene dos fuentes: la primera es que en la práctica sólo pueden hacerse medidas a tiempo discreto, y por tanto la velocidad medida, ω_m , es $\Delta\theta / \Delta t$; la segunda, tiene que ver con la medida de esos deltas, lo cual conlleva a otro error. El delta de posición $\Delta\theta$, es un múltiplo entero del número de pulsos del encoder pul ; por tanto, existe un error de truncamiento. De igual manera, el delta de tiempo Δt (el cual se llamará dt en adelante), es un múltiplo entero del reloj del sistema ts , donde se implemente el algoritmo.

En la literatura son dos los algoritmos clásicos para la aproximación de la velocidad; estos se conocen como a tiempo fijo y a espacio fijo [4]. Cuando la medición se hace a tiempo fijo se define un dt y se cuenta el número de pulsos en ese intervalo de tiempo. Si la medida se hace a espacio fijo, entonces se mide el tiempo que hay entre cada par de pulsos [5]. La medición a espacio fijo es recomendada cuando la velocidad es baja, mientras que tiempo fijo es útil cuando la velocidad es alta [6].

Un concepto importante, relacionado con dt y pul es la que se denominará velocidad límite ω_{lim} . Esta es la velocidad a la que el tiempo entre un impulso y el siguiente es exactamente dt , es decir, $\omega_{lim} = (1/pul)/dt$ en rev/s, definiciones similares pueden verse en [7], las cuales son utilizadas para definir lo que denominan la resolución de velocidad. Las velocidades menores a ω_{lim} se denominarán bajas, mientras que las superiores, altas.

2. MEDICIÓN DE VELOCIDAD A TIEMPO FIJO

En la medición de la velocidad angular a tiempo fijo se define un delta de tiempo y se cuentan los pulsos provenientes del encoder durante ese delta, como se explica en [8]. El número de pulsos es asociado con un ángulo recorrido. Este método es tal vez la alternativa más común de aproximación de velocidad por medio de un encoder, como puede verse en [9]. Aun así, el método presenta problemas, como la producción de una estimación de velocidad que oscila, aun cuando la veloci-

dad es constante, debido a los errores propios de los truncamientos que se generan al discretizar el espacio y el tiempo, como se indica en [10].

La aplicación de la medición a tiempo fijo, en este artículo, se hizo por medio del dsPIC dsPIC33FJ128MC802, de Microchip, el cual utiliza la librería *Embedded Target for PIC/dsPIC*, de Matlab [11]. Con el propósito de realizar comparaciones entre métodos, se fija el tiempo de observación dt , en 10 ms, y se fija la precisión del reloj del sistema ts en 0.1 ms. Así, la programación del método se divide en tres partes: 1) se miden intervalos fijos de tiempo dt ; 2) se detecta el flanco de subida de la señal proveniente de uno de los canales del encoder; por último, 3) se cuenta el número de pulsos del encoder cada dt . Con el dato de tiempo dt y el número de pulsos se realiza la aproximación de la velocidad, y ese valor se mantiene hasta que se calcule la aproximación siguiente.

Con el fin de garantizar la lectura de tiempo dt , dado que experimentalmente pudo verse que es el proceso crítico en la aproximación, se utilizaron dos dsPIC. Uno dedicado a generar impulsos cada dt , y otro para realizar el resto de procesos. Los datos de entrada se simularon con un generador de onda cuadrada. Esta señal emula los pulsos provenientes del encoder. La medición se hizo tomando datos por 20 s, a una frecuencia constante, y durante todo el rango de tiempo se realiza la estimación de velocidad. Este proceso se repite a intervalos equivalentes a 0.156 rev/s, desde 0.156 rev/s hasta 1.875 rev/s, como se muestra en la Fig. 1, en la cual la parte superior presenta el resultado del método, mientras la de abajo presenta el error medio evaluado a la largo de 20 s. No se prueban frecuencias superiores, dado que el dsPIC comienza a perder conteos de pulsos del generador, por la cual la aproximación aparece con errores altos.

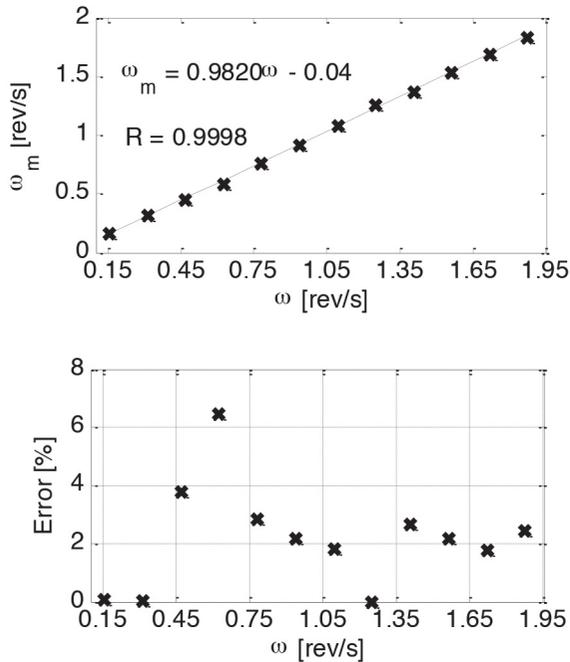
3. MEDICIÓN DE VELOCIDAD A ESPACIO FIJO

El método de aproximación de la velocidad a espacio fijo cuenta el número de intervalos de tiempo dt entre dos pulsos consecutivos del encoder, por lo cual resulta ideal para velocidades bajas, como puede verse en [12].

La implementación de este método requiere al menos de tres procesos. En el primer proceso, se

generan pulsos cada dt ; en el segundo, se generan impulsos asociados con los flancos de subida de la señal proveniente del encoder; finalmente en el tercer proceso, se cuentan los incrementos de tiempo dt y se estima la velocidad. La estimación tiene en cuenta el arco recorrido entre impulso e impulso del encoder, más el conteo de tiempo. Cada aproximación se mantiene hasta tanto aparezca un impulso de posición nuevo.

Fig. 1. RESULTADOS DE LA APROXIMACIÓN DE VELOCIDAD A TIEMPO FIJO



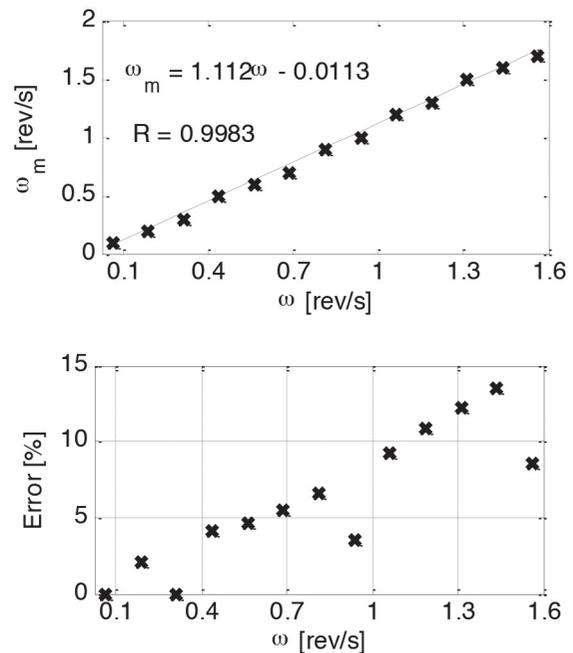
Fuente: autores

Al igual que en la sección anterior, se observó que no es posible correr el método completo en una sola pastilla, porque esto hace que el integrado empiece a perder conteos de tiempo, y como resultado la aproximación aparece con un error alto. En este caso, la solución final consistió en utilizar, no dos, como en la sección anterior, sino tres dsPIC. El primero, genera impulsos cada dt ; el segundo, emula velocidades bajas en el encoder, al tomar sólo un pulso cada k pulsos del generador que emula el encoder; lo cual se hizo con el fin de emular condiciones reales de operación del método; el tercer dsPIC realiza el resto del proceso de aproximación. Los experimentos se corrieron utilizando t_s igual a 1 ms, mientras dt es 10 ms y con k igual a 2.

La prueba experimental del método se hizo comenzando en 0.063 rev/s, a incrementos de

0.125 rev/s, hasta 1.563 rev/s. Como en la sección anterior, la lectura del error en la Fig. 2 corresponde al error medio, tomado a partir de la lectura durante 20 s a velocidad constante. Este error, además, es relativo al valor de velocidad real. En la Fig. 2, donde la parte superior presenta el resultado del método, mientras la de abajo presenta el error relativo, se observa que el error no supera el 15%, el cual se presenta a velocidades altas. La tendencia del error es a aumentar conforme la velocidad aumenta, según se espera teóricamente, como se afirma en [13].

Fig. 2. RESULTADOS DE LA APROXIMACIÓN DE VELOCIDAD A ESPACIO FIJO



Fuente: autores

4. IMPLEMENTACIÓN DEL MÉTODO NUEVO DE ESTIMACIÓN DE VELOCIDAD

En esta sección se presenta la implementación del método de estimación de velocidad angular nuevo. Este método elimina la oscilación en la estimación que se produce con los algoritmos tradicionales, lo cual aún aparece a velocidad constante. La causa de la oscilación es el truncamiento en la medida del tiempo y la posición, como se explica con detalle en [14].

La implementación completa del método implica dos partes. La primera se realiza en el dsPIC, en la cual se trata la señal de pulsos proveniente del encoder. La segunda, toma las señales del ds-

PIC, por medio de una tarjeta de adquisición de datos, y traduce las salidas del dsPIC en una lectura de velocidad, en Simulink. La parte del dsPIC, a su vez, se divide en dos procesos. El primero de estos procesos incluye la generación de pulsos de reloj, cada dt , los cuales son denominados ldt . Por otra parte, está la conversión de la señal del encoder a impulsos, cada k pulsos del encoder, los cuales se denominan lep . El conteo de estos impulsos ldt e lep corresponden con la estimación de tiempo y posición, y su conteo se almacena en las variables Ndt y Nep , respectivamente. La variable Ndt sólo se actualiza cuando hay un impulso de posición; de manera análoga, Nep se actualiza sólo cuando hay un impulso de tiempo.

El segundo proceso dentro del dsPIC consiste en enviar las variables Ndt y Nep por un puerto de salida del dsPIC para que estas variables sean procesadas en un computador. Dado que Nep y Ndt son conteos de impulsos, estas variables son números naturales, lo cual facilita su envío. En resumen el proceso consiste en convertir las variables a binario, para luego enviar el dato en forma serial, como se presenta en la Fig. 3.

El método de estimación de velocidad tiene una particularidad en cuanto al valor de Nep y Ndt : cuando Nep es diferente de uno, Ndt es uno, y viceversa. Esta característica permite que sea suficiente con enviar al computador sólo la variable diferente de uno, lo cual está ejemplificado en la Fig. 3, mediante el bloque denominado selección. A velocidades bajas el tiempo entre un impulso y otro de ldt siempre es menor que el tiempo entre un impulso y otro de lep ; por tanto, Ndt es siempre mayor o igual que uno. De otra parte, cuando la velocidad es alta, el tiempo entre impulso e impulso lep siempre es menor o igual al tiempo entre impulsos ldt , por tanto Nep es siempre mayor o igual que uno.

Ahora bien, una velocidad se dice alta o baja con referencia a la velocidad en la cual el tiempo entre un impulso ldt y el siguiente es igual al tiempo entre un impulso lep y el siguiente, es decir, la velocidad límite ω_{lim} . La velocidad límite en términos de los parámetros físicos del encoder es igual a $(1/pul)/dt$, donde pul indica el número de particiones del encoder, como se explica en [15].

La forma de identificar si el resultado del bloque selección corresponde a Nep o a Ndt es mediante la lectura de una variable denominada ω . Esta variable nueva es uno cuando Nep es mayor que uno, y es cero cuando Ndt es mayor que uno. Así, cuando se envía Nep o Ndt , se envía además la señal ω . Aún hace falta una señal más para enviar Nep o Ndt ; según se indica en la Fig. 4, esta es un impulso, el cual aparece cuando hay un cambio en el valor de Nep o Ndt . La comunicación se realizó de manera que se envía un *bit* proveniente de la conversión del dato a enviar cada ts .

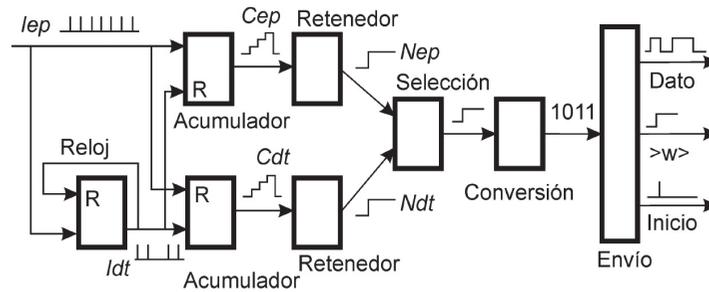
Los datos que envía el dsPIC los recibe Simulink, donde se reconstruye Nep o Ndt y se aproxima la velocidad, según se observa en la Fig. 4, la cual corresponde al diagrama de bloques del programa hecho en Simulink.

Dada la cantidad de código implementada dentro del dsPIC, durante la etapa experimental, se observó que cada delta de tiempo, aproximado de nueve instantes dt , aparece un impulso en el valor del dato enviado, el cual no corresponde al dato que se desea enviar. Esta particularidad obligó al tratamiento de la señal transmitida, de manera que se filtre ese ruido, además de otros ruidos que puedan aparecer, propios de la transmisión de datos en forma serial.

La lógica empleada para limpiar el dato enviado es observar la medida recibida del dsPIC, y comparar ese valor con el promedio de los 15 valores anteriores, distanciados un tiempo dt entre cada uno. Así, si ese cambio es mayor a una cota definida, se deja la lectura anterior, de lo contrario, pasa el entero más cercano al promedio. La cota tiene relación con el cambio de velocidad (aceleración), que puede presentar una máquina giratoria, y en esta aplicación fue de tres unidades en Nep . Para llegar a esta cota se partió de la velocidad nula, y se llevó el motor usado para experimentación a la velocidad máxima, y se observó el cambio máximo en Nep .

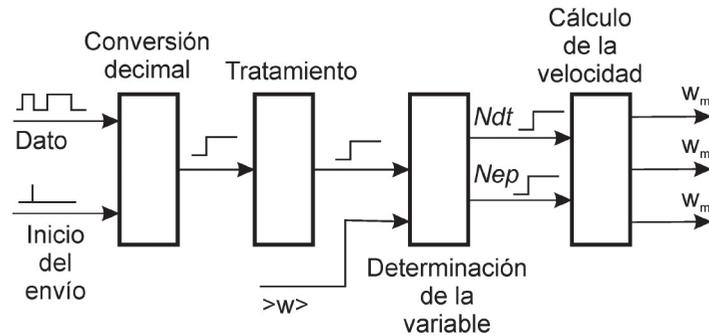
Una vez Nep y Ndt han sido reconstruidos en Simulink, como indica el último bloque en la Fig. 4, se calcula la estimación de la velocidad. La primera estimación ω_{m1} , consiste en la multiplicación de la fracción Nep/Ndt por ω_{lim} . Esta estimación siempre es mayor o igual que ω . La segunda estimación ω_{m2} , se define de manera

Fig. 3. DIAGRAMA DE LOS PROCESOS REALIZADOS EN EL DSPIC



Fuente: autores

Fig. 4. DIAGRAMA DE LOS PROCESOS REALIZADOS SOBRE EL COMPUTADOR



Fuente: autores

que siempre sea menor o igual que ω , según indica [1]. La aproximación final ω_{m3} , equivale al promedio armónico entre ω_{m1} y ω_{m2} . Esta estimación ω_{m3} resulta tener el menor error relativo posible, dada cualquier aproximación, por lo cual es óptimo en cuanto al error que produce.

$$\omega_{m2} = \frac{Nep}{Ndt} \omega_{lim} \text{ donde } \begin{cases} \begin{matrix} Nep \leftarrow -1 \\ Ndt \leftarrow -Ndt + 1 \end{matrix}, & \omega < \omega_{lim} \\ \begin{matrix} Nep \leftarrow Nep - 1 \\ Ndt \leftarrow 1 \end{matrix}, & \omega \geq \omega_{lim} \end{cases} \quad (1)$$

Con el método de aproximación de velocidad explicado en esta sección se realizan las pruebas de funcionamiento con valores diferentes para dt , k , y pul , las cuales serán mostradas en la sección siguiente.

5. RESULTADOS EXPERIMENTALES

En esta sección se prueba experimentalmente la parte del método de estimación de velocidad programada en el dsPIC. El procedimiento utiliza la señal cuadrada proveniente de un generador, en lugar de la señal de salida de un encoder acoplado a un eje giratorio; esto con el fin de aislar cualquier fuente de error externa al componente bajo análisis. Los flancos de subi-

da de la señal cuadrada equivalen a la entrada del dsPIC, es decir, la señal de pulsos lep , según se muestra en la Fig. 3. Así, el procedimiento de toma de datos consiste en fijar una frecuencia, equivalente a tener una velocidad constante en un eje, y tomar datos por algunos segundos. Como resultado se tiene la lectura de Nep y Ndt asociada a cada frecuencia.

Se realizaron tres experimentos con el propósito de explorar los límites del dsPIC. En estos experimentos se utilizan rangos de velocidad angular similares, además, en los tres casos la resolución del reloj ts , es 1 ms, mientras pul es 160. Este último valor se requiere para traducir Nep y Ndt a velocidades en rev/s. Recuérdese que ω_{lim} es $(1/pul)/dt$, y es $(k/pul)/dt$ si se utiliza sólo uno de cada k impulsos de la señal lep . El resultado de los experimentos se resume en la Fig. 5, en donde la altura de las curvas indica el valor de Nep medido cuando la fracción Nep/Ndt es mayor o igual que uno. A su vez, la altura de la curva indica el valor de Ndt medido cuando Nep/Ndt es menor que uno. En la Fig. 5 Izquierda: $dt = 20$ ms, $k = 1$. En la intermedia: $dt = 15$ ms y $k = 2$. En la derecha: $dt = 80$ ms, $k = 1$.

En el primer experimento se ajustan los parámetros de manera que el valor máximo de Nep esperado sea cercano al máximo de Ndt . Estos datos, según muestra la parte izquierda de la Fig. 5, son 10 para Nep y 12 para Ndt . Dado que ω_{lim} es 0.3125 rev/s; el rango de velocidades va de 0.025 rev/s a 3.125 rev/s. Véase que si bien el mínimo en la curva en V en la Fig. 5 está cerca del centro, la velocidad límite está cerca al lado izquierdo del rango de velocidades. Esto se debe a que la aproximación ω_{m1} es $(Nep/Ndt)\omega_{lim}$, es decir, proporcional a Nep e inversamente proporcional a Ndt .

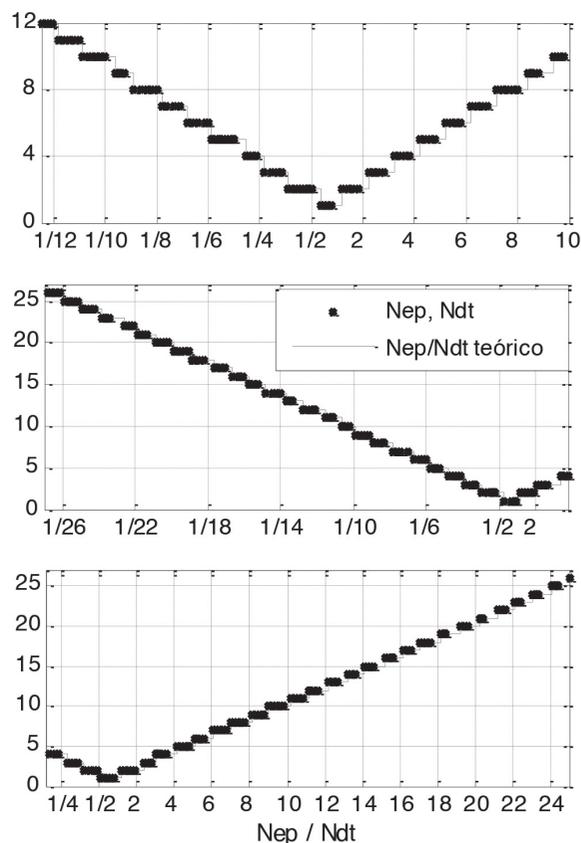
En el segundo experimento se ajustan los parámetros de manera que el mínimo de la curva en V está al lado derecho, como se muestra en la parte intermedia de la Fig. 5. Los parámetros son $dt = 15$ ms y $k = 2$, por lo tanto $\omega_{lim} = 0.833$ rev/s. El rango de velocidad correspondiente es de 0.06 rev/s a 6.25 rev/s.

En el tercer experimento, en la parte derecha de la Fig. 5, se define $dt = 80$ ms y $k = 1$, por lo tanto $\omega_{lim} = 0.078$ rev/s. Así, el rango de medida está entre 0.016 rev/s y 2 rev/s. Velocidades superiores, las cuales no tienen ninguna restricción en teoría, sí lo tienen en la práctica, dado que cuando la velocidad aumenta, aparece un punto en el cual los impulsos lep aparecen con tanta frecuencia que el dsPIC ya no es capaz de verlos, y pierde conteos; por tanto, la aproximación de la velocidad angular se pierde. El límite encontrado en estos experimentos es de 30. Cuando el conteo está por debajo de este límite, el conteo de Nep y Ndt coincide con lo esperado en teoría, para todas las muestras tomadas.

El efecto del cambio de la ubicación del mínimo de las curvas V en la Fig. 5 está en el valor del error relativo entre la velocidad real y la aproximada. Ese error relativo disminuye a medida que el número Nep o Ndt aumenta, y se hace máximo cuando Nep y Ndt son uno, como se muestra en [16]. La aplicación que se dé al método de aproximación que se propone define cuál curva en V es mejor. El inconveniente de la configuración intermedia en la Fig. 5 es que se produce un error muy grande a velocidades altas, a las cuales normalmente gira una máquina que genera el movimiento. De otra parte, el problema para la configuración de abajo, es que requiere de dt altos, por lo tanto la velocidad se actualiza

a intervalos de tiempo grandes. Este tiempo de actualización puede ser contraproducente para el control de velocidad, de lo cual se va a mostrar un ejemplo en la sección siguiente.

Fig. 5. RESULTADOS EXPERIMENTALES DEL CÁLCULO DE NEP Y NDT



Fuente: autores

6. APLICACIÓN ESTRATEGIA NUEVA: CONTROL DE VELOCIDAD

En esta sección se aplica el método propuesto para la aproximación de velocidad en el control de la velocidad angular de un motor de corriente continua. La salida del método de estimación de velocidad, ω_{m3} , es la señal del sensor en un esquema clásico de control, como se muestra en la parte superior de la Fig. 6. En este trabajo la estrategia de control fue implementada en Simulink.

La salida del controlador u en la Fig. 6, llega al motor pasando a través de una tarjeta de adquisición de datos y luego por una etapa de potencia. La tarjeta es la PCI 6024E, de 12 bits, y

con el rango de lectura entre -10 V y 10 V. La etapa de potencia es un amplificador clase B, tipo Darlington, hecho con transistores BJT. Por otra parte, el motor se opera entre 0 y 5.5 V. Una de las características del montaje, importantes para control, es que la etapa de potencia, y el mismo motor, poseen una banda muerta, la cual hace que el motor sólo gire cuando se supera un voltaje de 2.4 V; esta banda muerta se compensa adicionando 2.4, como se muestra en la Fig. 6. Por último, el encoder incremental, en la parte inferior de la Fig. 6, tiene 160 ranuras por vuelta, y su voltaje de alimentación es de 3.3 V.

La referencia r es una señal a escalones, la cual puede estar entre 0 rev/s y 18 rev/s, con duraciones aleatorias entre 30 a 55 segundos. A esta referencia se le aplica un filtro pasabajos, con el fin de producir transiciones lentas, dado que la capacidad del dsPIC sólo permite utilizar $ts \geq 1$ ms. El bloque denominado cuantización se encarga de que el valor de referencia coincida con los valores que entrega el sensor. Recuérdese que ω_{m3} se presenta a saltos, como se ve en la Fig. 5. Así, el método de estimación de velocidad nuevo se prueba a un tiempo dt de 40 ms, mientras que k es 12. Con estos valores se obtiene un valor para ω_{lim} igual a 1.88 rev/s, mientras la velocidad máxima es cercana a 18 rev/s.

El algoritmo de control es un PI (Proporcional Integral), como se muestra en la Fig. 6 y se describe en (2). La entrada del control es el error e , es decir, la diferencia entre el valor de referencia y el valor medido por el sensor (ω_{m3}). El criterio principal para realizar el control de velocidad es conseguir que el sistema sea estable, sin importar cuánto demore la velocidad en estabilizar. La razón de este criterio de control es que en lugar de estar diseñando un controlador, se está probando el método de estimación de velocidad nuevo.

$$C(s) = kp + \frac{ki}{s} \quad (2)$$

El ajuste experimental realizado para el control PI de velocidad angular dio como resultado que kp es de 5×10^{-3} y ki igual a 50×10^{-3} . Con estos valores se logró un tiempo de estabilización cercano a 15s, además un error de estado estacionario de 0% y un sobrepico de 0%. En la Fig. 7 se observan los resultados obtenidos para el control de velocidad con las configuraciones aquí descritas. Puede verse cómo la señal por medio del método de aproximación de velocidad propuesto sigue de cerca la referencia, como se espera. Así, el error es cero para valores de r constantes, y sólo cambia sólo cuando esta señal tiene transiciones.

Fig. 6. ESQUEMA DEL CONTROL DE VELOCIDAD

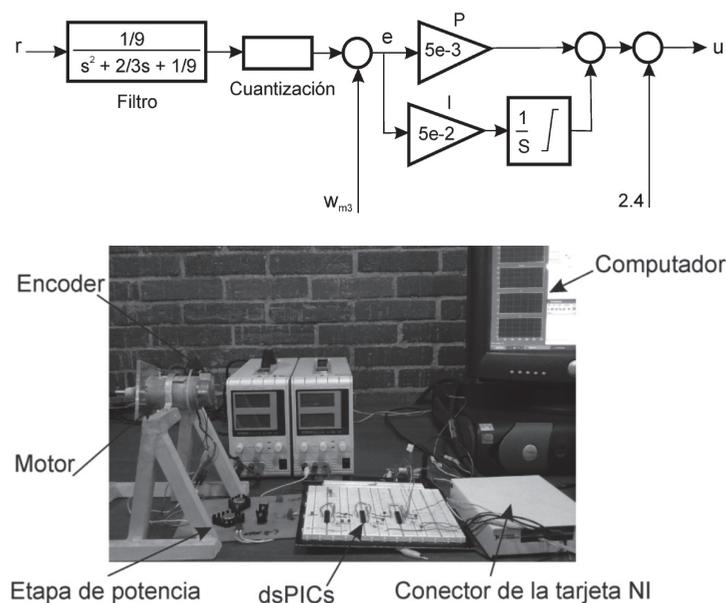
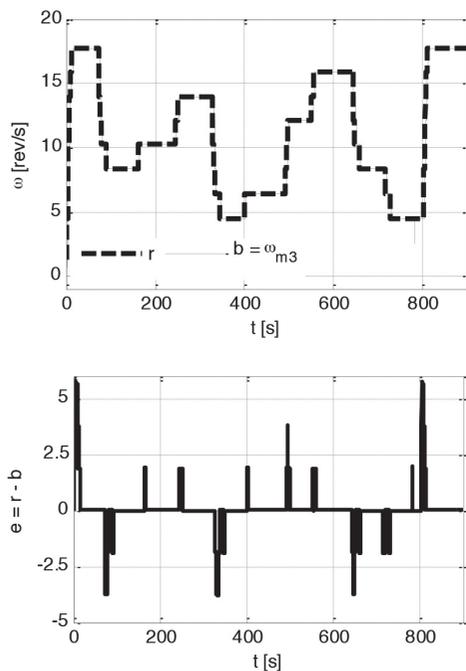


Fig. 7. RESULTADOS DEL CONTROL DE VELOCIDAD DE UN MOTOR



Fuente: autores

7. CONCLUSIONES

En este artículo se probó la aplicación de un algoritmo de estimación de velocidad nuevo en un dsPIC. Se verificó que el algoritmo presenta una sola aproximación de velocidad angular cuando la velocidad angular es constante, contrario a lo que ocurre cuando se aplican los algoritmos tradicionales conocidos como a tiempo fijo y a espacio fijo. Además, se pudo ver que el método también es útil cuando la velocidad es variable, lo cual se verificó mediante la aplicación de un lazo de control para regular la velocidad en un motor de corriente directa. Estos resultados permiten asegurar que el método nuevo es aplicable industrialmente.

Un aspecto crítico durante la implementación del diseño consiste en la sincronización de las señales, en especial los impulsos de posición y tiempo, lep e ldt , respectivamente. Esto dado que una señal influye en el conteo de la otra y viceversa. Mientras la definición del algoritmo se hace con la suposición de que las señales son perfectas; que las transiciones entre señales no tienen duración y que los impulsos son ideales; en la realidad estos procesos toman tiempo, lo cual se

constituyó en la dificultad más grande para implementar el método de medición.

Con el desarrollo de este proyecto se abren algunas posibilidades para estudios futuros, por ejemplo es importante buscar implementar el algoritmo en dispositivos que permitan tiempos muy inferiores a los milisegundos, como fue el caso en este artículo. De otra parte, puede ser interesante estudiar la aplicación de este método de aproximación de velocidad, o razón de cambio, para la aproximación de la razón de cambio en procesos distintos a las señales provenientes de un encoder incremental. Si se revisa la definición del método, puede verse que en realidad no es necesario asumir que la señal proviene de un encoder, si no que la única suposición es que la medida de la señal que cambia se hace a incrementos fijos. En el caso del encoder es de espacio, pero en realidad puede ser cualquier variable física.

REFERENCIAS

- [1] M. Nandayapa, C. Mitsantisuk y K. Ohishi, "High resolution position estimation for advanced motion control based on FPGA," in Proceedings of IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society, pp. 3808-3813, 2012.
- [2] C. Negrea, V. Hayward, C.J. Chen, "An improved speed identification method using incremental encoder in electric drives," in Proceedings of IEEE International Conference on Automation Quality and Testing Robotics (AQTR), pp. 536-540, 2012.
- [3] J. Voldán, "Use encoder speed sensors for the calculation of torque moment," in Proceedings of 13th International Carpathian Control Conference (ICCC), pp. 768-771, 2012.
- [4] S. D'arco, L. Piegari y R. Rivo, "An Optimized Algorithm for Velocity Estimation Method for Motor Drives," in Proceedings of Symposium on Diagnostics for Electric Machines, Power Electronics and Drivers, Atlanta, GA, USA, pp. 76 - 80, Ago., 2003.
- [5] G. Liu, "On velocity estimation using position measurements," in Proceedings of the 2002 American Control Conference, vol. 2, pp. 1115 - 1120, May., 2002.
- [6] K. Hachiya y T. Ohmae, "Digital speed control system for a motor using two speed detection methods of an incremental encoder," in Proceedings of Power Electronics and Applications, 2007 European Conference, pp. 1 - 10, Sep. 2007.

- [7] J. Lygouras, T.P. Pachidis, K.N. Tarchanidis, V.S. Kodogiannis, "Adaptive High-Performance Velocity Evaluation Based on a High-Resolution Time-to-Digital Converter," in *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 9, pp. 2035 – 2043, Sep. 2008.
- [8] F. Janabi-Sharifi, V. Hayward, C.S.J. Chen, "Discrete-time adaptive windowing for velocity estimation," *Control Systems Technology*, *IEEE Transactions on*, vol. 8, no. 6, pp. 1003 – 1009, Nov. 2000.
- [9] M-F Benkhoris y M. Ait-Ahmed, "Discrete speed estimation from a position encoder for motor drives," in *Proceedings of Power Electronics and Variable Speed Drives, Sixth International Conference on (Conf. Publ. No. 429)*, pp. 283 – 287, 23-25, Sep. 1996.
- [10] I.I. Incze, A. Negrea, M. Imecs, C. Szabó, "Error compensation methods in speed identification using incremental encoder," *Proceedings of Electrical and Power Engineering (EPE), International Conference and Exposition on*, pp. 441 – 445, 25-27, Oct. 2012.
- [11] L. Kerhuel, "Simulink block set embedded target for microchip devices," 2012, [Online]. Available: http://www.kerhuel.eu/wiki/Simulink_-_Embedded_Target_for_PIC
- [12] R. Merry, R. Molengraft, y M. Steinbuch, "Error modeling and improved position estimation for optical incremental encoders by means of time stamping," in *Proceedings of the 2007 American Control Conference*. New York City, USA, pp. 3570 – 3575, Jul., 2007.
- [13] R. Petrella, M. Tursini, L. Peretti, M. Zigliotto, "Speed measurement algorithms for low-resolution incremental encoder equipped drives: a comparative analysis," in *Proceedings of Electrical Machines and Power Electronics, 2007. ACEMP '07. International Aegean Conference on*, vol. 1, no. 1, pp. 780 – 787, Sep., 2007.
- [14] R. Petrella, y M. Tursini, "An Embedded System for Position and Speed Measurement Adopting Incremental Encoders," *IEEE Transactions on industry applications*, vol. 44, no. 5, pp. 1436 – 1444, Sep. 2008.
- [15] T. Tsuji, T. Hashimoto H. Kobayashi, M. Mizuochi, K. Ohnishi, "Wide-Range Velocity Measurement Method for Motion Control," in *IEEE Transactions on industrial electronics*, vol. 56, no. 2, pp. 510 – 519, Feb., 2009.
- [16] D. Rairán y J. Fonseca, "Algoritmo para la aproximación de la velocidad de giro de un eje mediante un encoder incremental," en *Revista Ingeniería y Universidad*, vol. 17, no. 2, pp. 293 – 309, Sep., 2013.