

Práctica 3 - Introducción a la plataforma Robotino FESTO

Roberto Cadena Vega

Objetivo

Utilizar algoritmos de control automático para programar comportamientos estables de movimiento en el Robotino FESTO.

Equipo

El siguiente equipo será proporcionado por el laboratorio, siempre y cuando lleguen en los primeros 15 minutos de la práctica.

- ▶ Plataforma de desarrollo Robotino FESTO
- ▶ Computadora con el software Robotino View¹ y Robotino SIM² instalado

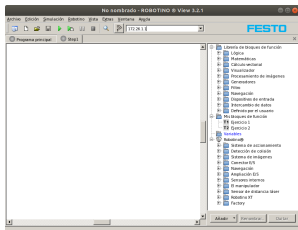
¹<https://www.festo-didactic.com/int-en/services/robotino/programming/robotino-view/?fbid=aW50LmVuLjU1Ny4xNy4zNC4xNDI2>

²<https://www.festo-didactic.com/int-en/services/robotino/simulation/?fbid=aW50LmVuLjU1Ny4xNy4zNC4xNDQy>

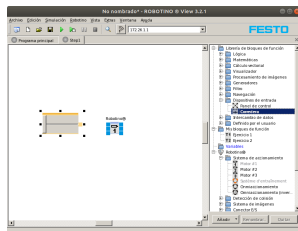
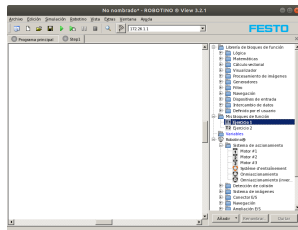
Conocimientos Previos

Uso del software de simulación

El software a utilizar en esta práctica está desarrollado por FESTO explícitamente para trabajar con el robot móvil Robotino, con el se puede programar algoritmos simples o complejos que utilicen toda la instrumentación en el robot. Empezaremos abriendo el software Robotino View, al abrir este programa se verá una pantalla como la de la figura 5. En esta ventana podemos ver del lado derecho un menú de navegación de funciones a utilizar dentro de nuestra programación.



Empezaremos con un ejemplo sencillo el cual requerirá que utilicemos el motor 1 del Robotino, función que se encuentra dentro del menú Robotino, Sistema de accionamiento, Motor # 1 como se ve en la figura 6. Una vez que logremos identificar la función tan solo tenemos que arrastrarla al area central para utilizarla dentro de nuestro programa; de la misma manera encontraremos la función Corredera dentro del menú Librería de bloques de función, Dispositivos de entrada, Corredera y la colocaremos a la izquierda del motor como se ve en la figura 6.

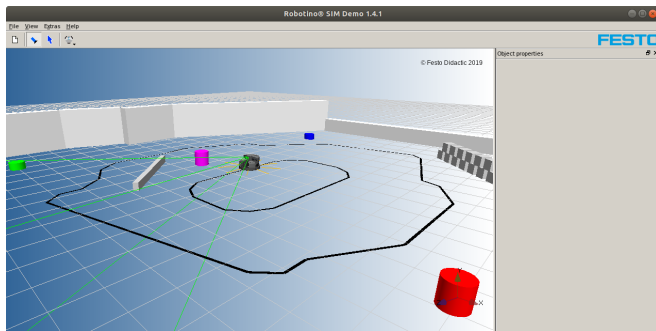




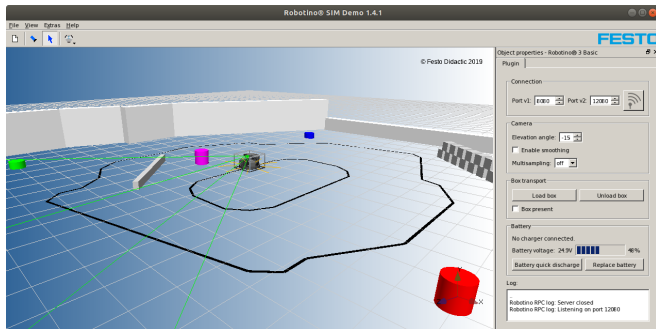
Por el momento estos elementos no tienen ninguna relación, por lo que este programa aun no haría nada, para esto necesitamos que la salida de la corredera este conectada con la entrada del motor 1, si ahora arrastras tu mouse desde el nodo de salida de la corredera al primer nodo de entrada del motor tendrás un diagrama como el de la figura 7, el cual ya es un programa valido.

Conexión al robot virtual

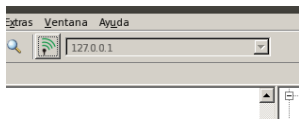
Si intentas correr este programa con el triangulo verde ubicado en la barra de herramientas de la ventana, no pasará nada, ya que no hay ningún Robotino conectado al programa; para esto empezaremos utilizando un Robotino virtual el cual este ubicado en el software Robotino SIM, si abrimos este programa podremos ver una ventana como la de la figura 8.



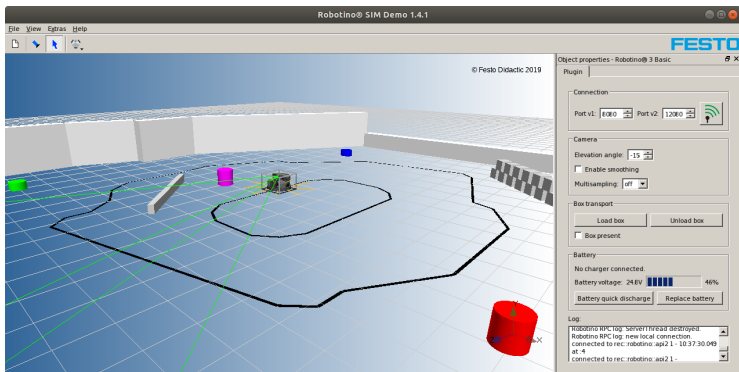
En esta ventana podemos ver el estado del robot virtual, así como el puerto de conexión que en este caso es 8080 (dependiendo de la versión del software necesitarás estos datos) como se ve en la figura 9.



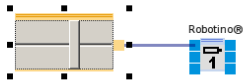
Para poder conectar nuestro ambiente de programación con el robot virtual tenemos que escribir la dirección IP en la cual el servidor del robot esta funcionando, ya que nuestro robot es virtual, esta dirección será la misma siempre 127.0.0.1, en algunas versiones del software es necesario agregar el puerto en el cual el robot esta escuchando las instrucciones, en este caso sería 127.0.0.1:8080. Si ahora damos clic sobre el icono de conexion de la izquierda tendremos el robot conectado como en la figura 10



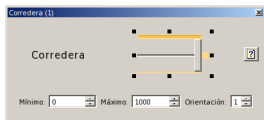
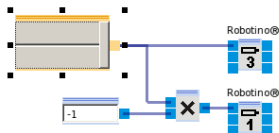
Si ahora damos clic sobre el triangulo verde nuestro programa correrá dentro del robot virtual en tiempo real, como el de la figura 11.



Sin embargo nuestro robot no se mueve, no es hasta que movemos nuestra corredera como en la figura 12, que el robot empezará a moverse.



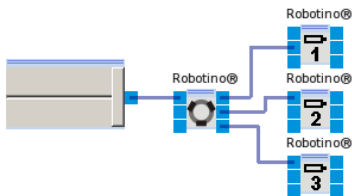
Movimiento del robot



Por el momento nuestro robot se ve muy triste, ya que solo puede mover uno de sus motores, sin embargo esto lo podemos resolver rapidamente al hacer funcionar un motor en un sentido y otro en el contrario, con un diagrama como el de la figura 13. Si ademas modificamos el limite máximo de nuestra corredera haciendo clic secundario en la parte superior para llevarlo hasta 1000 como en la figura 13, podremos hacer mover nuestro Robotino mas agilmente.

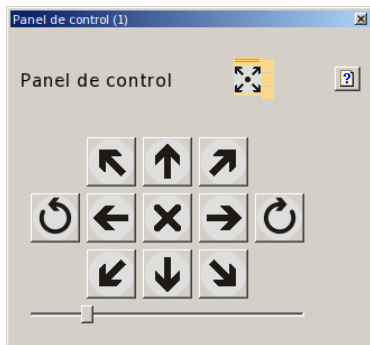
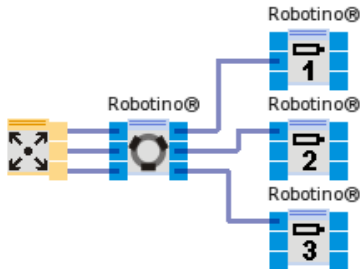
La razón por la que esto funciona para mover el Robotino hacia adelante, es que la configuración de motores del robot es como el de la figura 14 en donde cada motor esta acoplado a una rueda Ilon como la de la figura 14, la cual es capaz de moverse en cualquier dirección. Las dos ruedas traseras son las que se controlan con los motores 1 y 3, y cuando se mueven en sentidos contrarios se puede mover el robot hacia adelante o hacia atras.





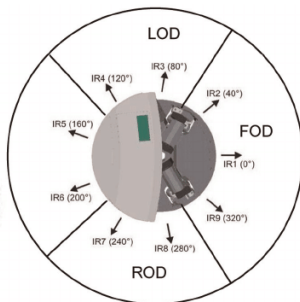
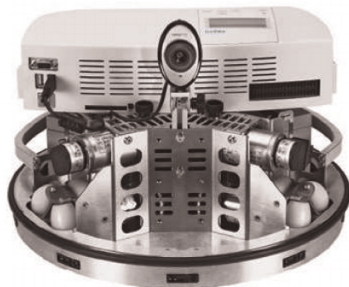
Sin embargo existe una función que se encarga de hacer todos los calculos necesarios para mover el Robotino de manera adecuada, si usamos el bloque Omniaccionamiento como en la figura 15, podemos obtener el mismo resultado.

O bien utilizar un panel de control y mover el Robotino en todos sus ejes.

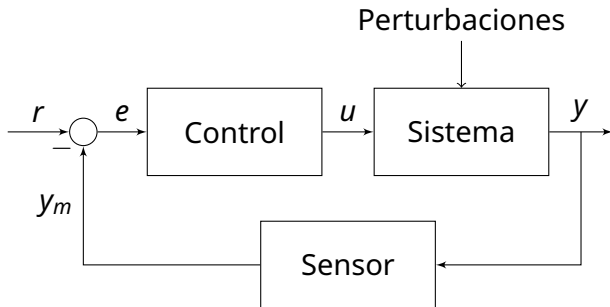


Control

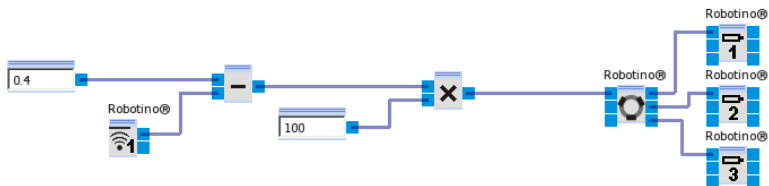
Si lo único que quisiéramos es un carrito a control remoto, podríamos parar en este momento, sin embargo la plataforma esta diseñada para desarrollar algoritmos de control con los diferentes sensores que existen en el Robotino. Empezaremos utilizando los sensores infrarojos de distancia que se encuentran en el robotino como muestra la figura 17.



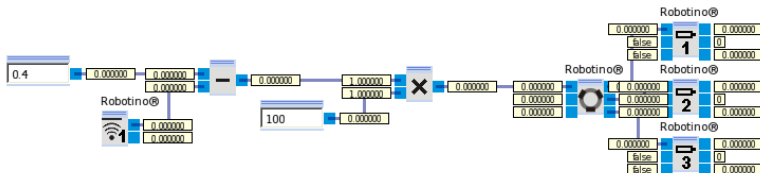
Para esto debemos implementar una ley de control con realimentación en lazo cerrado, es decir utilizaremos los sensores del Robotino para medir la distancia del Robotino a algún objeto y buscaremos que esta distancia disminuya hasta llegar a nuestro objetivo. El diagrama general de este sistema se verá como en la figura 18



En este caso nuestros sensores son los sensores infrarojos de distancia, estos nos darán una señal medida la cual es restada a una referencia constante, el resultado se multiplica por la ley de control, en este caso una constante con valor 100, y el resultado de esto se mete al sistema. el diagrama de control del Robotino se verá como el de la figura 19



Si queremos ver los valores de estos calculos en tiempo real, podemos encender la opción de Vista de valores en el menú Vista como en la figura 20



Desarrollo

Diseña un algoritmo de control para el Robotino FESTO, de tal manera que el Robotino se mueva lateralmente y se pare en cuanto este a aproximadamente 30cm (el tamaño aproximado del Robotino) de la pared. Valida tu programación en simulación y corre el programa en el Robotino

Conclusiones

El alumno deberá describir sus conclusiones al final de su reporte de práctica.