

Robot de bolas



El robot de bolas es capaz de disparar pequeñas bolas que están en un cargador. ¡Es muy divertido! Con este modelo también aprenderás los fundamentos de la programación. Ya conoces los componentes fischertechnik utilizados de las tareas anteriores. Controlarás el modelo con el sensor de gestos.

Actuadores y accesorios técnicos instalados en el modelo:

| Mini motor | Caja de cambios | Sensor gestual | Omnivheel |
|---|---|--|---|
|  |  |  |  |

Encontrará una explicación de los componentes en la página de inicio.

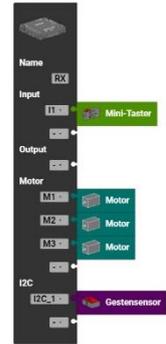
El modelo de "robot bola" se divide en 3 tareas de programación:

| | |
|------------------------------------|--|
| Tarea 1 Bola_robot_1.rpp | <u>Programación nivel 3</u> La orden de disparo debe activarse mediante control gestual |
| Tarea 2 Bola_robot_2.rpp | <u>Programa de nivel 3</u> El control gestual se utiliza para dirigir el robot balón en distintas direcciones. |
| Tarea 3 Bola_robot_3.rpp | <u>Programa de nivel 3</u> El control gestual se utiliza para dirigir el robot balón en distintas direcciones. La orden de disparo se activa según sea necesario. |

Importante: Al programar las tres tareas modelo, comience con un "nuevo proyecto" llamado "Ball_Robot_1".

A continuación, debe configurar el controlador.

Los actuadores y sensores se encuentran en las instrucciones de montaje. Los motores 1 y 2 controlan el accionamiento, el motor 3 el lanzador. El minipulsador sirve de final de carrera en el lanzador. El sensor de gestos detecta los movimientos. Estos se analizan y se convierten en instrucciones de dirección de desplazamiento.



Tarea 1

Para resolver la primera tarea, no necesita acceder a ninguna función de tareas anteriores, por lo que puede empezar por el programa principal.

Importante: He aquí algunas notas técnicas sobre la función de disparo. Las bolas se almacenan en un cargador y desde allí se introducen individualmente en el lanzador.



La barra en I verde se dobla hacia atrás mediante el mecanismo de manivela, el engranaje del motor de lanzamiento. El motor gira hacia la izquierda hasta el botón y libera el puntal I, que se dobla bajo tensión. Esto catapulta la pequeña bola hacia delante fuera del lanzador con su energía de tensión.



Como ya ha trabajado con el sensor de gestos en tareas anteriores, sabe que debe activarse inmediatamente después de iniciar el programa.

Seleccione "Gesto" para seleccionar el gesto. Establezca el valor del gesto en "= 1". Esto significa que todas las direcciones de movimiento se analizan como una sola dirección y el balón sale disparado. A continuación, se activa el bucle "Repetir continuamente".



Introduzca el comando "if make" en el bucle. En "if", consulta el sensor de gestos para ver si se ha determinado el valor "=1". Cambia también el comando por "gesture". Ahora la secuencia de disparo es la siguiente

El motor de cocción "RX_M3" gira "0,3" segundos hacia la izquierda. Esto significa que el soporte de cocción se coloca después del botón y gira hasta que se pulsa el botón.

Complete el programa principal utilizando la ilustración del programa adyacente.

```

Programmstart
  Gestensensor RX_I2C_1 - aktiviere - Geste -
  dauerhaft wiederholen
  mache + falls
  mache
    ist Gestensensor RX_I2C_1 - Geste - = - 1
    setze Motor RX_M3 - links - Geschwindigkeit - 512
    warte s - 0.3
    ist Mini-Taster RX_I1 - geschlossen -
    setze Motor RX_M3 - rechts - Geschwindigkeit - 512
    warte s - 0.15
    stoppe Motor RX_M3 -
  
```

A continuación, el motor se desplaza "0,15" segundos hacia la derecha -es decir, hacia atrás- y vuelve a soltar el botón. A continuación, el motor se detiene.

```

Programmstart
  Gestensensor RX_I2C_1 - aktiviere - Geste -
  dauerhaft wiederholen
  mache + falls
  mache
    ist Gestensensor RX_I2C_1 - Geste - = - 1
    setze Motor RX_M3 - links - Geschwindigkeit - 512
    warte s - 0.3
    ist Mini-Taster RX_I1 - geschlossen -
    setze Motor RX_M3 - rechts - Geschwindigkeit - 512
    warte s - 0.15
    stoppe Motor RX_M3 -
  
```

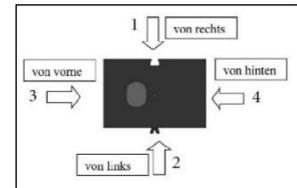
El programa vuelve al inicio del bucle y espera una nueva activación del sensor gestual para disparar una pelota.

El programa de lanzamiento ya ha concluido.

Prueba el programa y guarda **Ball_robot_1** en tu ordenador.

Tarea 2

Importante: En la imagen de la derecha, los movimientos se definen cuando la mano se mueve a través del sensor de gestos. La ilustración se refiere a la dirección de instalación en el modelo.



¿Qué valores proporciona el sensor de gestos cuando se realizan los siguientes movimientos?

3 - por delante, 4 - por detrás, 1 - por la derecha, 2 - por la izquierda

Antes de empezar a programar varias funciones, borra todos los comandos del bucle del programa "Ball_robot_1".

```

Programmstart
  Gestensensor RX_I2C_1 - aktiviere - Geste -
  dauerhaft wiederholen
  mache
  
```

Guarda el programa bajo **Ball_robot_2**.

A continuación, cree las siguientes funciones para las direcciones de conducción del modelo.

```

+ definiere mv_fw
  setze Motor RX_M1 - links - Geschwindigkeit - 256
  setze Motor RX_M2 - links - Geschwindigkeit - 256
  warte s - 0.5
  
```

```

+ definiere turn_left
  setze Motor RX_M1 - rechts - Geschwindigkeit - 256
  setze Motor RX_M2 - links - Geschwindigkeit - 256
  warte s - 0.5
  
```

```

+ definiere mv_bw
  setze Motor RX_M1 - rechts - Geschwindigkeit - 256
  setze Motor RX_M2 - rechts - Geschwindigkeit - 256
  warte s - 0.5
  
```

```

+ definiere turn_right
  setze Motor RX_M1 - links - Geschwindigkeit - 256
  setze Motor RX_M2 - rechts - Geschwindigkeit - 256
  warte s - 0.5
  
```

Una vez definidas las funciones, debe crear una variable "gesto". En esta variable del programa se lee siempre el valor actual del sensor de gestos. A continuación, este valor se utiliza para realizar diversas tareas dentro de una consulta "if do ...".

Verarbeitung - Primero inserte el comando "set ... to" del bloque "Proceso" - "Variable" en el área libre del bucle.

Sensoren - Añade el comando "get gesture sensor ..." del bloque "Sensores" - "I2C" al punto de acoplamiento. Cambia la entrada "HEX" por "Gesture".

El siguiente paso consiste en consultar el valor determinado de la variable. Para ello, utilice el comando "si no ..." del bloque "Procesamiento" - "Lógica". Inserte el comando "es cierto ..." en las posiciones "si". Inserte la variable "gesto" en la primera posición libre y el comando "un número" en la segunda posición. El movimiento debe ser "de frente", por lo que el valor debe ser "3".

Haga clic en el signo "+" para ampliar la consulta. Como se necesitan 3 consultas más, tiene que hacer clic tres veces. Duplique la primera consulta "if" e insértela en la posición correspondiente. Cambia el valor "=" de la 2ª consulta a "4" para "atrás", el valor "=" de la 3ª consulta a "2" para "izquierda" y el valor "=" de la 4ª consulta a "1" para "derecha".

Si ya ha programado las funciones para los movimientos, ahora puede insertarlas en los espacios vacíos de "make". Puede encontrarlas en "Procesamiento" - "Funciones".

Pero oops, algo se ha olvidado. Todavía necesita una función de "parada" para los dos motores.

```

+ definiere stop
  stoppe Motor RX_M_M1
  stoppe Motor RX_M_M2

```

Defina la función adyacente e insértela en el programa principal.

```

Programmstart
  Gestensensor RX_I2C_1 aktiviere Geste
  dauerhaft wiederholen
  mache setze gesture auf hole Gestensensor RX_I2C_1 Geste
  + falls gesture = 3
  mache mv_fw
  sonst falls gesture = 4
  mache mv_bw
  sonst falls gesture = 2
  mache turn_left
  sonst falls gesture = 1
  mache turn_right
  sonst stop

```

Esto resuelve la tarea 2. Vuelve a guardarlo en tu ordenador.

A continuación, prueba el programa en tu modelo.

Tarea 3

Puede utilizar la "Tarea 2" como base para el programa. Antes de adaptar el programa principal a la tarea, cree la función "incendio".

Guarda el programa con el nombre **Ball_robot_3**.

```

+ definiere fire
  setze Motor RX_M3 links Geschwindigkeit 512
  warte s 0.3
  warte bis ist Mini-Taster RX_I1 geschlossen
  setze Motor RX_M3 rechts Geschwindigkeit 512
  warte s 0.15
  stoppe Motor RX_M3

```

Consejo: Puede copiar la función de la tarea 1 y pegarla en una nueva función. Sólo tiene que ajustar los parámetros de esta función.

Ahora al programa principal. Borra todos los comandos del bucle. Inserta el comando "set to" y el comando "a number". Ajuste el valor a "0".

```

Programmstart
  Gestensensor RX_I2C_1 aktiviere Geste
  dauerhaft wiederholen
  mache setze gesture auf 0

```

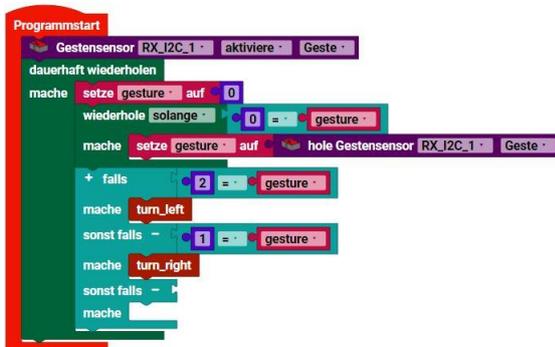
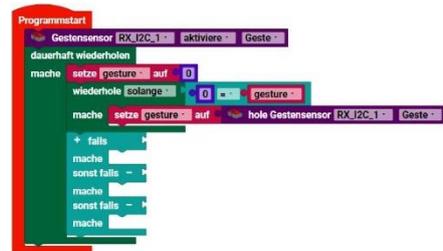
```

Programmstart
  Gestensensor RX_I2C_1 aktiviere Geste
  dauerhaft wiederholen
  mache setze gesture auf 0
  wiederhole solange 0 = gesture
  mache setze gesture auf hole Gestensensor RX_I2C_1 Geste

```

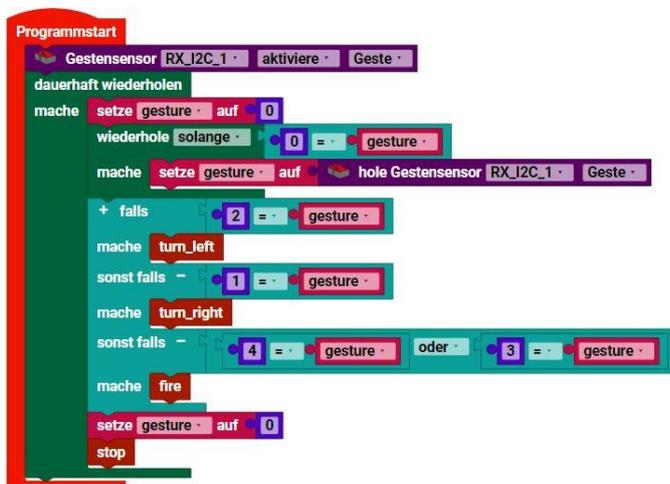
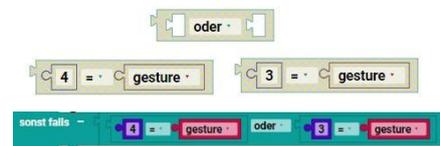
Se necesita otro bucle que se ejecute mientras el valor del sensor sea "=". A continuación, se determina el valor del sensor y se asigna a la variable "gesto".

A esto le sigue la evaluación con el comando "if do". Inserta esto después del comando loop y extiéndelo a un total de 3 consultas.



La consulta 1 (si gesto = 2) desplaza el modelo hacia la izquierda. La consulta 2 (si gesto = 1) desplaza el modelo hacia la derecha. Añade los comandos correspondientes a los puntos de acoplamiento.

Ahora pasamos al último punto de acoplamiento. Aquí debe consultar si el valor del sensor "4" es "0" el valor del sensor "3". Si la consulta es correcta, debe activarse el comando de disparo. El comando de consulta se compone como se muestra en la imagen.



Inserte la secuencia de comandos y, a continuación, la función "disparar" en la zona "hacer".

Después de las consultas, vuelve a poner la variable "gesto" a "0" y para los motores. Una vez procesados todos los comandos, el programa salta de nuevo al inicio del bucle y espera otro movimiento.

Hasta aquí no ha sido tan difícil.

Prueba el programa y guárdalo de nuevo en tu ordenador.

A continuación, desmonta el modelo y clasifica los componentes en tu caja de pies.