**PRÁCTICA VI GENERACIÓN DE TRAYECTORIAS**

**OBJETIVOS**

* Aprender a generar trayectorias en un robot móvil a partir de figuras geométricas simples.
* Aplicar los conceptos de control de velocidad y posición en la generación de trayectorias

**DURACIÓN**

* 2 horas

**MATERIALES**

* SIEBOT
* Hojas con trayectorias

**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.

* **Trayectoria**
* **Velocidad Tangencial**
* **Posición**
* **Vector**
* **Distancia Recorrida**
* **Perímetro**
* **Longitud de arco**
* **Radián**
* **Ángulos suplementarios**
* **Teorema del seno**

**TRABAJO PREVIO**

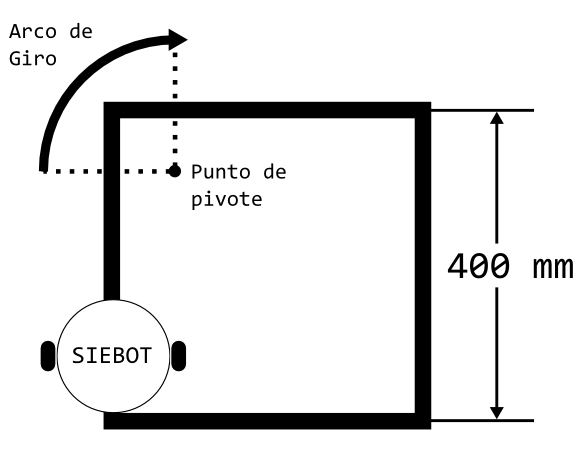
Es muy recomendable repasar los términos sugeridos en la sección previa y entender el funcionamiento de un encoder y como calcular la posición a partir de este elemento.

**TRABAJO PRESENCIAL**

Para esta práctica se van a seguir trayectorias simples a partir de figuras geométricas, con el fin de aprender las bases del seguimiento de una trayectoria, exploraremos el seguimiento de trayectorias cuadradas, triangulares y circulares. La dificultad de esta práctica radica en 2 aspectos principales, el primero es calcular la posición del robot y la distancia recorrida y el segundo calcular el giro del robot.

**Trayectoria Cuadrada**

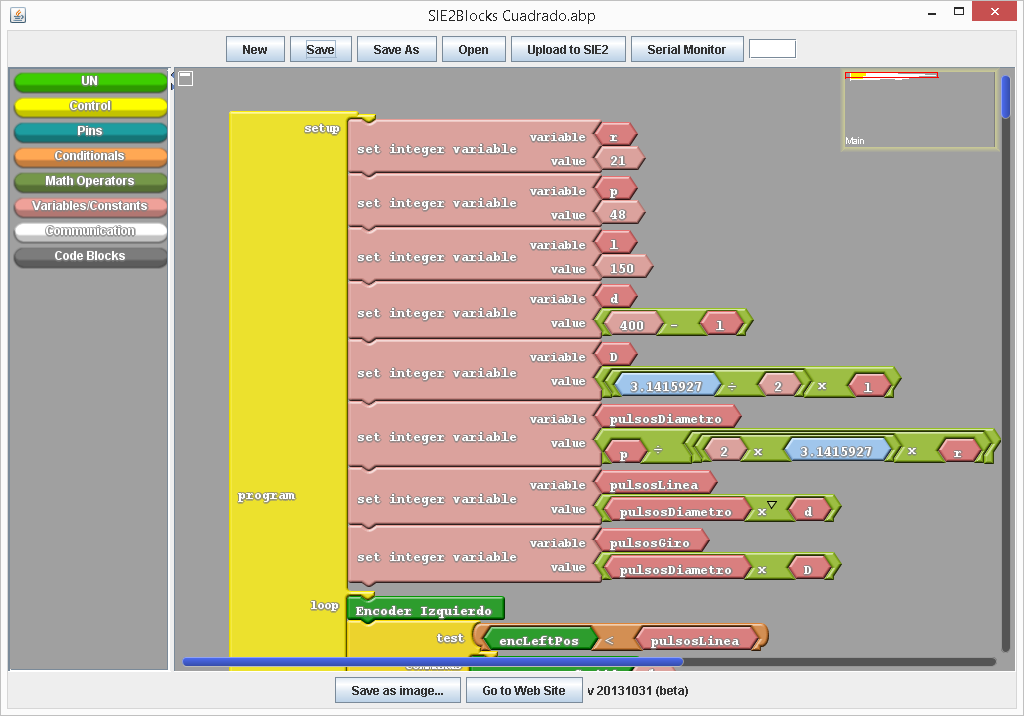
Vamos a iniciar la práctica haciendo el seguimiento de una trayectoria cuadrada de longitud 40 cm o 400 mm como se ve en la Figura 1.



**Figura 1. Trayectoria cuadrada y posición inicial del SIEBOT**

Es importante posicionar el SIEBOT como se ve en la Figura 1. puesto que el robot girará hacía su derecha, usando únicamente el motor izquierdo para esto, ya se explicará a continuación. Para que el robot se desplace en línea recta sus dos ruedas deben girar en el mismo sentido y a la misma velocidad, si una rueda más rápido que la otra el robot girará en la dirección contraria, por ejemplo, si la rueda izquierda gira más rápido que la derecha, el robot hará un giro hacía la derecha, es importante tomarse un tiempo para analizar esto.

Para hacer el seguimiento de la trayectoria se utilizará el código de las Figuras 2 y 3 que se explicará a continuación.



**Figura 2. Inicialización programa trayectoria cuadrada**

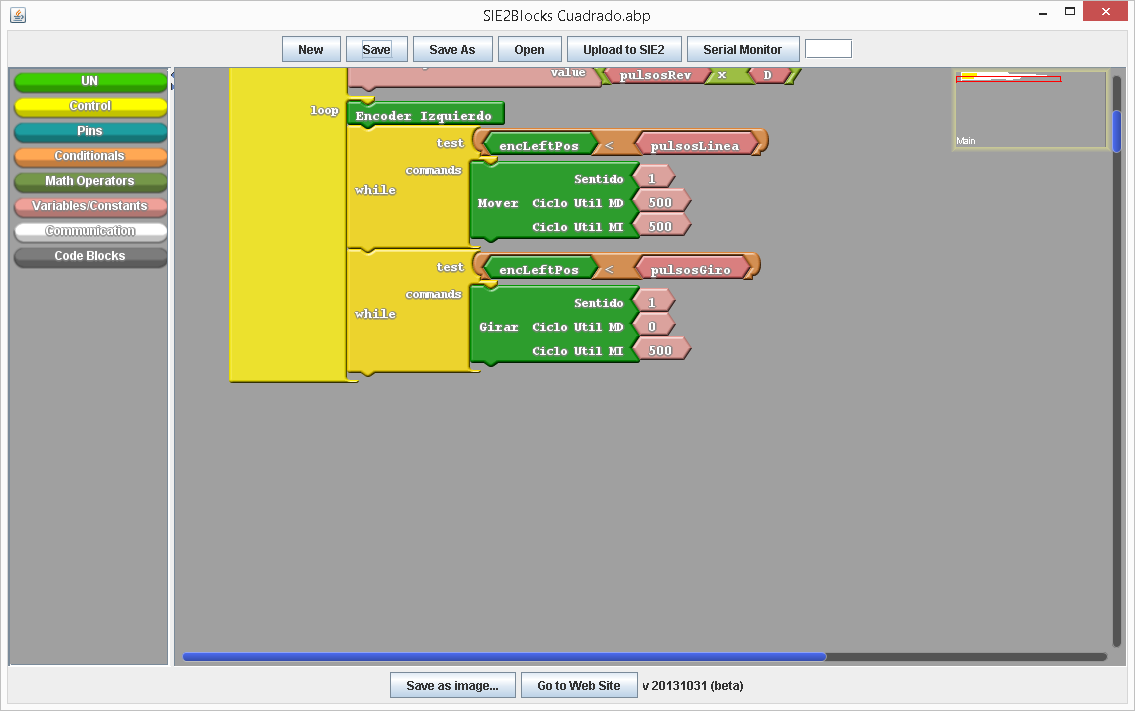
La Figura 2 contiene la inicialización del programa, en esta se configuran las siguientes variables que van a mantenerse fijas en la ejecución del programa

* **r = 21** radio de la rueda en milímetros
* **p = 48** número de pulsos por vuelta del encoder (resolución)
* **l = 150** distancia entre las ruedas del robot en milímetros

Las siguientes variables van a ser calculadas a partir de las anteriores

* **d** longitud de los lados del cuadrado en milímetros
* **D** arco de giro en las curvas en milímetros
* **pulsosRev** pulsos por recorrido en una revolución de las ruedas del robot
* **pulsosLinea** número de pulsos del encoder equivalentes a la trayectoria recta
* **pulsosGiro** número de pulsos del encoder equivalentes a la trayectoria de giro

la explicación de las variables anteriores es la siguiente: La variable **d** es la longitud que recorre en robot en línea recta, se debe calcular así porque se resta **l/2** en cada esquina por el punto de pivote del robot en cada giro. La variable **D** es la longitud del arco que debe seguir el robot para girar, en este caso el radio de este arco es igual a la distancia **l** entre ruedas del robot. La variable **pulsosDiametro** calcula pulsos que se cuentan en el recorrido que hace el robot después de un giro en las ruedas del robot, detengámonos un momento en esto; si se hacen girar las ruedas del robot una sola vez, el robot habrá avanzado una distancia, esta distancia es equivalente a un diámetro de las ruedas del robot y habrá contado los pulsos de una vuelta, con lo que se hace una conversión de pulsos por vuelta a pulsos por diámetro. La variable **pulsosLinea** corresponde a cuantos pulsos se deben contar para completar la trayectoria en línea recta, se calcula multiplicando los pulsos que se cuentan en el recorrido del diámetro por la longitud de la línea recta; así sabremos cuantos pulsos se cuentan al recorrer una distancia **d**. La variable **pulsosGiro** se calcula de manera similar a **pulsosLinea** pero con la distancia que se recorren en el arco de giro.

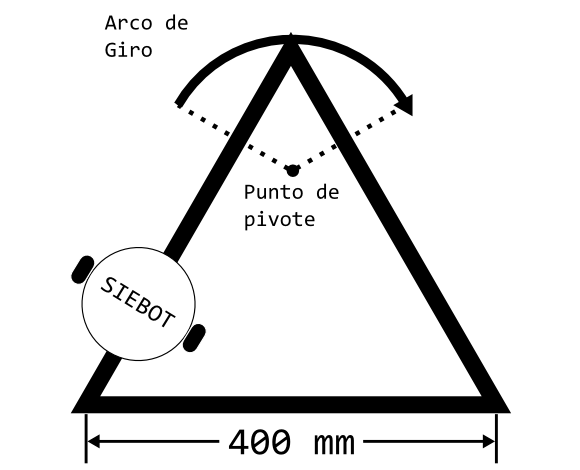


**Figura 3. Cálculo de trayectoria cuadrada**

La Figura 3 nos muestra como se usan los valores encontrados previamente para seguir la trayectoria, en un principio el SIEBOT se mueve en línea recta (con los dos motores girando a la misma velocidad) hasta que el conteo de pulsos iguale a los pulsos necesarios para completar la línea recta, con lo que pasa a detener el motor derecho para usarlo como pivote y permitir el giro, el motor izquierdo se activa únicamente hasta que se cuentan los pulsos necesarios para hacer el recorrido del arco de giro, para luego iniciar el proceso nuevamente ¿Qué modificaciones habría que hacer al programa para hacer un recorrido rectangular?

**Trayectoria Triangular**

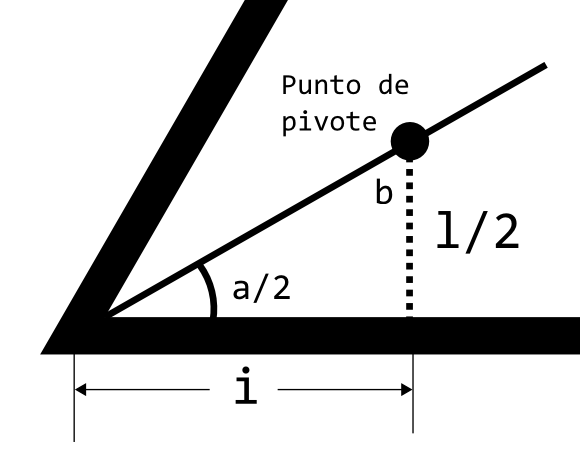
Ahora vamos a hacer un recorrido en forma de un triángulo equilátero con lados de longitud 400 mm como el de la Figura 4.



**Figura 4. Trayectoria triangular**

El cálculo de esta trayectoria es muy similar al de la trayectoria cuadrada, sin embargo se deben hacer modificaciones en las longitudes de los recorridos lineal y arco de giro, puesto que ahora se debe girar 120°, ángulo suplementario al ángulo de la figura. Si quiero girar 30° ¿Cuál debe ser el ángulo del arco de giro? ¿Por qué los ángulos del arco de giro y la figura deben sumar 180°?

Vamos a iniciar calculando el pivote, en la Figura 5 podemos ver que este punto está ubicado a una distancia **i** de la esquina y a una distancia **l/2** del borde, siendo **l** la longitud entre las ruedas del robot. A continuación aprenderemos a calcular esta distancia **i**.



**Figura 5. Punto de pivote**

Para calcular **i**, vamos a usar los valores conocidos, siendo **a** el ángulo de la esquina y **b** su ángulo complementario, a partir de esta información usaremos la ley del seno para calcular **i**.



Como **b** es igual a **90 - a/2** se puede simplificar como **cos(a/2)** y al hacer el despeje tenemos que

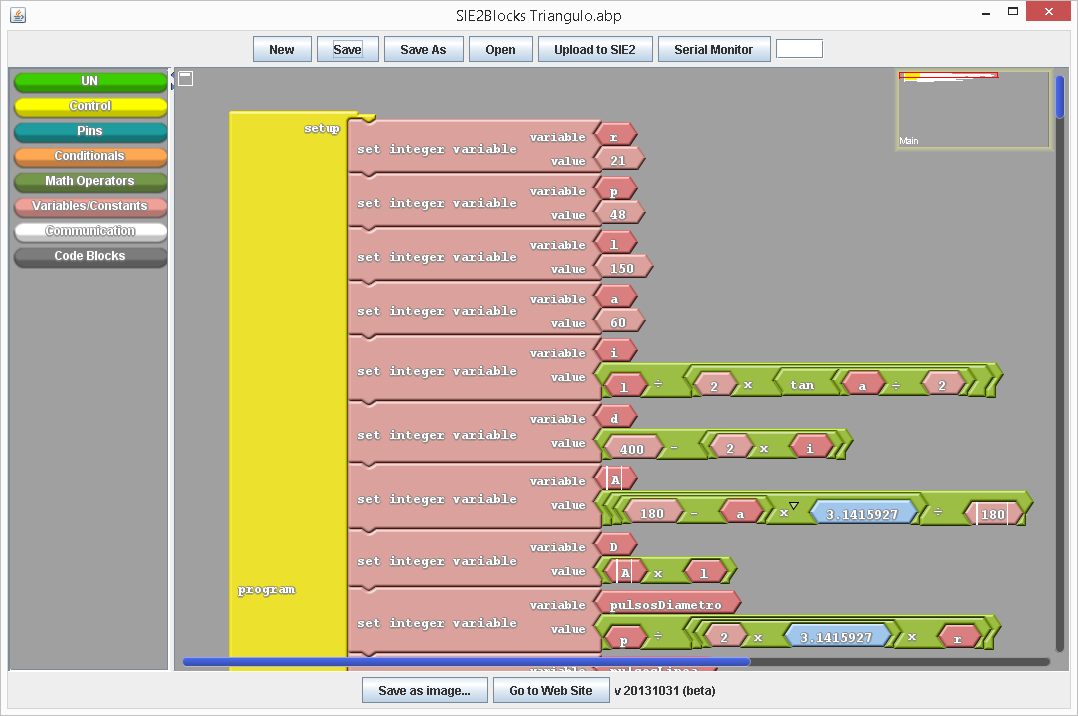


Como se había mencionado antes se debe hacer un pequeño cambio en el cálculo del arco de giro, puesto que ahora se debe determinar para cualquier ángulo, esto se logra al hacer la conversión del ángulo de grados a radianes y luego multiplicándolo por el radio de giro con la siguiente fórmula.



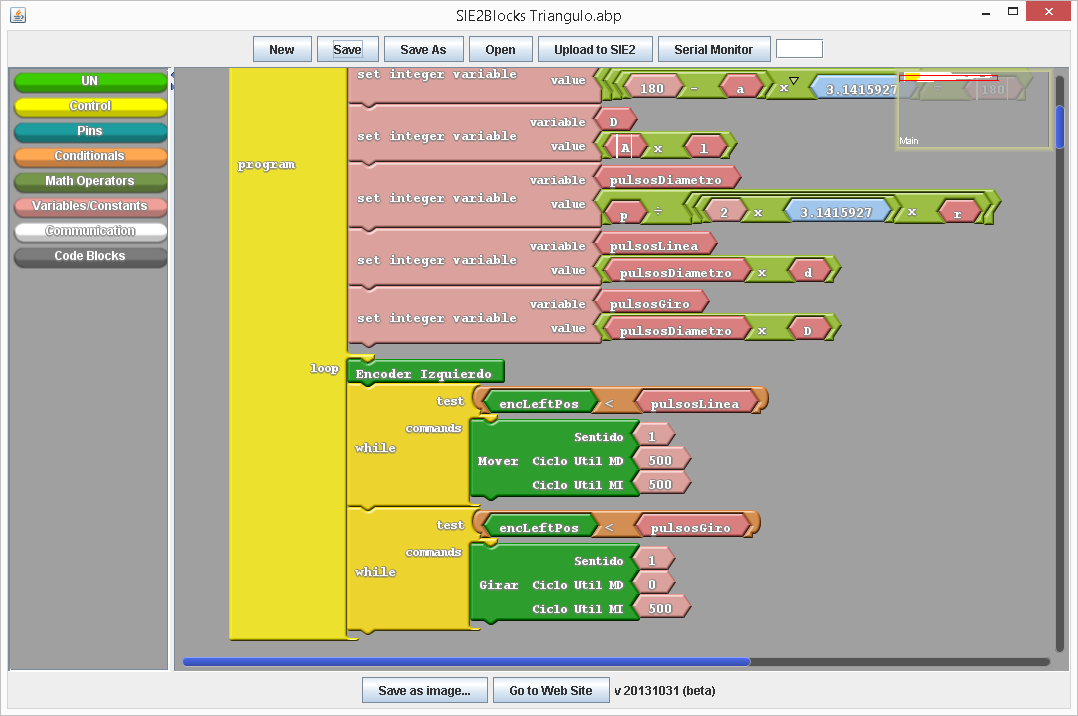
Siendo **A** el ángulo del arco de giro en radianes y **a** el ángulo de la figura que se desea seguir en grados.

Con esto ya podemos hacer la programación de la trayectoria del robot, la cual aparece en las Figuras 6 y 7.



**Figura 6. Inicialización programa trayectoria triangular**

En las Figuras 6 y 7 se puede ver la implementación de la explicación previa del cálculo de las trayectorias que debe seguir el robot y la ejecución del código necesario para hacer seguimiento a estas trayectorias, que no sufre ningún cambio respecto a la trayectoria inicial.



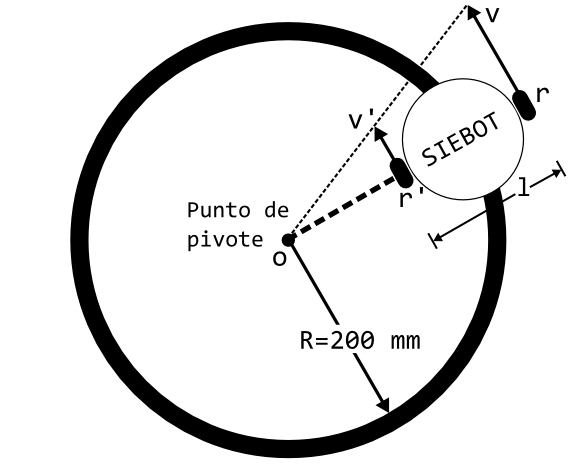
**Figura 7. Continuación inicialización y seguimiento trayectoria triangular**

Ahora solo resta ejecutar el programa en el SIEBOT y evaluar su comportamiento mientras se hace seguimiento de la trayectoria de un triángulo equilátero ¿Qué cambios habría que hacer en el programa para que siga una trayectoria triangular no equilátera?

**Trayectoria Circular**

El cálculo de la trayectoria circular es un poco diferente a las que se vienen trabajando hasta el momento, en especial si la circunferencia tiene un radio mayor a la mitad de la longitud entre las ruedas del robot, si el radio es menor, no se puede seguir esta trayectoria

Si la trayectoria circular tiene un radio igual a la mitad de la distancia entre ruedas el cálculo es simple, pues una de las ruedas se ubica en el punto de pivote y se mantiene con velocidad cero, mientras que la otra rueda gira a una velocidad que se establezca, en caso que el radio sea mayor, como en la Figura 8, ambas ruedas deben girar a una cierta velocidad, que no puede ser la misma, sino el robot iría en línea recta, pero entonces ¿Cómo puedo determina a que velocidad debe girar cada rueda? eso lo veremos a continuación.



**Figura 8. Trayectoria circular**

Primero que todo vamos a pensar que el robot gira respecto a un pivote "imaginario" ubicado en el centro de la trayectoria circular (punto **o**), pues el robot debe girar respecto a este punto para seguir la trayectoria satisfactoriamente sin estar "anclado" a un punto como lo hacía antes, cuando giraba respecto a una rueda que permanecía quieta.

Ahora debemos pensar que cada rueda, así como lleva una velocidad angular, también lleva una velocidad tangencial igual a la velocidad angular por el radio de la rueda, debemos pensar en esta velocidad tangencial como un vector, (una flecha que apunta hacía la dirección de la velocidad y cuya longitud es proporcional a la magnitud de la velocidad) así, y como se puede ver en la Figura 8, como la velocidad de la rueda externa a la trayectoria es mayor, su vector será más grande que el de la rueda en la parte interior de la trayectoria, cuya velocidad debe ser menor. De esta manera vemos como se forman dos triángulos semejantes, uno formado por los puntos **o** (pivote), **r** (rueda externa) y **v** (final del vector velocidad externo) y el otro triángulo formado por los puntos **o**, **r**' (rueda interna) y **v'** (final vector velocidad interno). Con esta explicación es muy fácil calcular la velocidad de las ruedas pues se parte de información conocida, el radio de la trayectoria, la distancia entre ruedas y la velocidad de una de las ruedas (que puede ser definida por nosotros), veamos como.

Vamos a partir del triángulo **orv**, del que conocemos todas sus medidas, para calcular el triángulo **or'v'**, del que no conocemos la medida **r'v'**. La medida **or** es igual a la suma del radio de la trayectoria **R** más la mitad de la distancia entre ruedas **l**.

**or = R + l/2**

La distancia **rv** es igual a la magnitud de la velocidad tangencial de la rueda externa, que es determinada por nosotros, por ejemplo, podemos determinar un ciclo útil para el motor externo de 500, como este ciclo útil no es una velocidad angular y mucho menos tangencial, debe multiplicarse por un factor de conversión **f** (cuyo valor es desconocido e irrelevante, porque se comparte con la otra rueda, entonces se cancelará al final de los cálculos, si el radio de las ruedas o la relación entre voltaje de alimentación y velocidad del motor fuera diferente si que importaría) que convierte las unidades de ciclo útil, con el que se alimenta el motor, a velocidad tangencial; entonces podemos decir que la velocidad tangencial del motor externo es 500f. Ahora vamos a plantear una ecuación de semejanza entre los dos triángulos.



Y despejando por sus valores

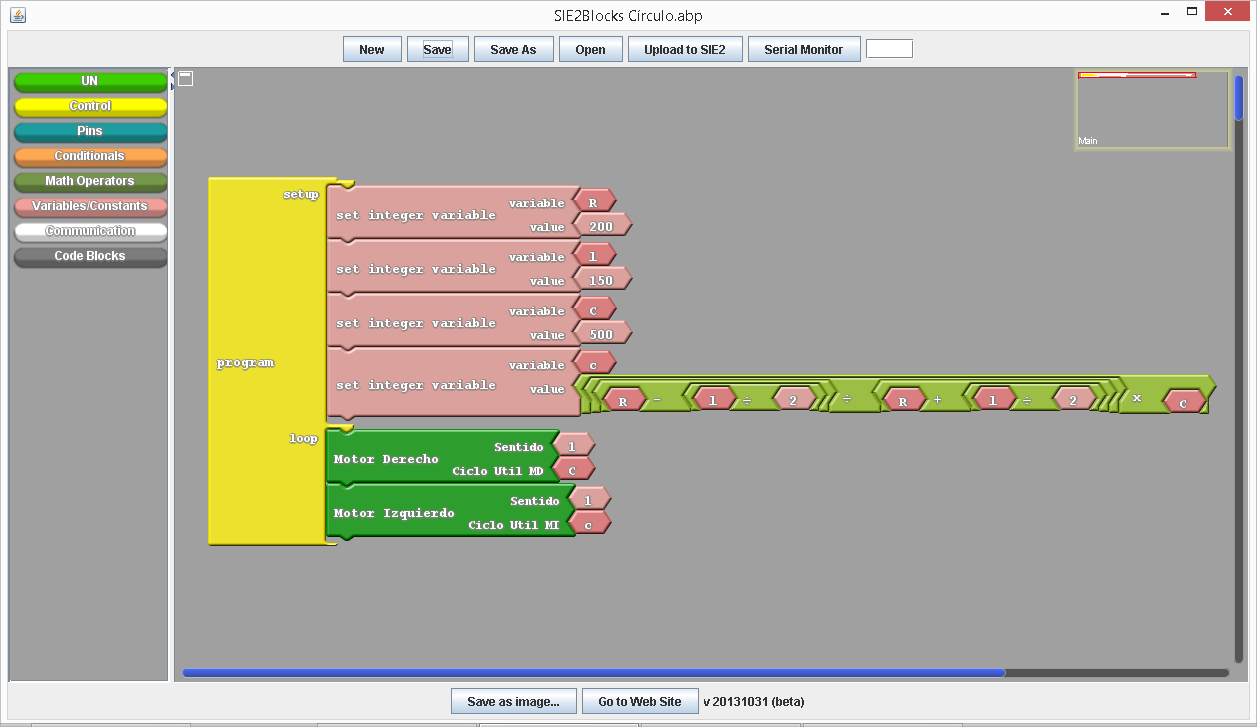


Siendo **c** el ciclo útil que debe tener el motor interno y **C** el ciclo útil del motor externo, que es lo que deseamos encontrar. Si despejamos **c** de la ecuación anterior nos dará como resultado.



Con esta relación podremos calcular el ciclo útil del motor interno **c**, que junto con un ciclo útil **C** en el motor externo podremos seguir la trayectoria circula. De la ecuación anterior podemos concluir que el ciclo útil del motor interno siempre será menor que el del motor externo, tal como se espera ¿Por qué?

En la Figura 9 podemos ver el programa que pone en práctica la explicación anterior sobre la generación de trayectorias circulares, como se ve, se define el radio de la circunferencia **R**, la distancia entre ruedas **l** y el ciclo útil del motor externo **C**, posteriormente se calcula el ciclo útil del motor interno **c**.



**Figura 9. Programa trayectoria circular**