**PRÁCTICA POSICIÓN Y VELOCIDAD**

**OBJETIVOS**

* Aprender a utilizar un encoder para realizar mediciones de posición y velocidad.
* Realizar un control de velocidad y posición utilizando un encoder.

**DURACIÓN**

* 2 horas

**MATERIALES**

* Tarjeta de desarrollo SIE-2
* Encoder
* Conectores para tarjeta
* Pilas
* Llanta y transportador

**DICCIONARIO**

En esta guía se encuentran algunos conceptos que pueden ser nuevos, a continuación una pequeña definición de ellos y algunos sitios de interés que se recomiendan para comprender mejor estos términos.

* **Sistema Discreto**
* **Sistema de Control**
* **Lazo Cerrado**
* **Realimentación**
* **Ciclo Útil**
* **Control Proporcional**
* **Velocidad Angular**
* **Posición Angular**
* **Resolución de un sensor**

**TRABAJO PREVIO**

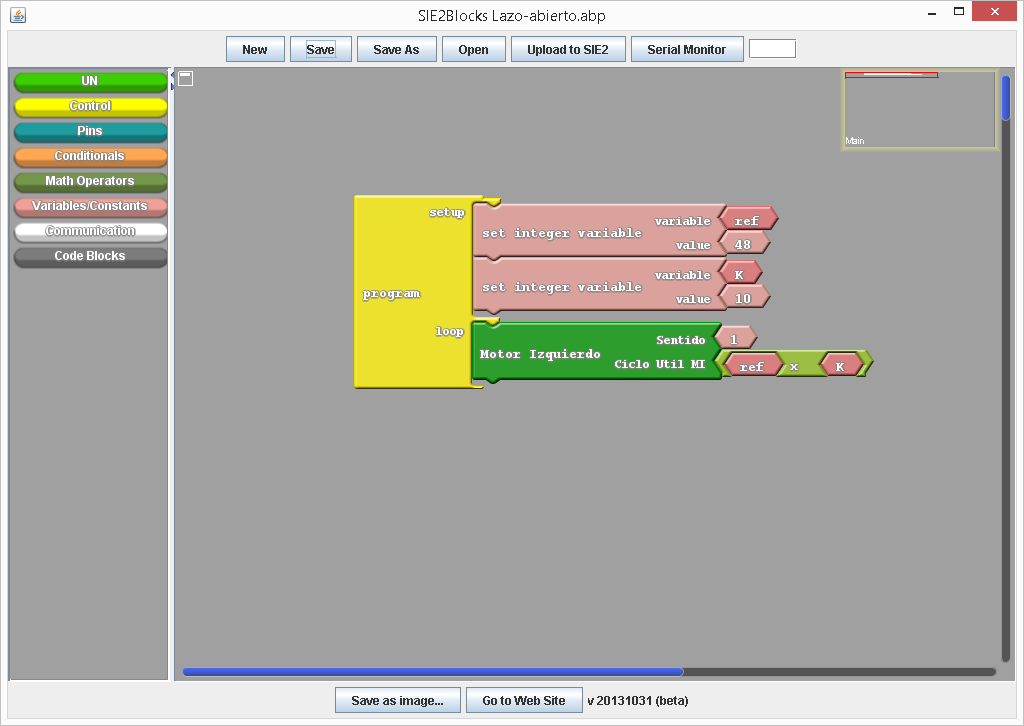
Es muy recomendable repasar los términos sugeridos en la sección previa y entender el principio de funcionamiento de un sistema de control y un sistema discreto, así mismo se deben tener claros los conceptos de posición y velocidad de un motor. Es muy importante aprender sobre el funcionamiento de un encoder, en especial las características del encoder Pololu 0J3334 que se está utilizando.

**TRABAJO PRESENCIAL**

Con el fin de entender mejor el concepto de control y realimentación se dividirá la práctica en dos partes, una primer parte donde se implementará un sistema de control sin realimentación, o de lazo abierto y una segunda parte donde se hará realimentación y se utilizará un sistema de lazo cerrado.

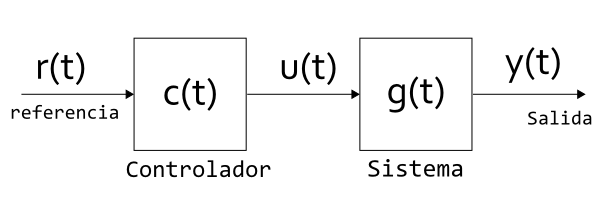
**Lazo Abierto**

Como se explicó previamente, para empezar vamos a realizar un sistema de control de lazo abierto, para ello vamos a necesitar el motor y la tarjeta de desarrollo. Vamos a conectar únicamente el motor a la tarjeta de desarrollo y ejecutar el código de la Figura 1.



**Figura 1. Código de lazo abierto**

La Figura 2 nos muestra un diagrama de un sistema de control de lazo abierto, que nos servirá de explicación al programa anterior, primero encontramos una referencia o valor deseado **ref** o **r(t)**, luego encontramos un bloque correspondiente al controlador **c(t)** que en este caso es un control proporcional cuya salida **u(t)** es igual a la multiplicación de la contante **K** por la referencia **ref**, posteriormente sigue el sistema **g(t)** que representa el motor y finalmente la salida **y(t)** igual a la velocidad o posición del motor.



**Figura 2. Diagrama de lazo abierto**

Inicialmente vamos a analizar nuestro sistema desde el punto de vista de la posición, como vemos nuestra referencia está fijada en el valor 48, lo que indica que el motor después de alcanzar este valor debe detenerse. Ahora vamos a ejecutar este código y ver que ocurre con el motor, ¿Se detiene en algún punto? Si el motor se detiene ¿En que punto lo hace? ¿Cuáles son las unidades de la referencia de 48? ¿Grados? ¿Vueltas? ¿Pulsos del encoder? ¿Otra unidad?

Ahora vamos a modificar las propiedades de nuestro controlador, en el código de Ardublock vamos a modificar la variable **K** y a darle valores mayores y menores. ¿Esto modifica el comportamiento del sistema? ¿Al modificar estos valores logramos que el motor se detenga en algún punto? ¿Cuál es el mayor valor y el menor valor antes que el sistema deje de presentar cambios? ¿Por qué?

Ahora vamos a analizar nuestro lazo de control desde el punto de vista de la velocidad. Ejecutaremos nuevamente el código de la Figura 1 y analizaremos la velocidad angular del motor ¿Es constante la velocidad angular del motor? Desde el punto de vista de la velocidad del motor, ¿A que unidades corresponde al referencia de 48? ¿Grados por segundo? ¿Vueltas por segundo? ¿Pulsos del encoder por segundo? ¿Otra unidad?

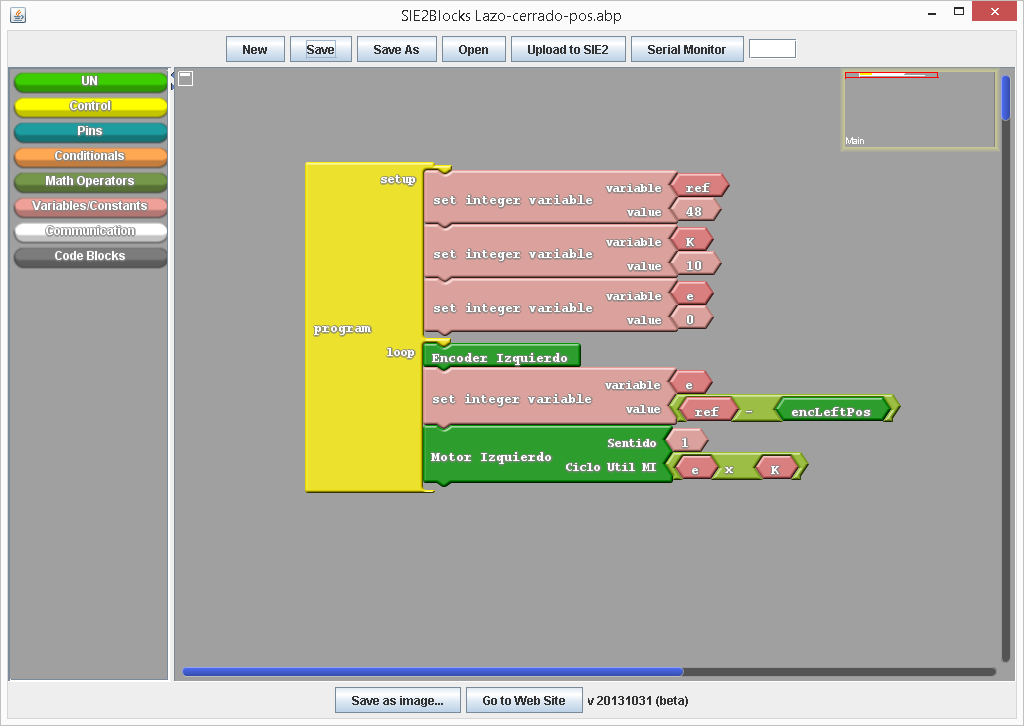
Ahora vamos a poner nuestro dedo muy suavemente en el motor con la intensión de disminuir un poco la velocidad ¿Es esta la velocidad de referencia? ¿El controlador hace algo para retomar la velocidad previa a poner el dedo?

**Lazo Cerrado**

Continuando con el desarrollo de la práctica vamos ha hacer un control de posición en lazo cerrado, para esto vamos a instalar el sensor que servirá de realimentación al sistema, en este caso el encoder. En la Figura 3 podemos ver como debe quedar el encoder después de su instalación.

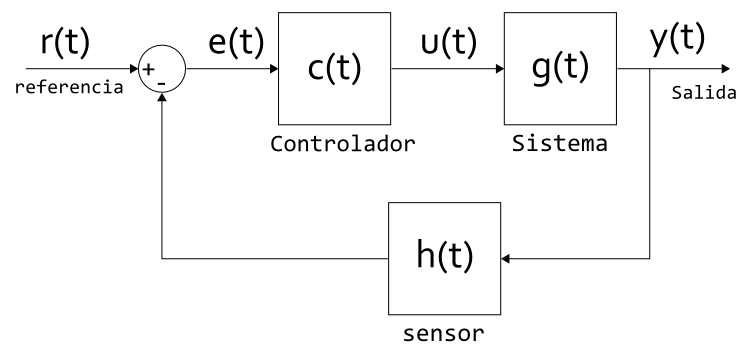
  
**Figura 3. Instalación del encoder**

Después de ubicar el encoder procedemos a conectar los cables de alimentación y salida de los canales A y B a la tarjeta de desarrollo. Ahora vamos al programa Ardublock y corremos el programa de la Figura 4.



**Figura 4. Código de lazo cerrado de posición**

Este programa es similar al anterior con unos pequeños cambios, puesto que se han incluido dos elementos nuevos, el sensor, representado por bloque **h(t)** en la Figura 5, y el error, o **e(t),** que es calculado como el valor medido en la salida por el sensor, menos la señal de referencia. Ahora la señal de entrada al controlador no es la referencia deseada, sino la diferencia entre esta referencia y el valor actual de posición o velocidad.



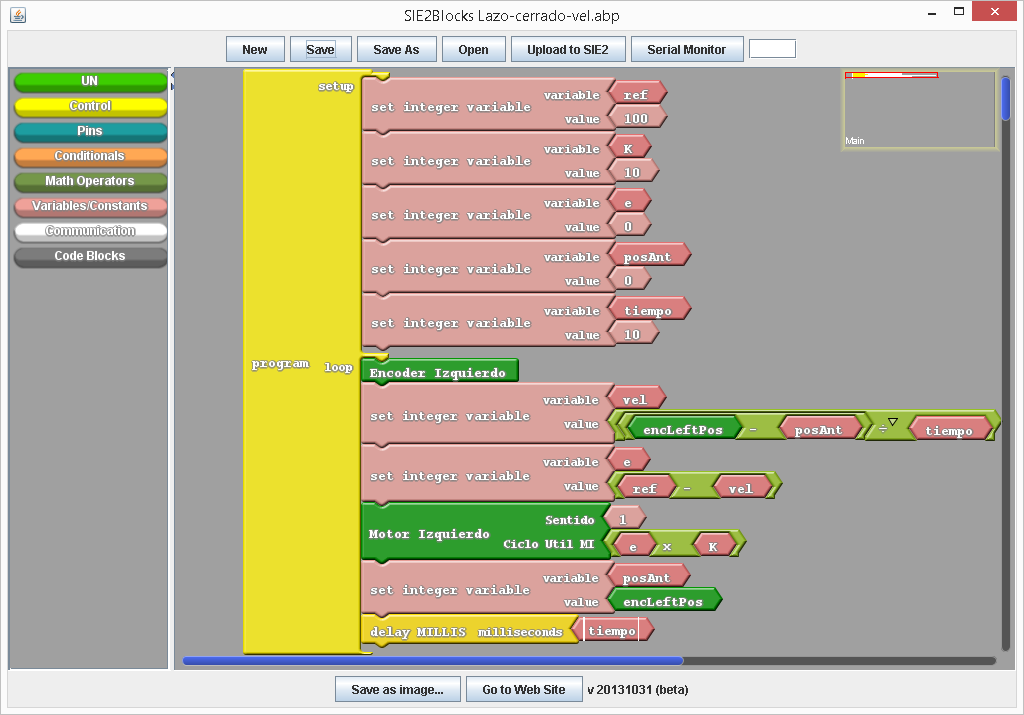
**Figura 5. Diagrama de lazo cerrado**

Vamos a iniciar analizando el lazo de control en posición, ejecutando el código de la Figura 4, que tiene una referencia con valor de 48, esto quiere decir que una vez se cuenten 48 pulsos del encoder se alcanzará el valor de referencia y el motor se detendrá luego de dar una vuelta ¿Qué valor debe tener la referencia para que la rueda gire 180°? ¿A qué unidades corresponde el valor en la referencia de 48? ¿Qué conversión debe hacerse para que la referencia esté en grados, vueltas o pulsos? Si devolvemos suavemente el motor unos pocos grados ¿Este retoma a la posición previa o se mantiene donde se dejó? ¿Esta nueva posición corresponde a una nueva referencia?

Ahora vamos a modificar el valor de la constante del controlador e ir probando valores mayores y menores al actual ¿Qué comportamiento tiene el sistema con valores mayores y con valores menores? ¿Qué ocurre cuando se usan valores muy grandes en la constante proporcional?

Continuando con nuestra práctica ahora experimentaremos con un control de velocidad. Primero que todo se debe tener claro el concepto de velocidad angular y su representación y cálculo en sistemas discretos; esto es muy importante porque nuestro sensor, el encoder, es discreto, y nos permite medir únicamente la posición de un cuerpo en rotación contando pulsos, por lo que la velocidad angular debe calcularse contando estos pulsos en intervalos de tiempo fijos. Teniendo esto claro, vamos a analizar el código de la Figura 6.

Como podemos ver desde un principio hay dos nuevas variables, una llamada **posAnt**, que almacena la posición anterior y otra llamada **tiempo**, con el intervalo de tiempo fijo.



**Figura 6. Código de lazo cerrado de velocidad**

Lugo se hace el cálculo de la velocidad al dividir la posición actual menos la posición anterior en el intervalo de tiempo, ya con este valor hacemos el cálculo del error igual que en el caso de la posición y con este error alimentamos el controlador. Posteriormente se mide la posición, y se almacena en la variable que será la posición anterior en el siguiente ciclo, y se espera el intervalo de tiempo fijo, en este ejemplo de 10 milisegundos.

En este momento se tiene un mejor entendimiento del código por lo que vamos a programar la tarjeta de desarrollo con este y analizar que sucede. ¿Cuál es el comportamiento del motor cuando se ejecuta el código? ¿La velocidad del motor es estable? ¿Cuáles son las unidades de la referencia? ¿Ciclos por segundo? ¿Grados por segundo? ¿Pulsos por segundo? ¿Qué conversión debe hacerse para que la referencia esté en grados por segundo, vueltas por segundo o pulsos por segundo? Si ponemos suavemente el dedo para frenar un poco el motor ¿Qué comportamiento presenta? ¿La velocidad es la misma al momento previo de poner el dedo? ¿Qué pasa cuando se suelta el dedo?

Si ahora variamos los valores de la constante del controlador **K** ¿Qué comportamiento presenta ahora el sistema para valores mayores y menores? ¿El motor ahora es más rápido o más lento?

Si se modifica la variable **tiempo** (tiempo de muestreo) ¿Qué comportamiento tiene el motor si el tiempo es mayor o menor al actual? ¿Qué pasa con tiempos de muestreo muy grandes o muy pequeños?