# Ideas para medir la velocidad con los encoders

## General

Al inicio, cada vez que el timerx desborda incremento una variable auxiliar cantOVerflow. Cuando detecto el primer cambio de estado o flanco (interrupción por pin), el tiempo transcurrido va a ser:

Ya que la cantidad de pulsos contados por el timer es igual al valor actual del timer más cantOVerflow veces el máximo valor que cuenta.

Una vez guardado el valor del timer actual, reseteamos cantOVerflow. Así, la siguiente vez que se genere una interrupción el valor de cantOVerflow reflejará la cantidad de overflows que hubo desde la interrupción pasada. Si hacemos el cálculo de tiempo como antes, tendremos un error de cálculo debido a que en realidad no transcurrieron cantOVerflow ciclos de conteo completo del timer, sino que el primero arrancó a contar desde el valor que tenía el timer cuando se hizo la interrupción anterior. Para eso agregamos una variable auxiliar de corrección cuyo valor sea siempre el valor del timer al momento de la útima interrupción:

## Fórmula de Velocidad y Propagación de Errores

Conociendo el tiempo entre interrupciones, podemos calcular la velocidad en rpm usando:

Donde es la cantidad de dientes del encoder (como interrumpo por flancos de subida y de bajada se generan el doble de interrupciones por vuelta).

Juntando la cantidad de conteos realizados por el timer en una sola variable:

La fórmula final resulta:

Podemos considerar la frecuencia del cristal lo suficientemente estable para fines prácticos, por lo que el error estará dado por la mínima variación que permite el Timer. Trabajando a una frecuencia dada , la discretización haría que las frecuencias anteriores y posteriores a detectar sean:

$$$

## Filtro para Uniformidad de Medición

Dado que la fabricación del encoder no es perfecta, los tiempos entre interrupciones no van a ser perfectamente iguales, lo cual introduciría error en la medición si se desea estimar conociendo una sola medición de . Para evitar este problema, calculamos con una versión filtrada de igual al promedio de las últimas mediciones.

Cuando ocurra un cambio de velocidades se deberá esperar un cierto transitorio antes de poder conocer la velocidad real ( mediciones). Hasta que no haya terminado este tiempo la velocidad indicada será incorrecta.

## Rango de Tiempos a Medir

Si la velocidad máxima a medir es 1500rpm, usando el encoder de 8 dientes (como interrumpo por flancos de subida y de bajada se generan 16 interrupciones por vuelta) tendríamos un tiempo entre interrupciones mínimo de:

Si usamos la ecuación (1), el error de la medición va estar dado por la variación de tiempo entre dos valores del timer:

El peor caso para el erro relativo va a ser cuando el tiempo medido sea mínimo, en cuyo caso:

## Determinación del Preescalador

Las interrupciones por overflow del timer2 se hacen cada:

Para utilizar el mismo timer para mediciones de tiempo conviene que el preescalador tenga el menor valor posible. Para ello utilizamos un valor menor al necesario para generar el tiempo de muestreo deseado para el sistema y compensamos esta diferencia con un contador que haga las veces de preescaler. Es decir, en cada interrupción se incrementa un contador; cuando el contador llegue a un valor determinado significará que transcurrió el tiempo deseado y se procederá con la subrutina base del sistema, la cual consiste en medición, cálculo e implementación del PID y posible transmisión de datos a la computadora. De esta manera, el tiempo de muestreo real será un múltiplo de .

El detalle importante de esta idea es que el tiempo entre interrupciones por timer, , debe alcanzar para realizar todas las operaciones necesarias en el ciclo en que el contador alcanza el valor de referencia. Se realizó una prueba inicial que medía la demora en realizar una serie de instrucciones representativas, específicamente, dos transmisiones por serial y varias cuentas aritméticas (cantidad sobredimensionada) que reemplazarían lo que más adelante será el cálculo del controlador. El resultado a una velocidad de transmisión de 9600 baudios era de unos 5,2ms, pero con 2.106baudios se redujo a unos 192us máximos (el valor no es siempre constante). Para dejar un margen de seguridad, se toma .

Considerando el máximo valor posible de , 255, y sabiendo que , podemos encontrar el valor mínimo del preescalador que cumple la restricción impuesta:

Dado que los valores posibles para el preescalador son 1, 8, 32, 64, 128, 256 o 1024, seleccionamos que valga 32. Una vez fijado el preescalador, la precisión de la medición de tiempo ya está dada, por lo que aumentar el valor final de con no genera conflicto y permite mejorar el margen de seguridad para la realización de las instrucciones de la subrutina base del sistema.