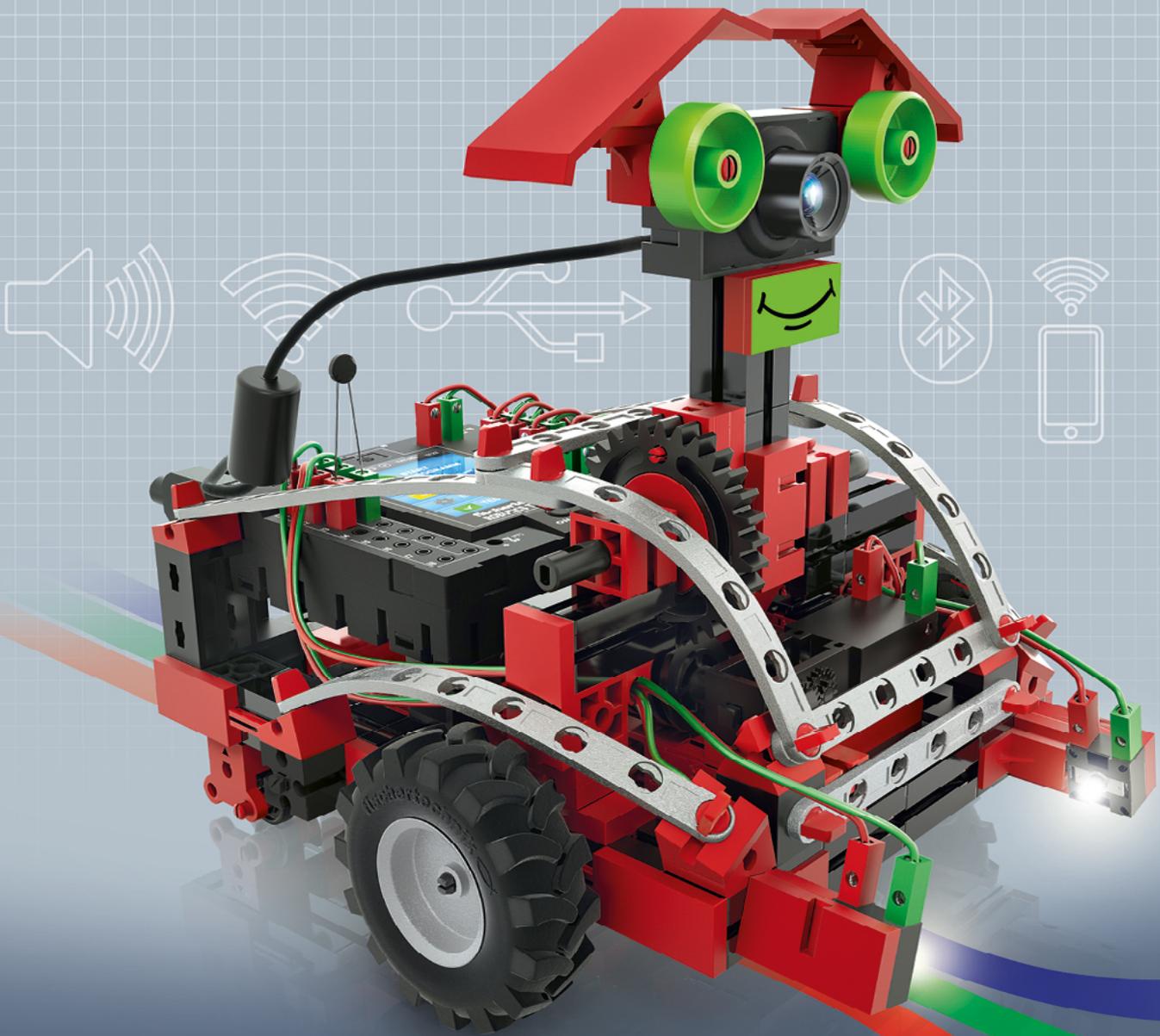




**fischertechnik** 

**ROBOTICS**



**TXT** *Discovery Set*

**14** MODELS  
MODELS

<b>Bienvenido al mundo ROBOTICS de fischertechnik</b>	<b>4</b>
<b>Algunas informaciones generales</b>	<b>5</b>
Electricidad	5
Sobre este cuaderno adjunto	5
¿Robot, el ser humano artificial?	6
ROBOTICS, (casi) todo automático	7
<b>Explicaciones sobre componentes</b>	<b>8</b>
Motores de codificador	8
Motor XS	9
LEDs	9
Lámpara de lente	9
Fototransistor	10
Pulsadores	10
Sensor de calor (NTC)	11
Módulo de cámara	11
ROBOTICS TXT Controller	12
<b>Algunas sugerencias</b>	<b>13</b>
<b>Primeros pasos</b>	<b>14</b>
<b>Modelos de iniciación</b>	<b>15</b>
Semáforo para peatones	15
Secador de manos	16
Regulador de temperatura	18
Barrera	19
"Operador de cámara"	20
Cámara basculante	22

**Robots móviles – el siguiente desafío** **23**

---

Robot desplazable	23
Detector de obstáculos	27
Detector de obstáculos con cámara	28
Rastreador	30
Robot de reconocimiento	33
Robot de fútbol con control de movimiento	35
Robot de fútbol	38

**Búsqueda de errores** **43**

---

Prueba de interfaz	43
Cables y cableados	43
Contacto flojo	43
Cortocircuitos	44
Alimentación de corriente	44
Error en el programa	44
Función de la cámara	44
Los últimos comodines	45

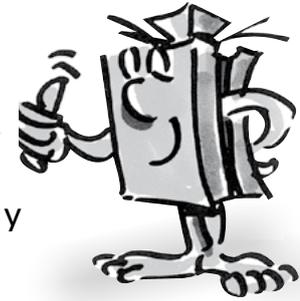
**¿Y cómo sigue ahora?** **46**

---

## Bienvenido al mundo ROBOTICS de fischertechnik

¡Hola!

Nos alegramos que te hayas decidido por el kit de construcción "ROBOTICS TXT Discovery Set" de fischertechnik. Con este kit de construcción puedes realizar una cantidad de experimentos interesantes y solucionar atrayentes tareas.



Al leer este cuaderno de pantalla y probar las tareas y los experimentos, aprenderás paso a paso como con el ROBOTICS TXT Controller de fischertechnik se pueden controlar y programar sencillas y también complejas máquinas y robots.

Como sucede al aprender, no se puede comenzar inmediatamente con las cosas más difíciles, aun cuando estas naturalmente con frecuencia son un poco más interesantes que las algo más sencillas. Por eso hemos estructurado los experimentos y las tareas en este cuaderno de tal manera, que con la nueva tarea aprendes algo más, que puedes a su vez aplicar en la siguiente tarea.

Entonces, no temas, comenzamos modestamente y juntos avanzaremos hasta los robots grandes.

Ahora te deseamos mucho éxito y satisfacción al experimentar con el ROBOTICS TXT Discovery Set.

Tu equipo de

**fischertechnik** 

## Algunas informaciones generales

Antes que podamos empezar a hacer algo con el kit de construcción, tienes que saber aún algunas cosas. Los componentes con los que trabajaremos, si bien son robustos pero si no los tratamos correctamente, bajo ciertas circunstancias puede ser dañados.

## Electricidad

Como seguramente sabes, muchos de los componentes del ROBOTICS TXT Discovery Set funcionan con corriente eléctrica. En asuntos que tienen que ver con la corriente, se debe observar especialmente, de no cometer ningún error. Por esta razón observa siempre exactamente las instrucciones de construcción, cuando se trata del cableado de componentes eléctricos.

De manera alguna puedes conectar el polo positivo y el negativo entre sí, o sea ponerlo en cortocircuito. De este modo el ROBOTICS TXT Controller, o también el acumulador, puede dañarse.

Electricidad y electrónica es un tema tan interesante como la robótica (o sea, de lo que se trata en este kit de construcción), y fischertechnik dispone de un kit de construcción que se ocupa de este tema. Si te interesas por ello, tendrás tantas satisfacciones con el kit de construcción "PROFI Electronics" como con el ROBOTICS TXT Discovery Set.



## Sobre este cuaderno adjunto

Este cuaderno adjunto PDF posee algunas funciones que en un cuaderno impreso no existen y que seguramente ya conoces de Internet.

- **Vínculos dentro del cuaderno**

Cuando se menciona algo en el texto que en otro punto del cuaderno está explicado con mayor exactitud (por ejemplo los componentes), este texto está escrito en [color azul oscuro y subrayado](#). Puedes hacer clic en el texto y de ese modo hojear automáticamente a la página en donde está la explicación. A esto se le llama una "referencia cruzada".

- **Informaciones en segundo plano**

En parte existen en este cuaderno conceptos o extranjerismos que pueden ser necesarios para la explicación. Estos conceptos están escritos en verde y subrayados. Si tocas el texto con el puntero del ratón, aparece un campo con una explicación.

- **Vínculos fuera de este cuaderno**

Para algunos vínculos necesitas una conexión a Internet (por ejemplo a la página de Internet de fischertechnik), o bien un ROBO Pro instalado (para la llamada a la ayuda ROBO Pro). Estos vínculos están escritos en azul claro y subrayados.

- **Imágenes**

Una imagen dice más que 1000 palabras. Esta frase seguramente ya la has escuchado alguna vez. Como tiene mucho de verdad, tocando las palabras escritas en marrón y subrayadas puedes activar una imagen, sobre la que podrás reconocer que es lo que el texto quiere decir.

- **El símbolo ROBO Pro**

Este lo encontrarás siempre cerca de tareas. Esto es conveniente, ya que en el momento que hagas clic sobre él se abre un programa de ejemplo adecuado con una posible solución.

Leeme.rpp

Todos los programas de ejemplo los encontrarás también bajo **C:\Archivos de programa (x86)\ROBOPro\Programas de ejemplo\ROBOTICS TXT Discovery Set**.

## ¿Robot, el ser humano artificial?

¿En qué piensas cuando escuchas la palabra "robot"? ¿Has visto ya alguna vez un robot? ¿En el cine o en la televisión? ¿O quizás ya a

Existen innumerables tipos diferentes de robots.

Algunos se parecen un poco a un humano, otros sólo están constituidos de uno o varios brazos. ¿Qué es entonces lo que hace que un robot sea lo que es?

En el diccionario dice: "Robots son máquinas fijas o móvil que cumplen tareas prefijadas de acuerdo a un determinado programa."



## ROBOTICS, (casi) todo automático

Robots entonces son máquinas, las cuales están controladas a través de un programa. Este control de máquinas (o en nuestro caso modelos) los llamamos "ROBOTICS".

Con el "ROBOTICS TXT Discovery Set" puedes acceder de forma fantástica a este tema. Porque el kit de construcción contiene todo lo que necesitas para la construcción y control de muchas máquinas diferentes.

Los programas para el control de los modelos los puedes confeccionar tú en el PC, con ayuda del software ROBO Pro 4.0 (o superior) y a continuación a través de una conexión USB, Bluetooth o WLAN transmitirlos al [ROBOTICS TXT Controller](#). El Controller "controla" y comanda entonces el modelo de acuerdo a la programación que tú has creado.

## Explicaciones sobre componentes

### Todo está dentro del kit de construcción

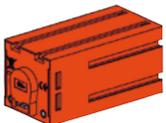
Primero encontrarás numerosos elementos fischertechnik, además de motores, lámparas y sensores, así como unas coloridas instrucciones de construcción de los diferentes modelos.

Cuando hayas desembalado todos los elementos, debes montar primero algunos componentes antes que puedas empezar (p.ej. cables y conectores). Cuales son exactamente, está descrito en las instrucciones de construcción bajo "Sugerencias de montaje". Realiza esto de preferencia inmediatamente como primero.

### Actores

Se denominan actores a todos los componentes que pueden ejecutar una acción. Esto significa, cuando se los conecta a una corriente eléctrica, de alguna forma se tornan "activos". En la mayoría de los casos, esto se puede ver directamente. El motor gira, una lámpara se enciende, etc.

### Motores de codificador



Como accionamiento para nuestros robots empleamos dos motores codificadores contenidos en el kit de construcción. A primera vista son motores eléctricos normales, que están dimensionados para una tensión de 9 Voltios y un consumo de corriente de máximo 0,5 Amperios.

Sin embargo los motores codificados pueden aún más: adicionalmente a la conexión para el suministro de corriente del motor, aún tienen una hembrilla para un cable de conexión de 3 polos, a través de la cual, con ayuda de los así llamados codificadores, se puede evaluar el movimiento de rotación del motor.

El codificador funciona de forma similar como el tacómetro de una bicicleta. Un imán (en la bicicleta se encuentra en la mayoría de los casos en uno de los rayos) pasa a cada vuelta junto a un sensor (en la bicicleta en la mayoría de los casos fijado en la horquilla), con lo que el sensor genera un impulso. Estos impulsos pueden ser contados, y para el ejemplo del tacómetro ser multiplicados por el perímetro del neumático. De este modo se obtiene el trecho recorrido.

Los codificadores en los motores de codificador fischertechnik generan por rotación del árbol del motor, tres impulsos. Además, como los motores

de codificador tienen un engranaje con una [relación de transmisión de 21:1](#) (dígase "21 a 1"), una vuelta del árbol que proviene del engranaje, 63 impulsos del codificador.

## Motor XS



El motor XS es un motor eléctrico, que es tan largo y tan alto como un elemento fischertechnik. Además es sumamente ligero. De este modo lo puedes montar en lugares, en los cuales no hay lugar para los motores grandes.

El engranajes, que asimismo se adjunta al kit de construcción, se adaptan exactamente al motor XS.

El motor XS está dimensionado para una tensión de alimentación de 9 Voltios y un consumo de corriente de máximo 0,3 Amperios.

## LEDs



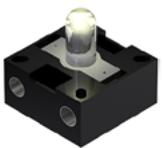
Dos LEDs están contenidos en el kit de construcción. Estos pueden ser empleados de forma muy versátil. Por ejemplo como luces de señalización en un semáforo, como luz intermitente en un robot o para una mejor iluminación de la imagen que suministra la cámara, que a su vez está incluida en el kit de construcción.

Los LEDs están dimensionados para una tensión de 9 Volt y consumen aprox. 0,01 Amperios de corriente.

### Atención:

**En la conexión del LED al suministro de corriente debes observar la correcta polaridad. El polo positivo debe ser conectado a la marcación roja en el LED.**

## Lámpara de lente



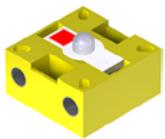
En esta lámpara incandescente se ha incorporado una lente, que reúne la luz en un haz. Presenta un aspecto similar al LED. Tienes que cuidar, que no los confundas. En la lámpara de lente la polaridad es indistinta, por esta razón no encontrarás ninguna marca en el portalámparas. Tú necesitas la lámpara de lente para construir una [barrera de luz](#) junto con el [fototransistor](#).

La lámpara de lente está dimensionada para una tensión de 9 Volt y consumen aprox. 0,15 Amperios de corriente.

## Sensores

Los sensores son en cierta medida las contrapiezas de los [actores](#). Porque no ejecutan ninguna acción, sino reaccionan a determinadas situaciones y sucesos. Un pulsador reacciona por ejemplo a la "presión de un botón", dejando pasar o interrumpiendo una corriente eléctrica. Un sensor de calor reacciona a la temperatura de su entorno.

### Fototransistor



Se define el fototransistor también como "sensor de luminosidad". Este es un "sensor", que reacciona ante la luminosidad.

Él forma en una [barrera de luz](#) la contrapieza a la lámpara de lente. Ante una elevada luminosidad, o sea cuando el transistor está siendo irradiado por la lámpara, este conduce corriente. Cuando el rayo de luz se interrumpe, el transistor no conduce corriente.

#### Atención:

**En la conexión del fototransistor al suministro de corriente debes observar la correcta polaridad. El polo positivo debe ser conectado a la marcación roja en el fototransistor.**

### Pulsadores



Al pulsador también se le llama sensor de contacto. Al accionar el botón rojo se conmuta mecánicamente el interruptor, fluye corriente entre los contactos 1 (contacto central) y 3. Simultáneamente se interrumpe el contacto entre las conexiones 1 y 2. De este modo puedes emplear el pulsador de dos modos diferentes:

#### Como "cierre":

Se conectan los contactos 1 y 3.

[Pulsador oprimido](#): fluye corriente.

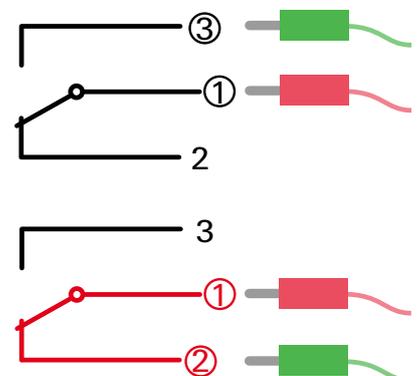
Pulsador no oprimido: no fluye ninguna corriente.

#### Como "ruptor":

Se conectan los contactos 1 y 2.

[Pulsador oprimido](#): no fluye ninguna corriente.

Pulsador no oprimido: fluye corriente.



## Sensor de calor (NTC)



Con este componente se trata de un sensor de calor, con los que se pueden medir temperaturas. A 20°C su **resistencia eléctrica** es de 1,5 k $\Omega$  (kilo-Ohm). NTC significa Coeficiente Negativo de Temperatura. Esto quiere decir simplemente, que el valor de resistencia desciende con el aumento de la temperatura.

Las informaciones que nos suministran los sensores (p.ej. claro-oscuro, presionado-no presionado, valor de temperatura) se puede, como aún veremos más tarde, a través del [ROBOTICS TXT Controller](#) encaminar al PC y después con ayuda del software p.ej. programar un motor de tal manera que un ventilador sople en el momento en que se interrumpa una **barrera de luz**.

## Módulo de cámara



Un sensor especialmente versátil es el módulo de cámara. La resolución de imagen es de 1 megapixel (esto significa, que cada imagen está constituida de un millón de puntos). Conecta la cámara a través de la hembra USB grande (USB1) al [ROBOTICS TXT Controller](#). Las imágenes que captura la cámara puedes transmitir al PC y visualizarlas en la pantalla. Verás siempre exactamente lo que el robot vea en ese momento. Además el ROBOTICS TXT Controller puede procesar las imágenes y de ese modo reconocer movimientos, colores o pistas, de manera que con ellos puedes controlar tus modelos de robot. Existe también la posibilidad de conectar la cámara directamente a la interfaz USB de tu PC y procesar entonces las imágenes en el Software ROBO Pro. También esta posibilidad la emplearemos en algunos modelos.

Puedes ajustar la nitidez de la imagen de la cámara girando el objetivo de la cámara.

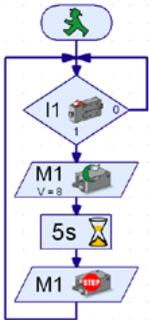
## Software ROBO Pro 4.x



ROBO Pro es una superficie gráfica de programación, con la que puedes crear los programas para el [ROBOTICS TXT Controller](#).

"Superficie gráfica de programación" significa, que tu no necesitas "escribir" los programas línea por línea, sino con ayuda de símbolos gráficos los puedes componer sencillamente con imágenes.

## TXT Discovery Set

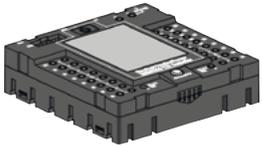


Un ejemplo para un programa de estas características lo encontrarás en la figura a la izquierda.

Cómo se crea exactamente un programa así está descrito detalladamente en el capítulo "Primeros pasos". También [ROBO Pro Ayuda](#) te muestra en los capítulos 3 y 4 como funciona esto.

El software ya ha sido instalado en tu PC con este material adjunto.

## ROBOTICS TXT Controller



El ROBOTICS TXT Controller es el núcleo de este kit de construcción ROBOTICS. Este controla los [actores](#), y evalúa las informaciones de los [sensores](#).

Para esta tarea el ROBOTICS TXT Controller dispone de numerosas conexiones a los que puedes conectar los componentes. Qué componentes se pueden conectar a qué conexiones y cuáles son las funciones de las conexiones, está descrito en el manual de instrucciones del ROBOTICS TXT Controller.

A través del display táctil a color puedes manejar confortablemente el ROBOTICS TXT Controller. En la hembra principal USB (USB-1) puedes conectar el módulo de cámara contenido en el kit de construcción. Una especial exquisitez es la interfaz integrada Bluetooth y WLAN. A través de ella puedes conectar sin cables tu PC con el ROBOTICS TXT Controller, y también varios Controller entre sí.

Como el Controller se maneja con los componentes individuales y qué es lo que individualmente deben hacer, lo determinas en el programa que escribes en el [software ROBO Pro](#).

### Alimentación de corriente (no incluida)



Como muchos de los componentes de ROBOTICS TXT Discovery Set funcionan con corriente, necesitas naturalmente también una alimentación de corriente.

Para ello se adapta especialmente el acumulador fischertechnik Accu Set. Este no está incluido en el kit de construcción.

## Algunas sugerencias

Experimentar produce las máximas satisfacciones, cuando los experimentos también funcionan. Por esta razón al montar los modelos debes observar algunas reglas básicas:

### **Trabajar cuidadosamente**

Tómate tu tiempo y mira detenidamente en las instrucciones de construcción para el modelo. Cuando posteriormente se debe buscar un error, se lleva mucho más tiempo.

### **Comprobar las piezas móviles**

Controla durante el montaje siempre si las piezas que deben moverse también permiten moverse con facilidad.

### **Utilizar prueba de interfaz**

Antes que comiences a escribir un programa para tu modelo, debes comprobar todas las piezas conectadas al [ROBOTICS TXT Controller](#) con ayuda de la [prueba de interfaz](#) de ROBO Pro. Como funciona esto exactamente está explicado en [ROBO Pro Ayuda](#) en el capítulo 2.4.

## Primeros pasos

Bien. Tras todos los preparativos e informaciones ahora puedes poner manos a la obra.

En este capítulo queremos mostrarte, como tu

- construyes el primer modelo sencillo, o sea el ventilador y lo conectas al [ROBOTICS TXT Controller](#)
  - conectas el ROBOTICS TXT Controller con la alimentación de corriente y el PC
  - llamas el [Software ROBO Pro](#) y compruebas el modelo
  - cargas e inicias un programa ROBO Pro para el modelo
  - creas e inicias tú mismo un primer programa sencillo en ROBO Pro
- [Haz clic aquí](#), para que se te muestren los primeros pasos fácilmente comprensibles.



Como para experimentar, además de los componentes de fischertechnik también trabajarás con el [Software ROBO Pro](#), como primera medida debes familiarizarte con él y aprender cómo se crea un programa. Como esto está realmente explicado de forma excepcional en los capítulos 3 y 4 de [ROBO Pro Ayuda](#), de preferencia deberías seguir con el tema, de estudiar este capítulo detalladamente.

Aquí también vale la sugerencia: tómate tu tiempo y ocúpate de ello, entonces a continuación tendrás tantas más satisfacciones con los modelos.

## Modelos de iniciación

Después que hayas leído los capítulos 3 y 4 de [ROBO Pro Ayuda](#), puedes ahora programar ya algunos modelos del kit de construcción. Por esta razón nosotros también queremos poner manos a la obra. En ROBO Pro existen diferentes escalas de dificultad, llamados niveles, que puedes configurar en tu barra de menú. Comenzamos con programas sumamente sencillos en nivel 1. Siempre que hayas terminado de construir y cablear un modelo, comprueba con ayuda de la [prueba de interfaz](#), si en el [ROBOTICS TXT Controller](#) todas las entradas y salidas están correctamente conectadas y los [sensores](#), [motores](#) y [lámparas](#) funcionan correctamente.

### Semáforo para peatones

Delante de vuestra casa se ha instalado un semáforo para peatones. Dado que el montador de la empresa instaladora está con mucha prisa, tú te ofreces asumir para él, la programación del control del semáforo.



El hombre te explica, como debe funcionar el control. Pero primero monta el modelo.

#### Tarea: (nivel 1)

El semáforo en principio debe estar en rojo. Cuando un peatón oprime el pulsador I1, el semáforo 3 segundos más tarde tiene que pasar a verde. La fase verde debe durar 10 segundos, a continuación debe volver al rojo.



#### Sugerencias de programación:

Los diferentes LEDs pertenecen a las siguientes salidas del TXT Controller

- Rojo – M1
- Verde – M2

Cuando la lámpara se conecta y desconecta sucesivamente de esta manera, se ha realizado el desarrollo deseado.

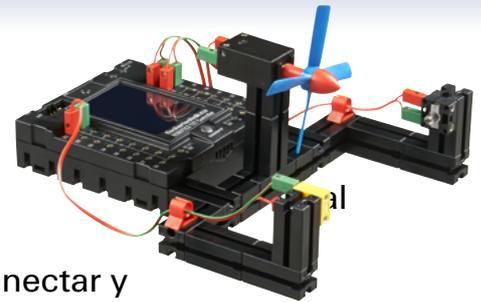
El programa terminado puedes cargarlo nuevamente haciendo clic sobre la imagen a la derecha:

[Semáforo\\_para\\_peatones.rpp](#)



## Secador de manos

En tu escuela se han instalado en los aseos nuevos secadores de manos junto lavamanos. Estos están provistos con una [barrera de luz](#), a través de la cual se puede conectar y desconectar el ventilador.



Monta primero el modelo como se describe en las instrucciones de construcción.

### Tarea 1: (nivel 1)

El secador de manos debe ser programado de tal manera, que en el momento que se interrumpe la barrera de luz, se conecta el ventilador y tras 5 segundos se vuelve a desconectar.



### Sugerencias de programación:

En el desarrollo del programa conecta primero la lámpara para la [barrera de luz](#) en la [salida M2](#).

A continuación esperas un segundo, para que el fototransistor tenga tiempo de reaccionar a la luz. Recién entonces la barrera de luz funciona correctamente.

Entonces consultas al [fototransistor](#) en la [entrada I1](#). Si el valor es "1" (la barrera de luz no está interrumpida), la entrada debe ser consultada de forma continua a través de un bucle.

En el momento que el valor se torna "0" (barrera de luz interrumpida), conectas el motor M1 y lo desconectas nuevamente tras 5 segundos.

A continuación se debe consultar nuevamente le fototransistor etc.

Inicia tu programa con el [botón Start](#) y comprueba si funciona como es deseado. En caso afirmativo, estás por el mejor camino de ser un programador profesional ROBO Pro.

Si aún no funciona, intenta descubrir a que se debe:

Con la [prueba de interfaz](#) puedes comprobar, si todas las entradas y salidas funcionan y están correctamente conectadas.

Mientras que se desarrolla el programa, puedes hacer un seguimiento de la ejecución del programa en función de los elementos marcados en rojo. De ese modo puedes reconocer rápidamente, donde se ha introducido un error.

Finalmente puedes comparar tu programa con el programa de ejemplo terminado, que puedes llamar a través del símbolo a la derecha:

Secador\_de\_manos\_1.rpp

Después de haber superado esta valla, queremos modificar algo el planteamiento del problema:

### Tarea 2: (nivel 1)

Al rector, que siempre está pendiente de ahorrar energía, no le gusta que los secadores de manos continúen funcionando aún un tiempo más, si bien las manos ya están secas. Él te solicita, crear el programa de tal manera, que el ventilador se desconecte, en el momento que se retiran las manos. Ningún problema para ti, ¿o?



### Sugerencias de programación:

Como en el primer programa tú consultas con una derivación al [fototransistor I1](#). Cuando el valor es "0", conectas el motor M1, cuando el valor es "1" desconectas el motor M1 etc.

También para esta tarea existe en caso de emergencia un programa terminado:

Secador\_de\_manos\_2.rpp

## Regulador de temperatura



En vuestra casa se ha instalado un nuevo equipo de aire acondicionado. Naturalmente le has preguntado inmediatamente al instalador, cómo funciona el regulador de temperatura. Él te ha explicado, que un sensor de temperatura mide permanente la temperatura actual. En el momento que se sobrepasa el valor límite superior, se conecta la refrigeración. Cuando se está por debajo de un valor límite inferior, se desconecta la refrigeración y se conecta la calefacción. Ahora quieres en función del modelo "Regulador de temperatura" intentar, asimismo de programar un [circuito de regulación](#) de este tipo. Monta primero el modelo.

### Tarea: (nivel 1)

La calefacción se simula a través de la lámpara de lente M2. Como "grupo de refrigeración" sirve el soplador en la [salida M1](#). Para la medición de la temperatura empleamos la resistencia NTC en la [entrada I8](#).

Programa el modelo de tal manera, que sobre un determinado valor de temperatura se desconecta la calefacción y se conecta el soplador. Este debe refrigerar hasta tanto se haya alcanzado el valor límite inferior. A continuación se debe desconectar el soplador y conectar la calefacción.



### Sugerencias de programación:

**Observa:** el valor de resistencia de la [resistencia NTC](#) desciende con el aumento de la temperatura. El valor límite superior de temperatura es entonces el menor valor de I8. A este valor límite se debe conectar el soplador. El valor límite inferior de temperatura es el mayor de I8. A este valor límite se debe conectar la calefacción.

Qué valor posee I8 a temperatura ambiente, lo puedes descubrir con la [prueba de interfaz](#). Tú conectas la lámpara M2 y observas, hasta donde desciende el valor. Entonces conectas el soplador y verificas hasta donde asciende el valor. De acuerdo a ello seleccionas los valores límite para calefacción y refrigeración.

A través de un clic sobre el símbolo, puedes abrir el programa terminado.

Regulador\_de\_temperatura.rpp



## Barrera

Frecuentemente uno encuentra una barrera de este tipo en la entrada o salida de un aparcamiento. Monta primero el modelo como se describe en las instrucciones de construcción.



Nuestra barrera podría encontrarse en la salida.

### Tarea: (nivel 2)

Cuando un vehículo se acerca a la barrera la barrera de luz se interrumpe con ello. Entonces la barrera debe abrirse, mantenerse abierta un tiempo determinado (p.ej. 5 segundos) y luego volver a cerrarse, pero solo cuando la barrera de luz no está interrumpida, debido a que en caso contrario la barrera probablemente dañe el vehículo que aún se encuentra debajo de la barrera. Para indicarle al conductor del automóvil si el camino a través de la barrera está libre, el semáforo debe brillar en verde cuándo la barrera está abierta y en rojo cuando se cierra.



### Sugerencias de programación:

Emplea en cada caso para los procedimientos "Abrir" y "Cerrar" la barrera, un subprograma (véase también [ROBO Pro Ayuda](#) capítulo 4).

En ROBO Pro ajustas para ello el nivel 2.

A través de un clic sobre el símbolo, puedes abrir el programa terminado.

Barrera.rpp



## "Operador de cámara"

Ahora por primera vez nos ocuparemos en este cuaderno adjunto con el apasionante tema de la cámara y el procesamiento de imágenes. Para ello montas el modelo "Operador de cámara" como se describe en las instrucciones de construcción y conectas la cámara directamente al PC a través de la conexión USB. La cámara se reconoce y el controlador correspondiente se instala automáticamente.

Inicia ahora el software ROBO Pro.



### Tarea 1:

Monta el operador de cámara delante de tu PC y deja que ROBO Pro indique qué es lo que la cámara "ve".



### Sugerencias de programación:

Para ello ni siquiera tienes que crear un programa ROBO Pro. Abre en ROBO Pro simplemente un programa nuevo, ve al comando de menú "Cámara", establece allí la conexión de la cámara a "PC" y haz clic en "Conectar cámara".

Mediante giro del objetivo de la cámara puedes dar nitidez a tu imagen.

### Tarea 2: (nivel 3)

El operador de cámara tiene que reaccionar ahora a los movimientos en la imagen. En el momento que alguien atraviesa la imagen en el TXT Controller debe parpadear tres veces una luz roja en la salida M1.

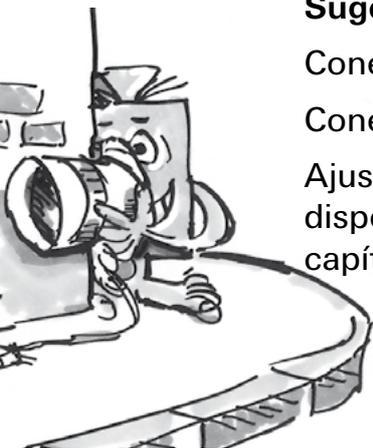


### Sugerencias de programación:

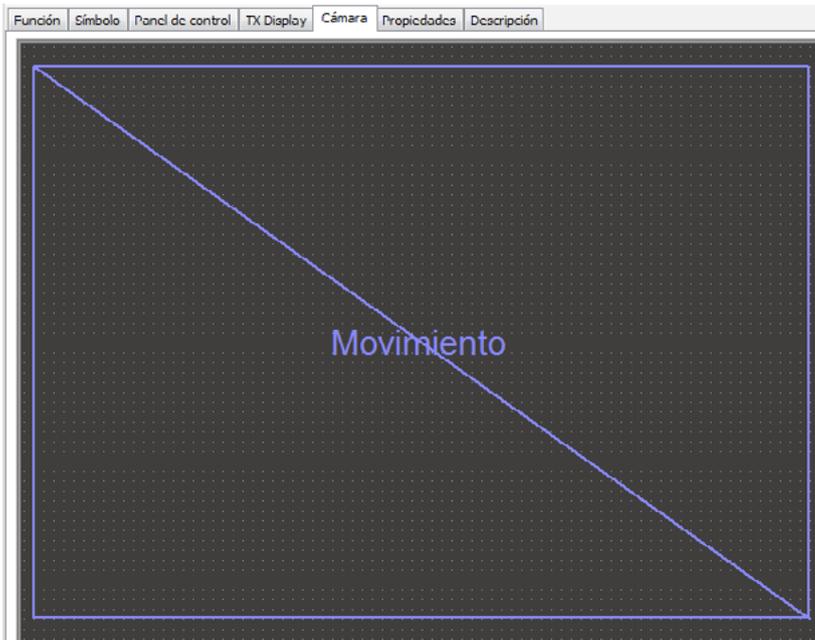
Conecta el TXT Controller a tu PC a través de una segunda interfaz USB.

Conecta al TXT Controller un LED con cubierta roja a la salida M1.

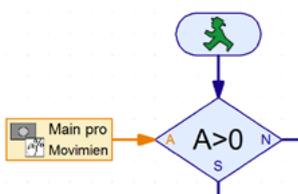
Ajusta en ROBO Pro en nivel 3. Cuáles son las funciones que allí están disponibles adicionalmente, puedes leerlo en [ROBO Pro Ayuda](#) en el capítulo 5.



Emplea en ROBO Pro el campo de sensor de la cámara "Movimiento" y delinéalo en el campo de la cámara como rectángulo en el tamaño deseado.



En el campo de diálogo del detector de movimientos que puedes abrir con un clic derecho del ratón sobre el campo de sensor introducido, se permite ajustar la intensidad del movimiento para que se active el sensor y cual el tamaño del movimiento con relación al campo de sensor total para que reaccione. De preferencia intenta hacerlo con valores preconfigurados y pruebas más tarde, que es lo que provocan los ajustes.



Crea en ROBO Pro un desarrollo, en el que consultas, si el campo de sensor ha detectado un movimiento. Para la consulta del campo de sensor emplea el elemento de entrada de la cámara del nivel 3 de ROBO Pro y consultas con una derivación, si existe un movimiento:

Consulta de la entrada "Movimiento C"  (contraste de modificación). ¿El valor es >0? Si este es el caso, la luz tiene que parpadear.

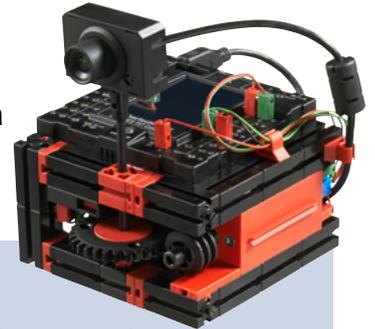
Véase otras indicaciones sobre los campos de sensor de la cámara en [ROBO Pro Ayuda](#) capítulo 11.

El programa de ejemplo terminado puedes abrirlo aquí.

Operador\_de\_cámara.rpp

## Cámara basculante

Como siguiente paso montas una cámara de vigilancia para tu habitación. Monta primero el modelo como se describe en las instrucciones de construcción.



### Tarea: (nivel 3)

La cámara accionada por el [motor de codificador](#), siempre tiene que girar una pequeña sección y a continuación comprobar si algo se mueve en el ambiente. Si este es el caso, debe emitir una alarma a través del altavoz del TXT Controller, si no se mueve nada, debe girar un poco más y volver a comprobar.

Una vez alcanzado el rango de basculación, debe moverse nuevamente en pequeños pasos de regreso a su posición de partida.



### Sugerencias de programación:

La cámara debe primero desplazarse a su posición de partida (hasta oprimir el pulsador I1).

Entonces siempre debe desplazarse una determinada cantidad de impulsos en la otra dirección. Prueba simplemente con ayuda de la [prueba de interfaz](#) cuántos impulsos son convenientes por movimiento de giro.

Consulta con el elemento de sensor de la cámara "Movimiento", si en la imagen se modifica algo. En caso afirmativo, el TXT Controller emite un tono de alarma a través de su altavoz, véase ([ROBO Pro Ayuda](#)).

Repite este procedimiento tantas veces, hasta que el rango de basculación de la cámara ha finalizado y desplázate en la otra dirección, de vuelta al final de carrera I1.

Aquí encontrarás nuestra solución para la tarea:

Cámara\_basculante.rpp

Este modelo puedes operarlo

independientemente del PC en modo de descarga.

La perfecta cámara de vigilancia para ahuyentar intrusos no autorizados de tu habitación.

## Robots móviles – el siguiente desafío

### Robot desplazable



Con este modelo queremos descubrir conjuntamente, como se puede controlar un robot móvil. ¿Cómo se logra ponerlo en movimiento, cómo funciona la dirección y se puede quizás mejorar aún su exactitud? Estas preguntas se responden con ayuda de las tareas en este capítulo.

Pero primero naturalmente debes montar el robot. La descripción la encontrarás como siempre en las instrucciones de construcción.

Tómate tu tiempo para el montaje. Mira detenidamente los planos en las instrucciones de construcción, también el cableado. Si combinas los componentes con el [ROBOTICS TXT Controller](#) de manera diferente a lo descrito en las instrucciones de construcción, posiblemente el robot se comportará diferente a lo que tú esperas.

Comprueba tras el montaje todos los componentes conectados al ROBOTICS TXT Controller a través de la [prueba de interfaz](#) del Software ROBO Pro. Si dejas girar los motores hacia la izquierda, el robot se debe desplazar hacia delante.

Dirección de marcha	Dirección de rotación Motor 1	Dirección de rotación Motor 2
Avance	Izquierda	Izquierda
Retroceso	Derecha	Derecha
Izquierda	Izquierda	Derecha
Derecha	Derecha	Izquierda
Detener	Detener	Detener

#### Tarea 1: simplemente circular recto

Deja que el robot se desplace 3 segundos en línea recta (¡no sobre la mesa, peligro de caída!), y a continuación nuevamente 3 segundos recto hacia atrás.

¿Ha llegado el robot realmente de nuevo a su punto de partida?

Repite el programa varias veces y observa si el robot verdaderamente se mueve exactamente hacia delante y hacia atrás.



**Sugerencias de programación:**

Aun cuando la tarea seguramente no sea un problema para ti, aquí nuestra propuesta:

Robot\_desplazable\_1.rpp

**La dirección**

Aún si es divertido observar el robot desplazarse en línea recta, esto es algo monótono. Por ello ahora tiene que aprender, a desplazarse en curvas. ¿Cómo funciona esto? Muy sencillo:

**Tarea 2: circular también en una curva**

Deja el robot desplazarse nuevamente en línea recta a durante 3 segundos (ambos motores giran a la misma velocidad), modifica entonces por 1 segundo el sentido de rotación del motor derecho (M1), y a continuación deja circular el robot nuevamente 3 segundos en línea recta (o sea ambos motores a la misma velocidad en la misma dirección).

Descubre entonces, cuanto tiempo tienes que hacer funcionar los motores en diferentes direcciones para que el robot gire 90°.

**Sugerencias de programación:**

Modifica para ello el tiempo de espera tras el elemento de control, en el que se modifica el sentido del segundo motor.

El programa terminado puedes abrirlo nuevamente haciendo clic sobre el símbolo a la derecha:

Robot\_desplazable\_2.rpp

**Tarea 3: circular en una figura**

Después que ahora sabes, cuanto tiempo debes revertir el sentido de rotación de un motor para que el robot doble hacia la izquierda o hacia la derecha, programa el robot de tal manera que recorra un cuadrado y a continuación arrije nuevamente a su posición de partida.

Comprueba con una marcación, si el robot realmente se desplaza exactamente a su posición de partida.



**Sugerencias de programación:**

Para "doblar la esquina" puedes crear un subprograma. De este modo tu programa principal quedará más fácil de comprender.

Seguramente ya tienes en tu mente la solución para esta tarea. Pero de todas maneras, hay nuevamente una propuesta por nuestra parte:

Robot\_desplazable\_3.rpp

**¿Siempre lo mismo, y aún no es igual?**

Como seguramente ya has percibido, la exactitud de repetición del robot aún tiene capacidad de ser mejorada. Aun cuando él ejecuta varias veces la misma tarea, el resultado no es siempre el mismo. Esto tiene diferentes razones. Una de ellas es, que ambos motores no giran exactamente a la misma velocidad. Por ejemplo el engranaje en un motor puede funcionar más pesado que en el otro. Como ambos motores se operan con la misma tensión (9 Voltios), un motor entonces gira más lentamente que el otro. Debido a que hasta ahora siempre hemos controlado nuestro robot a través de tiempos de espera, quizás una rueda haya girado algo más que la otra en este tiempo.

La solución sería entonces, permitir que ambos motores giren exactamente a la misma velocidad. Exactamente esto es muy simple de realizar con los motores de codificador.

**Tarea 4: utilizar motores de codificador**

Repite las últimas tres tareas, y utiliza en lugar de los normales elementos de salida del motor y de tiempos de espera los elementos del motor de codificador. Como se emplean está explicado en ROBO Pro Ayuda en el capítulo 12.6.

**Sugerencia de programación:**

Con el elemento de motor de codificador puedes operar simultáneamente ambos motores con un elemento de programa. A través del campo de entrada de distancia te aseguras, que cada motor realmente gire hasta donde debe hacerlo.

Las propuestas de solución las puedes abrir nuevamente haciendo clic sobre los símbolos a la derecha:

Robot\_desplazable\_1\_sync.rpp

Robot\_desplazable\_2\_sync.rpp

Robot\_desplazable\_3\_sync.rpp

Para el conteo de los impulsos en las rápidas entradas de conteo C1-C4 no necesitas ningún otro elemento de programa en ROBO Pro. Al motor M1 se le asigna internamente de forma automática la entrada de conteo C1, M2 pertenece a C2 etc.

Nota: en caso que a pesar del empleo de los elementos de motor de codificador el modelo no se desplace en línea recta, la causa puede estar en el modelo mismo. Cuando p.ej. una tuerca de cubo, que transmite la fuerza del eje sobre la rueda, no está suficientemente apretada, el eje patina y el modelo realiza una curva, si bien los motores giran a la misma velocidad. Por esta razón aprieta firmemente las tuercas de los cubos.

## Detector de obstáculos

Tu robot ahora puede desplazarse en línea recta y doblar. Hasta ahora lo hace precisamente sólo de manera exacta como se la especificas con tu programa.

Pero un robot en realidad debería en lo posible reaccionar de manera autónoma. Por esta razón ahora queremos posibilitarle de reaccionar ante obstáculos, y con la ayuda de un parachoques.

Monta primero el modelo como se describe en las instrucciones de construcción.



### Tarea:

El detector de obstáculos debe circular en línea recta. En el momento de que con su parachoques izquierdo se tropiece con un obstáculo debe detenerse, retroceder un tramo corto, girar algo hacia la derecha y a continuación continuar su marcha en línea recta. Si se tropieza con su parachoques derecho sobre un obstáculo, debe desviarse correspondientemente hacia la izquierda.



### Sugerencias de programación

Parachoques izquierdo: Pulsador I6; Parachoques derecho: Pulsador I5

Emplea para las diferentes acciones: marcha adelante / atrás, así como desviarse hacia la izquierda / hacia la derecha en cada caso un subprograma.

Observa que el robot no se desvíe los mismos impulsos hacia la izquierda y hacia la derecha, sino puede suceder que al meterse en un rincón, ya no pueda salir. Si por el contrario la cantidad de impulsos es diferente, logrará en todos los casos volver a salir del rincón.

El programa terminado lo encontrarás aquí:

Detector\_de\_obstáculos.rpp



## Detector de obstáculos con cámara

Tu robot móvil tiene que recibir ahora una cámara, para que pueda ver a dónde se dirige y tú también lo puedas controlar a distancia. Para ello todas simplemente el detector de obstáculos y le montas una cámara como se describe en las instrucciones de construcción. La cámara la conectas a la interfaz [USB-1](#) del TXT Controller.



### Tarea 1:

Primeramente el robot debe comportarse de la misma manera como un detector de obstáculos sin cámara. Adicionalmente debe detenerse y girar, cuando le muestras la tarjeta roja.



Emplea para el reconocimiento de colores el campo de sensor de la cámara "Color" y agrándalo de manera que su superficie ocupe una gran parte de la pantalla de la cámara.

Programa el robot ahora de tal manera, que circule en línea recta cuando ve un tablero rojo y no haya ningún obstáculo. La detección de obstáculos puedes extraerla del programa "Detector de obstáculos".

Para la consulta del color emplea de preferencia un subprograma propio. Tu preguntas a través de la entrada de la cámara si el valor de rojo es mayor que el de verde y mayor que el de azul y si el color es más claro que negro.

Como tablero de color emplea la tarjeta roja que puedes recortar del modelo grande para cortar, incluido en el kit de construcción.

En la ventana de la cámara de ROBO Pro pueden leer bajo "valores de sensor", que valor de color tiene la tarjeta roja.

Nuestra solución la vuelves a encontrar aquí:

Nota:

[Detector\\_de\\_obstáculos\\_con\\_cámara\\_1.rpp](#)

Impide la radiación directa del sol. Puede suceder, que el robot reconozca la luz solar como tablero rojo. Ante todo una luz solar brillante, que posee una elevada proporción de rojo.

**Tarea 2: control remoto del robot**

Nota: para poder procesar esta tarea necesitas una conexión WLAN entre el TXT Controller y tu PC. Como puedes establecerla está descrito en el manual de instrucciones del TXT Controller.

**Sugerencias:**

Una vez que hayas establecido exitosamente la conexión WLAN, abres a través del símbolo el programa adecuado.

Cambia a la interfaz de usuario y encontrarás allí la superficie para el control remoto del modelo.

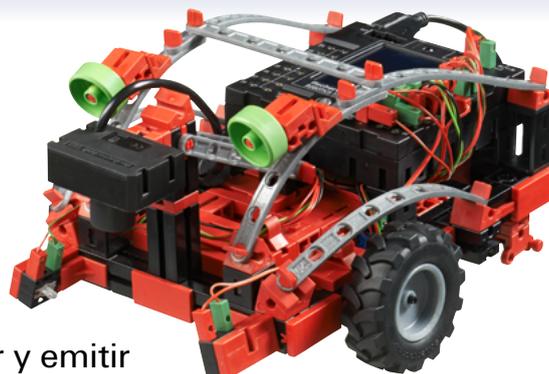
Detector\_de\_obstáculos\_con\_cámara\_2.rpp

Inicia el programa en modo online. Ahora puedes controlar a distancia el robot a través de diferentes botones y además puedes ver hacia dónde se dirige. Si en algún momento no percibes un obstáculo, este es reconocido por los parachoques y el modelo se desplaza un tramo corto hacia atrás.

¡Te deseamos muchas satisfacciones al investigar el entorno!

## Rastreador

En este modelo se emplea la cámara para dejar circular el robot a lo largo de una línea. Lo genial en el seguimiento de una pista con la cámara es que la cámara no solo reconoce si se dispone de una pista o no, sino puede reconocer y emitir su posición exacta en la imagen de la cámara. Ante ello el robot puede reaccionar y ya sea circular en línea recta, cuando la imagen se encuentra exactamente en el centro de la imagen, o corregir hacia la izquierda o hacia la derecha, cuando "ve" que la pista se encuentra fuera del centro de la imagen.



El objetivo es que el robot busque la línea negra y se desplace a lo largo de ella.

Pero una cosa detrás de la otra. Como primera medida tienes que montar el modelo rastreador. Como se hace estos, se encuentra naturalmente en las instrucciones de construcción.

Después de que has finalizado el montaje de tu modelo, debes comprobar con la [prueba de interfaz](#), si todos los componentes están conectados al [ROBOTICS TXT Controller](#) y funcionan.

### Tarea 1: reconocer una pista

Programa el robot de tal manera, que siga una pista recta negra sobre la cual ha sido colocado. Si pierde la pista o ha llegado a su final, el robot debe detenerse. Emplea para esta tarea la pista de recorrido del kit de construcción.



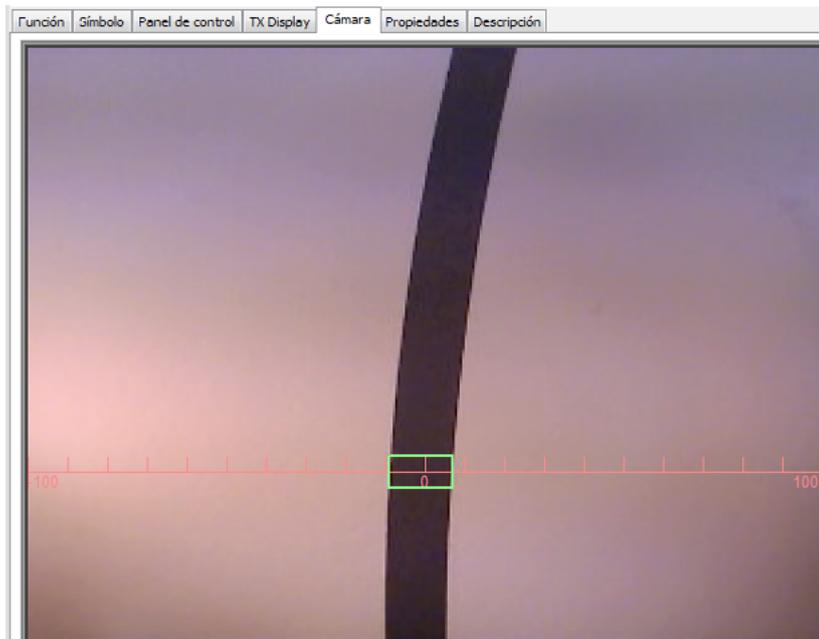
### Sugerencias de programación:

Emplea para la prueba la pista negra del recorrido del kit de construcción.

En la ventana de la cámara de ROBO Pro ajustas la cámara de tal manera que la pista pueda ser reconocida con nitidez.

Para el seguimiento de la pista emplea el campo de sensor de cámara "Línea". Está constituida de una línea recta en la imagen de la cámara, que se delinea de izquierda a derecha a través de la imagen y contiene una escala de -100 a +100. 0 es exactamente el centro.





Véase otros detalles sobre el campos de sensor "Línea" en [ROBO Pro Ayuda](#) capítulo 11.

Ahora le preguntas a tu programa en qué posición se encuentra la pista.

- Si se encuentra entre  $-10$  y  $+10$  el robot tiene que desplazarse en línea recta (M1 y M2 = izquierda:  $v=5$ )
- Entre  $-11$  y  $-40$  debe corregir ligeramente hacia la izquierda (M1 izquierda:  $v=5$ , M2 izquierda:  $v=2$ )
- Entre  $+11$  y  $+40$  debe corregir ligeramente hacia la derecha (M1 izquierda:  $v=2$ , M2 izquierda:  $v=5$ )
- Si el valor es  $< -40$  debe corregir intensamente hacia la izquierda (M1 izquierda:  $v=5$ , M2 derecha,  $v=3$ )
- Si el valor es  $> +40$  debe corregir intensamente hacia la derecha (M1 derecha:  $v=3$ , M2 izquierda:  $v=5$ )
- Si pierde la pista, tiene que detenerse.

El programa terminado lo encontrarás aquí:

Rastreador\_1.rpp

Ahora tu robot puede reaccionar. Sin embargo sería aún mejor, si el robot al final de la pista en lugar de detenerse girara y la siguiera nuevamente.

**Tarea 2: girar al final de la pista y seguirla nuevamente**

Amplia tu programa en la función, que el robot al abandonar la pista de la vuelta y la siga nuevamente.

**Sugerencias de programación:**

Hay varias posibilidades de corregir la dirección. Puedes detener un motor y dejar que el otro continúe girando, o hacer girar un motor en sentido contrario al de desplazamiento. Prueba que método se adapta mejor.

Aquí nuestra propuesta de solución:

Rastreador\_2.rpp

Pues bien. Ahora el robot puede circular por un "carril" óptico. La desventaja es, que primero se lo debe asentar sobre el carril. Esto queremos modificarlo. El robot tiene que buscar su pista ahora independientemente.

**Tarea 3: encontrar la pista y a continuación seguirla**

Escribe un subprograma "buscar", con el que el robot busque la pista negra, cuando al iniciar el programa no encuentra ninguna. Para ello el robot primero tiene que girar 1 vez en círculo. Si en este caso no encuentra ninguna pista, debe avanzar un tramo en línea recta. En el momento que el robot detecta una pista, deberá seguirla. En caso contrario la búsqueda debe comenzar desde el inicio. Si ha girado 10 veces en círculo sin encontrar una pista, debe detenerse y parpadear 3 veces.

**Sugerencias de programación:**

En caso que aún no te encuentres en la pista correcta, aquí nuestra propuesta de solución:

Rastreador\_3.rpp



## Robot de reconocimiento

En este modelo queremos ahora combinar varias funciones de la cámara. El robot debe poder mirar hacia delante para poder reconocer su entorno, pero también tiene que estar en condiciones de mirar hacia abajo para poder seguir diferentes pistas de color sobre la pista de recorrido suministrada. Además debe reunir "datos de temperatura" de su entorno con su resistencia NTC.

Monta primero el modelo como se describe en las instrucciones de construcción.

### Tarea 1: el robot tiene que funcionar como sigue:

Primero se lo coloca sobre la pista negra. Este se desplaza a lo largo de ella, hasta que finaliza. A continuación se da la vuelta y desplaza en la otra dirección. Una vez que ha llegado al final de la pista negra en donde la misma se ensancha, debe levantar la cámara y mirar hacia delante. Ahora si le "muestras" una tarjeta roja o verde debe a continuación seguir primero la pista negra y entonces doblar sobre la pista roja o la verde. Antes de que se ponga nuevamente en marcha mide aún todavía la temperatura del entorno y la emite en el display del TXT Controller.



Las cartas de color puedes recortarlas del modelo para recortar grande en el kit de construcción

### Sugerencias de programación:

Con este programa queremos mostrarte, todo lo que es posible con el ROBO Pro. Debido a que somos conscientes que programar esto no es muy sencillo, carga simplemente el programa preparado y Pruébalo:

Robot\_de\_reconocimiento.rpp

### Tarea 2:

El subprograma "Pista" es el responsable para el control de los motores de codificador durante el seguimiento de la pista. Compáralo con el subprograma del modelo rastreador. ¿Qué diferencias te llaman la atención?

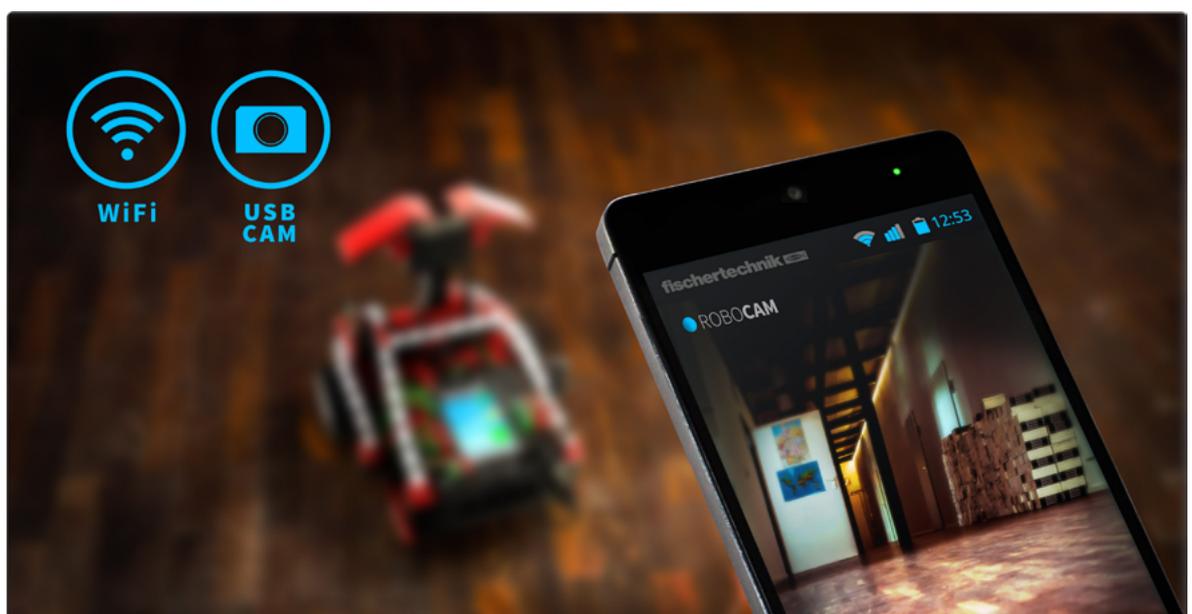


**Solución:**

El subprograma en el robot de reconocimiento es considerablemente más complejo que en el rastreador. Contiene una regulación que cuida de que el modelo corrija la dirección con mayor intensidad, cuanto mayor esté distanciada la pista del centro. En el rastreador, por el contrario, solo se ha diferenciado entre línea recta, ligeramente corregido e intensamente corregido. Para poder crear una regulación de este tipo, ya se debe tener mucha experiencia al programar (así como nuestros programadores). Pero para ello el resultado de la regulación es también considerablemente mejor que en el modelo de rastreador. Para algunos programas puedes simplemente en el futuro emplear este subprograma y alegrarte de que funciona tan bien.

**Tarea 3:**

En caso de que poseas un Smartphone (de momento con sistema operativo Android) puedes controlar a distancia el robot de reconocimiento. Puedes descargar una aplicación de tu Google Playstore. Esta se llama "TXTCamdroid". Conecta el TXT Controller a través de WLAN con tu Smartphone y ya está. En el display de tu Smartphone puedes ver lo que la cámara ve. A través de diversas teclas puedes controlar tu modelo y enviarlo a realizar un reconocimiento. ¡Mucha diversión!



## Robot de fútbol con control de movimiento



¿Has escuchado alguna vez de la Robo-Cup? Este es el campeonato mundial de fútbol para robots.

Este se realiza cada año en un país diferentes. Hay diversas ligas, en la que se reúnen los diferentes tipos de robots. Más información la encontrarás por ejemplo en la página principal de Robo-Cup

<http://www.robocup.org>.

En las instrucciones de construcción encontrarás tres propuestas para el montaje de robots de fútbol. Esta son exactamente tan maniobrables como nuestros otros robots, pero tiene adicionalmente aún una **barrera de luz** para reconocer un balón y con ello activar un "mecanismo de disparo". Con la cámara podemos equipar y controlar ahora el robot de diversas maneras.

Monta primero el modelo "Robot de fútbol con control de movimiento" como se describe en las instrucciones de construcción.

A continuación nos pondremos manos a la obra para programarlo y "entrenarlo" con algunos trucos de jugadas. Como siempre debes primero comprobar las funciones del modelo con la **prueba de interfaz**, antes de que comiences con la programación.

Como balón empleas la pelota de tenis de mesa de color naranja, incluida en el kit de construcción.

Nota: el pulsador que se emplea como final de carrera para la palanca del mecanismo de disparo, debes ajustar eventualmente algo, de manera tal que si bien se presiona, cuando la palanca no se encuentra delante, sin embargo simultáneamente no bloquee la palanca.

En este robot la cámara no se incorpora al modelo, sino se encuentra sobre un soporte propio a un lado del modelo. Esta se enchufa directamente en la interfaz USB del PC.

**Tarea 1: "Ha atrapado el balón, y dispara ..."**

En el primer paso nuestro artista electrónico del balón tiene que aprender a disparar el balón en el momento de ser reconocido por la [barrera de luz](#). Experimenta un poco con la "velocidad de disparo". Eventualmente una corta pausa entre "reconocer" y "disparar" puede conducir a una mejora.

**Sugerencias de programación:**

Para esta tarea puedes conectar primero el TXT Controller con el PC a través del cable USB.

Como en el [secador de manos](#) debes esperar un segundo en la barrera de luz tras la conexión de la [lámpara de lente](#) antes de consultar el [fototransistor](#).

Como entrenador uno no lo tiene fácil. En caso que tu jugador robot no te haga caso, quizás lo puedas convencer con nuestra propuesta de programa:

Robot\_de\_fútbol\_con\_control\_de\_movimiento\_1.rpp

Debido a que un verdadero artista del balón tiene que saber más cosas que sólo disparar el balón, queremos ampliar un poco aún las habilidades de nuestro robot de fútbol.

**Tarea 2: control de movimiento del robot de fútbol**

El robot de fútbol ahora debe ser programado de tal manera, que reaccione a movimientos que ejecutas delante de la cámara. Si saludas con la mano izquierda, tiene que desplazarse hacia la izquierda, si saludas con la derecha, debe girar hacia la derecha. Si saludas con ambas manos en el centro de la imagen, debe desplazarse en línea recta. Cuando la barrera de luz reconoce el balón, este debe ser disparado, de preferencia naturalmente dentro de la portería.

**Sugerencias de programación:**

Nota: para esta tarea el TXT Controller debe ser conectado con tu PC a través de WLAN o Bluetooth.

Aún si de momento este planteamiento del problema parece difícil, la

mayoría de las funciones para este programa ya las has empleado en tus otros robots.

Para el control del movimiento empleas tres campos de sensor de cámara "Movimiento"\*. Uno de ellos lo ubicas en el centro de la imagen de la cámara uno en la sección izquierda y el otro en la derecha. Ubica la cámara delante de la pantalla de tal manera, que con tus brazos puedas alcanzar bien las tres secciones en la ventana de la cámara, pero que los campos de sensor no reaccionen de forma indeseable. Para ello ubicas los campos de preferencia en la mitad superior de la superficie de la cámara.

Entonces simplemente consultas en tu programa los tres campos de sensor y según el cual reconoce un movimiento, controlas tu robot hacia la izquierda, derecha o en línea recta. El mecanismo de disparo ya lo has programado en la tarea 1. A él lo puedes integrar en tu programa.

A continuación inicias el programa desde ROBO Pro en modo online. La imagen de la cámara llega directamente al PC a través de la interfaz USB, el control del robot se realiza sin cables a través de Bluetooth o WLAN.

Del estadio de fútbol, incluido en el kit de construcción, solo necesitas la portería para este modelo. Las "bandas" como máximo las puedes emplear para impedir que el balón ruede a cualquier lugar. Para la orientación del robot sin embargo recién se necesita en el próximo modelo. Ahora mucha diversión al "saludar" y disparar goles.

También para este programa existe de vuelta una propuesta de solución:

Robot\_de\_fútbol\_con\_control\_de\_movimiento\_2.rpp

\* véase también [ROBO Pro Ayuda](#), capítulo 11.

## Robot de fútbol

Si bien es gracioso controlar el robot de fútbol saludando con los brazos, en realidad, sin embargo, un verdadero robot de fútbol tiene que estar en condiciones de llevar su propio balón de forma independiente a la portería. Eso es lo que queremos enseñarles e instalamos para ello la cámara directamente en el robot.



Cómo se hace esto, te muestran nuevamente las instrucciones de construcción.

La cámara la conectas a la interfaz [USB-1](#) del TXT Controller.

Para este modelo se necesita el estadio de fútbol completo. Está constituido de un reverso blanco sin impresión de la pista de recorrido ya empleada en el rastreador y en el robot de reconocimiento, así como las bandas impresas con las líneas negras y la portería.

El sustrato totalmente blanco lo necesita el robot para reconocer sobre él con el campo de sensor de cámara "Balón" el balón naranja.

En función de las líneas negras sobre las bandas que de cada lado tienen una apariencia diferente, el robot puede reconocer en qué punto del campo de juego se encuentra y donde está la portería. Las líneas en la pared posterior de la portería le indican a dónde tiene que disparar cuando tiene el balón.

### Tarea 1:

De forma similar al robot de reconocimiento también la programación de este robot no es demasiado sencilla. Prueba por ello primero el programa suministrado, cargándolo sobre el TXT Controller.

Aquí nuestra propuesta de solución:

Robot\_de\_fútbol.rpp

### Sugerencias:

Ajusta la cámara en el objetivo de tal manera, que las líneas de banda sobre las bandas de la portería en el estadio sean reconocibles nítidamente

cuando el robot se encuentra en el otro extremo del estadio.

Coloca el robot en el estadio y arroja el balón al campo de juego. El robot busca el balón, intenta atraparlo y dispara, en el momento que ha girado en dirección a la portería, el balón en dirección a la portería. Con un poco de suerte el robot logra disparar un gol.

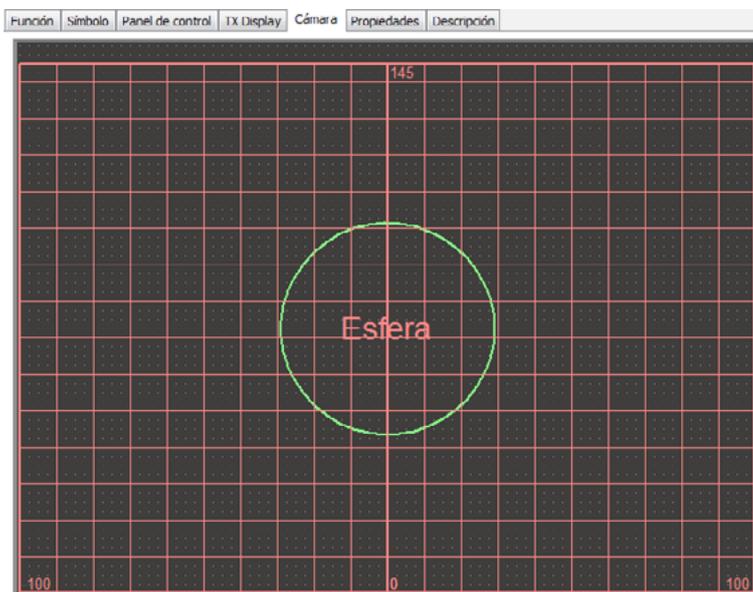
### Tarea 2:

Intenta en función de la descripción en ROBO Pro Ayuda a entender, cómo funciona el reconocimiento de balón.



### Solución:

El campo de sensor "Balón" reconoce un balón de color sobre un sustrato blanco. Este emite la posición del centro del balón en el campo de sensor. Véase también [ROBO Pro Ayuda](#), capítulo 11.



De izquierda a derecha, de forma similar al rastreador, 0 exactamente en el centro. En la altura el valor va desde 0 a un valor de altura máxima calculado automáticamente. Según donde se encuentre el balón, el robot elige su dirección de marcha.

**Tarea 3:**

¿Qué elemento de sensor de cámara emplea el programa para evaluar las líneas sobre la banda?



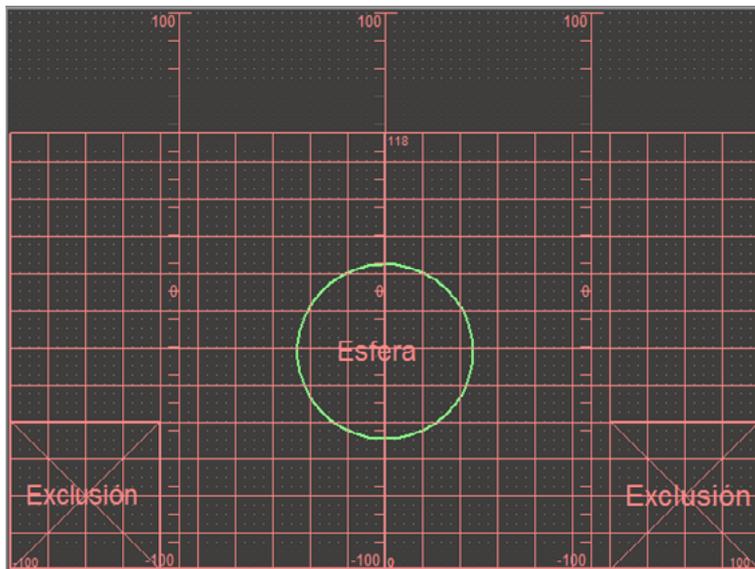
**Solución:**

Se emplea el elemento de sensor "Línea". Tres reconocimientos de línea se ubican en contrapartida al rastreador de arriba hacia abajo en la ventana de la cámara y reconocen así las tres líneas delineadas horizontalmente sobre la banda. En función de su ancho el robot calcula, donde se encuentra en ese momento y a donde tiene que ir.

Él puede además reconocer en función de las líneas, cuándo llega demasiado cerca de la banda y se detiene a tiempo para evitar que golpee contra la banda.

**Tarea 4:**

En la ventana de la cámara del programa ROBO Pro para este robot encontrarás 2 campos de sensor "Entalladuras". ¿Puedes imaginarte qué función tienen?



**Solución:**

Estos campos de sensor complementan el campo de sensor de reconocimiento del balón. Las áreas entalladas se excluyen de la búsqueda

del balón. Esto puede ser de gran utilidad, cuando en la imagen se encuentran objetos, p.ej. el [fototransistor](#) del robot de fútbol, que son erróneamente reconocidos como balón por el reconocimiento del balón.

### ¿Cómo se orienta el robot en el campo de juego?

Quizás alguna vez ya te has preguntado cómo hace el robot para ubicarse en función de los códigos de líneas sobre las bandas en el campo de juego. Esto en realidad no es del todo sencillo y funciona en principio de la siguiente manera:

Efectivamente el robot está en condiciones en función de la banda determinar exactamente su posición en el campo de juego con una precisión de unos 2 cm y el ángulo a unos 5 grados. Para ello el robot mide la altura de las bandas en diferentes puntos y calcula de la altura, las distancias y de ellas finalmente el ángulo y la posición absoluta. Este procedimiento se lo conoce como triangulación.

La banda tiene siempre 3 líneas negras y entre ellas 2 blancas. Una de las líneas negras es más ancha que las otras. De ella el robot reconoce de qué banda se trata. En total resultan así de ello 5 diferentes modelos de banda, uno para cada lado del campo de juego y uno para la portería.

Más allá de ello el robot emplea los impulsos de conteo de los motores de codificador para calcular la posición y el ángulo entre las triangulaciones.

Este procedimiento se le llama odometría. Para distancias y tiempos cortos la odometría es más exacta. Sin embargo tiene una desventaja, que los errores con el tiempo se suman. Por esta razón los datos obtenidos a través de la odometría se corrigen con los datos de la triangulación.

En el programa robot de fútbol existe para estos cálculos los dos subprogramas Triangulación y Odometría. Si tienes interés, míralo una vez con detenimiento. ¡Pero no te asustes! Parecen bastante complejos y es que lo son también.

Si todo te parece demasiado teórico esto tampoco es un problema. Toma simplemente el programa existente, sorpréndete de cómo funciona y apuesta con tus amigos cuántos intentos necesita el robot y cuánto tarda hasta que acierte la portería.

**Aquí aún algunas sugerencias sobre el campo de juego:**

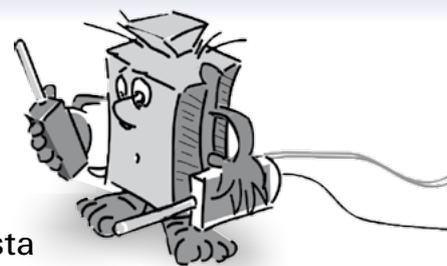
Observa, que entre el suelo y las bandas no exista una hendidura muy grande. Sino podría ser que la cámara reconozca esta hendidura como una línea y entonces el cálculo de posición ya no sería correcto.

Para evitar esto es importante que pliegues los cantos exactamente y que el campo de juego apoye sobre una superficie nivelada. Adicionalmente puedes reforzar los lados exteriores de las bandas p.ej. con libros u otros objetos, de manera que las bandas se presionen adicionalmente hacia abajo.

Cuando el balón continuamente rueda hacia una esquina o frecuentemente permanezca detenido en una banda, puedes suplementar el campo de juego en los bordes con objetos planos, p.ej. revistas, cuadernos escolares, cartones, de manera tal que el balón no se detenga en el borde sino rueda nuevamente hacia el centro. ¡De ese modo se incorpora aún más acción al juego!

## Búsqueda de errores

Si no funciona a la primera esto en la mayoría de los casos esto tiene una razón sencilla. Sólo que esta no es siempre igualmente de sencilla de encontrar. Por esta razón queremos en este punto darte algunas indicaciones sobre posibles fuentes de error.



### Prueba de interfaz

También aquí nuevamente la indicación: comprueba el funcionamiento de los componentes individuales con ayuda de la [prueba de interfaz](#) en ROBO Pro.

### Cables y cableados

Cuando un componente eléctrico no funciona de manera alguna, comprueba primero el cable, con el que los has conectado al [ROBOTICS TXT Controller](#). Conecta para ello el acumulador con un cable a lámpara de lente. Si la bombilla se enciende, el cable debería estar en orden.

También conectores mal montados (p.ej. un conector verde sobre un cable rojo) pueden ser una fuente de errores.

Comprueba también, si "+" y "-" están conectados correctamente. Compara para ello tu modelo con las imágenes de las instrucciones de construcción.

### Contacto flojo

Un componente, que alternadamente funciona y luego ya no lo hace, tiene presumiblemente en algún lugar de su cableado, un contacto flojo.

**Las causas más frecuentes para ello son:**

- Conectores de asiento flojo

Cuando los conectores de los cables asientan flojos, o sea se mueven en las hembrillas, no tienen suficiente contacto. En este caso puedes separar cuidadosamente los muelles de contacto con un destornillador en los conectores correspondientes. Sólo muy ligeramente, de manera que los conectores asienten nuevamente firmes en la hembrilla cuando los enchufes.

- Mal contacto entre cable y conector

Comprueba también los contactos entre los extremos desaislados del cable en el conector y el conector propiamente dicho. Posiblemente sea suficiente apretar un poco más los tornillos en el conector.

## Cortocircuitos

Tienes un cortocircuito, cuando una conexión positiva y una negativa se tocan. Tanto el acumulador, como también el [ROBOTICS TXT Controller](#) tienen incorporado un fusible, de manera tal que ante un cortocircuito no sean dañados. Estos simplemente cortan la alimentación de corriente durante un momento. En este caso naturalmente tu modelo tampoco funcionará.

La causa para un cortocircuito puede ser ya sea, un error en el cableado o también tornillos poco apretados en los conectores. Estos puede tocarse cuando el conector está enchufado adecuadamente y de ese modo generar un cortocircuito. Por esta razón debes enroscar siempre completamente los tornillos, de tal modo, que estos no puedan tocarse.

## Alimentación de corriente

Pequeñas paradas o motores con marcha lenta indican mayormente un acumulador descargado. En este caso debes cargar el acumulador con el cargador que se adjunta. El acumulador estará totalmente cargado, cuando el LED rojo en el cargador deje de parpadear y brille continuo.

## Error en el programa

Aun cuando a nadie le gusta admitirlo: cada uno comete errores. Ante todo en programas complejos se introduce rápidamente un error de este tipo.

Entonces cuando has comprobado todo en el modelo mismo y has subsanado todos los errores y tu modelo aún no hace lo que tú quieres, debes también comprobar aún tu programa. Verifícalo paso por paso y controla si encuentras algún error.

En modo online, o sea cuando el [ROBOTICS TXT Controller](#) está conectado con el PC, puedes seguir el programa también en la pantalla durante la ejecución. El elemento de programa activo en cada caso se resalta, de manera tal que siempre puedas ver en qué punto se encuentra justamente el programa y donde se presenta el error.

## Función de la cámara

La cámara funciona de manera óptima con la luz correcta.

**Luz insuficiente:** En la oscuridad no puede reconocer ni colores ni movimientos.

**Solución:** LEDs que iluminen el campo de visión de la cámara.

**Demasiada luz:** con luz excesiva; p.ej. por luz solar directa; se modifica el valor de color y el contraste, de manera que el reconocimiento de línea ya no detecta suficientemente.

**Solución:** evitar o reducir la radiación solar directa, p.ej. por sombreado con cortinas o persianas enrollables.

**Solución adicional:** adaptar las características de la cámara mediante software. Esto se produce en Software ROBO Pro y allí en los campos de características de los campos de sensor de la cámara. Allí p.ej. se puede configurar la sensibilidad para el reconocimiento de un objeto y ser adaptada al entorno. Los detalles puedes leerlos en [ROBO Pro Ayuda](#) en el capítulo 11.

### Los últimos comodines

Si a pesar de todo no has encontrado el error, existen aún dos posibilidades de cómo obtener ayuda:

- Comodín E-Mail:

Nos puedes enviar un E-Mail a fischertechnik, y relatarnos tu problema.

La dirección de E-Mail es [info@fischertechnik.de](mailto:info@fischertechnik.de).

- El comodín del público

Puedes visitarnos también en Internet bajo:

<http://www.fischertechnik.de>.

Allí entro otros existe un foro, donde seguramente se te podrá ayudar.

Además puedes hacerte socio sin cargo del fischertechnik Fanclub.

## ¿Y cómo sigue ahora?

### **Aún no hemos llegado al final**

¿Esto ha sido todo? No, naturalmente que no. Los experimentos y modelo que has conocido y probado en este cuaderno, sólo tienen que ser el comienzo. Dicho de otra manera, tus "primeros pasos" en el área temática enorme e interesante "ROBOTICS".

### **Fantasía**

Lo que te hemos mostrado aquí es sólo una mínima parte de las posibilidades que tienes con el ROBOTICS TXT Controller y los componentes fischertechnik. A partir de ahora está todo en tus manos. Puedes desplegar libremente tu fantasía y sencillamente construir con toda libertad.

### **Modificar lo existente**

Si aún no tienes una idea para un modelo propio completo, echa una mirada simplemente a los modelos de este cuaderno. Quizás se te ocurra algo, de lo que harías diferente en tus modelos. O modificas la función de un modelo.

### **Máquina de dibujo**

Por ejemplo en un robot de desplazamiento también se podría fijar un lápiz, que entonces puede hacerse elevar o descender y que dibuje sobre un papel grande mientras que el robot pasa sobre él. En ese caso no sólo puede desplazarse por las figuras, sino hasta delinearlas. Del robot de desplazamiento ahora se ha convertido en una máquina de delinear.

### **Carreras con amigos**

También puedes delinear una pista de recorrido propia sobre un papel con un marcador grueso de fibra y hacer circular el robot a lo largo de ella. Por otra parte, si tus amigos también tienen un ROBOTICS TXT Controller, las cosas se ponen aún más interesantes. Se pueden organizar carreras, en las que el robot más rápido sigue con maestría una pista de recorrido preestablecida. Además a través de la interfaz Bluetooth no sólo tu PC se puede comunicar con el ROBOTICS TXT Controller, sino también varios Controller entre sí. Así pueden por ejemplo programar 2 robots, que reaccionan entre sí. O bailen juntos. En los capítulos 4.5 y 7 de [ROBO Pro Ayuda](#) .

**Desprenderse  
de forma  
inalámbrica**

¿Tiene tu ordenador una interfaz WLAN? En caso afirmativo, puedes conectar entonces el ROBOTICS TXT Controller con el ordenador también a través de WLAN en lugar de un cable USB. En caso negativo, puedes comprarte un USB-WLAN-Stick y a través de él comunicar el ROBOTICS TXT Controller de forma inalámbrica con tu PC. Como funciona esto lo puedes consultar en el manual de instrucciones del ROBOTICS TXT Controller y en:

<http://www.fischertechnik.de>.

**La perseverancia  
merece la pena**

¿Entonces, a qué esperas? En marcha. Descubre y experimenta. Y no te dejes detener por pequeños reveses. Para experimentar se necesita ante todo también paciencia y dedicación. El premio es después un modelo en funcionamiento.

Te deseamos mucha diversión para la experimentación de tus propias ideas.

