

Práctica 4 - Sensores y actuadores

Roberto Cadena Vega

Objetivo

Utilizar algoritmos de control automático y un sistema de visión artificial 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

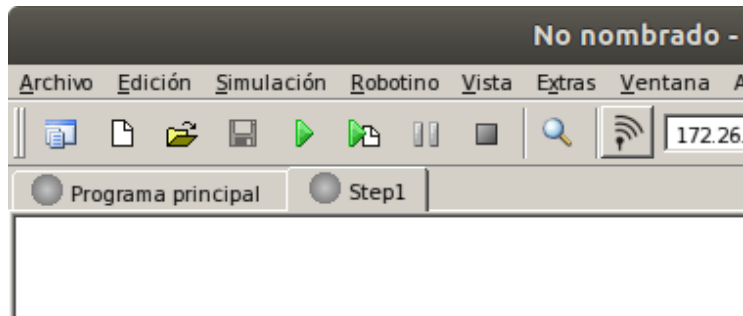
Introducción

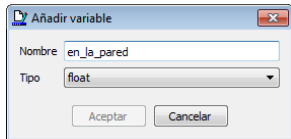
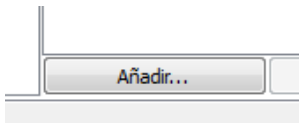
En la práctica anterior programamos un algoritmo de control automático para controlar la posición del Robotino relativa a un objeto cercano, por medio de los sensores infrarojos de distancia.

Sin embargo este algoritmo desarrollaba solo una función, si queremos programar comportamientos complejos en el Robotino necesitamos aprender la capacidad de ejecutar diagramas de tipo GRAFCET; lo que nos permitirá ejecutar diferentes programas en diferentes condiciones.

Control secuencial

Hasta el momento hemos utilizado solo la pestaña que esta marcada como Step 1, para diseñar el diagrama tipo GRAFCET con la secuencia de pasos a ejecutar tenemos que entrar en la pestaña de Programa Principal (Main).

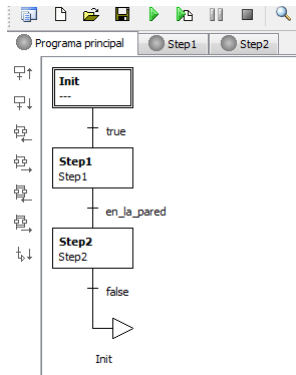




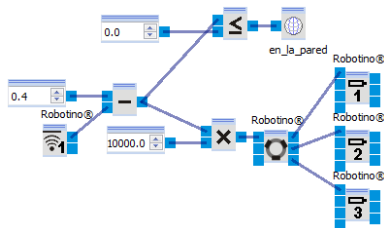
Nombre	Tipo	Valor
en_la_pared	float	0.000000

El primer paso es definir una variable, la cual utilizaremos para determinar en que momento debemos cambiar de un programa a otro; en este caso se llamará `en_la_pared`, será de tipo flotante y tendrá un valor de 0.

Una vez que tenemos nuestra variable, podemos hacer nuestro diagrama de GRAFCET el cual, en un principio, tiene solo el programa Step1 y agregaremos nosotros el programa Step2 con el boton de agregar bloque abajo estando seleccionado el primer programa. Ademas modificaremos la condición necesaria para cambiar del primer programa al segundo, colocando el nombre de la variable que definimos.

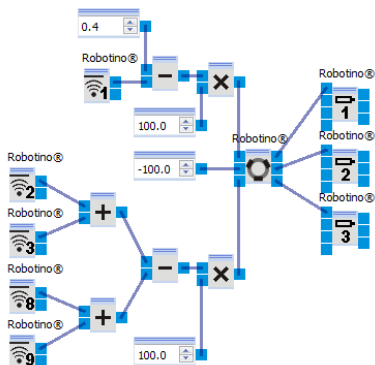


Ahora tenemos que programar los algoritmos de control de cada paso, empezando en el programa Step1, el cual es muy similar al de la práctica 3, pero agregamos una comparación para saber cuando el Robotino ya ha llegado a la pared, lo cual hará que la variable `en_la_pared` tenga un valor verdadero cuando el Robotino llegue a la pared.

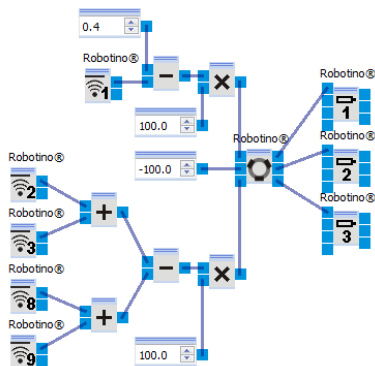


Y en el programa 2 ponemos el mismo comportamiento para el movimiento hacia adelante y hacia atras, pero agregamos un algoritmo de control para el movimiento rotacional sobre el eje del Robotino; recordando que los sensores 2 y 3 estan al frente y a la izquierda del Robotino y los sensores 8 y 9 estan al frente y a la derecha del Robotino, la operación que estamos realizando se reduce a

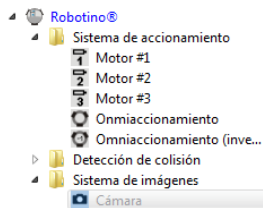
$$\dot{\theta} = (S_2 + S_3 - S_8 - S_9) \cdot 100$$



El movimiento lateral simplemente se mueve en una dirección todo el tiempo, de tal manera que el comportamiento final será que el Robotino tratará de seguir la pared a una distancia definida, girando cuando se encuentre algún obstáculo. En este punto, deberás guardar tu programa y validarlo con la plataforma Robotino en la vida real.



Visión artificial

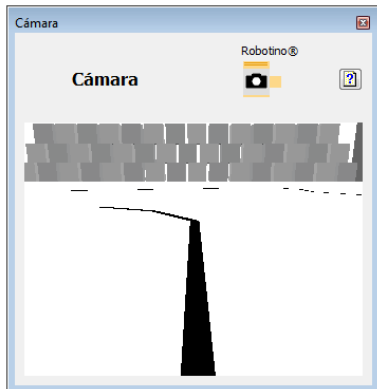


Robotino®

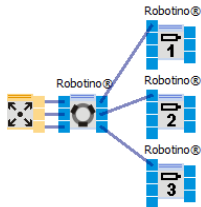


Ya que entendemos como crear comportamientos complejos para nuestro Robotino el siguiente paso es agregar un nuevo sensor a nuestro arsenal de automatización; en un programa completamente nuevo agrega el elemento Camara que se encuentra en Sistema de imagenes.

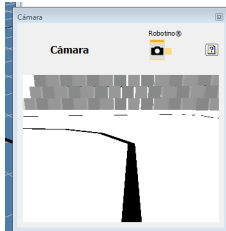
Si corremos este programa con solo la camara en el, y hacemos doble clic en el icono de la función, podremos tener una ventana con las imagenes de la camara frontal del Robotino.



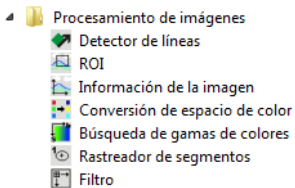
Robotino®

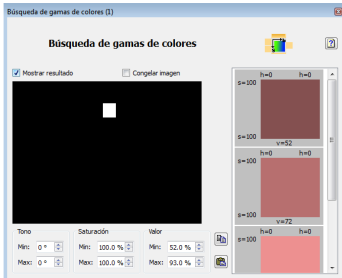
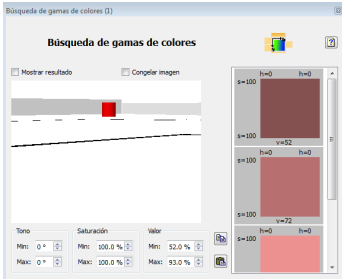


O bien, si agregamos el control de motores a nuestro programa, podremos manipular el Robotino mientras vemos las imágenes de su cámara frontal



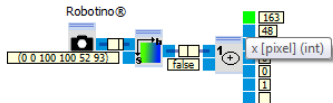
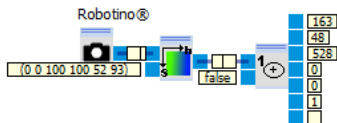
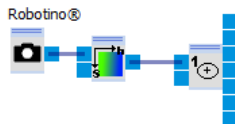
Pero las imágenes de la cámara por si solas no son suficientes para manipular el comportamiento de nuestro Robotino, tenemos que hacer un procesamiento de los datos para obtener información relevante, por ejemplo, para encontrar objetos con un cierto color en nuestro ambiente. Ahora agregaremos la función de Búsqueda de gamas de colores, que se encuentra en Procesamiento de imágenes.

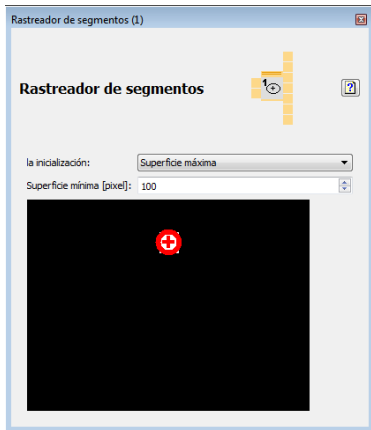




Si abrimos la función recién insertada, tendremos una ventana en donde podemos escoger la gama de colores que queremos aislar, una vez seleccionados los colores debemos ver como todo el resto de la imagen se vuelve negra si seleccionamos mostrar resultado.

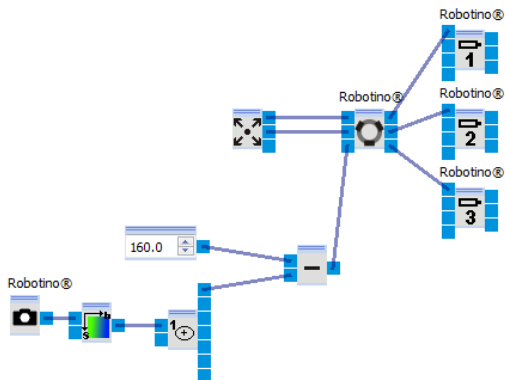
Agregando ahora el Rastreador de segmentos, podremos ver como este nos va dar información importante de las imagenes obtenidas, lo mas importante, la posición en el eje x.



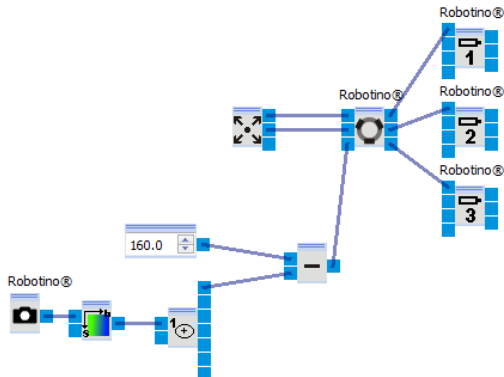


Analizando esta información, el valor que obtenemos como primer salida de la función Rastreador de segmentos es la posición del centroide del objeto detectado dentro de la imagen.

Si aplicamos un algoritmo de control a este parametro, queriendo que el objeto encontrado este en el centro de la imagen (la imagen tiene 320 pixeles de ancho, por lo que el centro es la posición 160), podremos construir un diagrama como el de al lado.



El resultado de este algoritmo de control es que una vez que el Robotino ha encontrado un objeto con el color que queremos, cuando movamos el Robotino lo hará siempre viendo hacia ese objeto. En este punto, deberás guardar tu programa y validarlo con la plataforma Robotino en la vida real.



Desarrollo

- ▶ Valida los programas requeridos en el area de Conocimientos previos
- ▶ Diseña un conjunto de programas de tal manera que el Robotino gire sobre su eje hasta encontrar un objeto del color definido por ti y una vez que lo encuentre se mueva hacia el y pare a aproximadamente 30cm de el. Valida tu programación en simulación y en la vida real.

Conclusiones

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