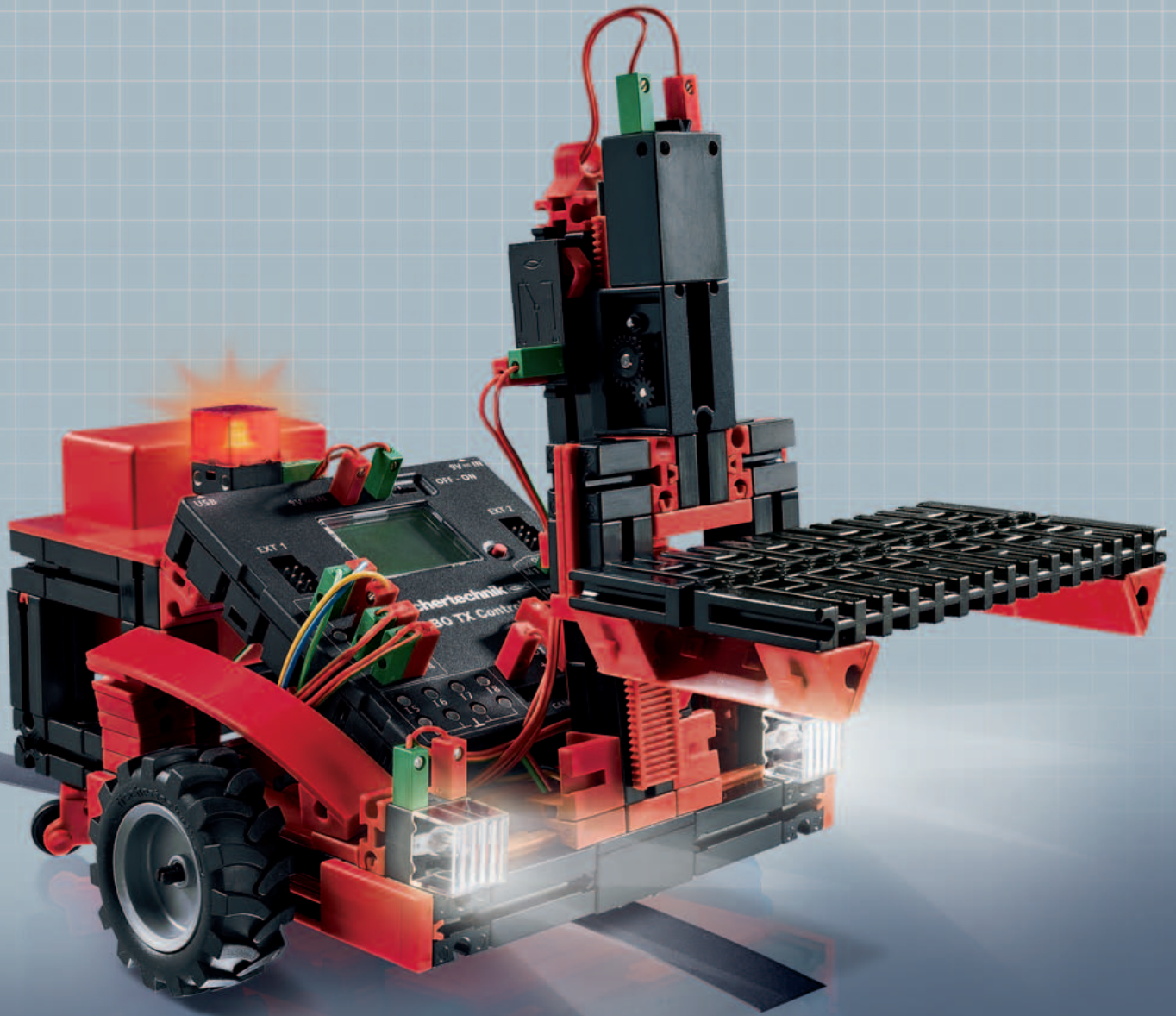


fischertechnik 

COMPUTING

Cuaderno adjunto



ROBO TX Training Lab

11 MODELS
MODEL 2

Contenido

Bienvenidos al ROBO TX Training Lab	3
Algunas informaciones generales	3
Electricidad	3
Sobre este cuaderno adjunto	3
¿Robot, el ser humano artificial?	4
Computing, (casi) todo automático	4
Explicaciones sobre elementos	4
Motores codificadores	5
Motor XS	5
Bombilla	5
Lámpara de lente	5
Fototransistor	5
Sensor de pista	6
Pulsador	6
Sensor de calor (NTC)	6
Algunas sugerencias	7
Primeros pasos	7
Modelos iniciales	8
El secador de manos	8
El semáforo	9
El elevador	10
El lavavajillas	10
Regulador de temperatura	11
Robot – el siguiente desafío	12
Modelo básico	12
El rastreador	13
La cortadora de césped	14
El robot de fútbol	15
El robot de medición	16
La carretilla elevadora	17
Si no funciona inmediatamente ...	19
¿Y como sigue ahora?	20

Bienvenidos al ROBO TX Training Lab

¡Hola!

Nos alegramos que te hayas decidido por el kit de construcción "ROBO TX Training Lab" de fischertechnik. Te prometemos, que tu interés será recompensado. Por qué 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 ROBO TX 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, aún 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 da 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 el experimentar con el ROBO TX Training Lab.

Tu equipo de

fischertechnik



Algunas informaciones generales

Antes que podamos empezar ha hacer algo con el kit de construcción, tienes que saber aún algunas cosas. Los elementos 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 elementos del ROBO TX Training Lab 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 elementos eléctricos.

De manera alguna puedes conectar el polo positivo y el negativo entre sí, o sea ponerlo en cortocircuito. De este modo el ROBO TX 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 E-Tech" como con el ROBO TX Training Lab.



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 elementos), 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 "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 en línea ROBO Pro). Estos vínculos están escritos en azul claro y subrayados.

Imágenes

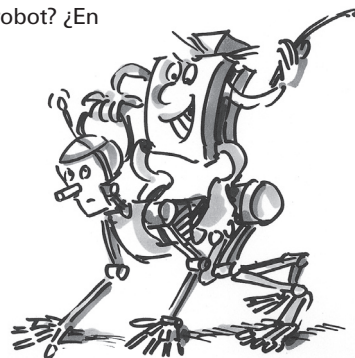
Una imagen dice más que mil palabras. Esta frase seguramente ya la has escuchado alguna vez. Como tiene mucho de verdad, tocando las palabras escritas en marrón y subrayadas puedes activar una imagen, sobre la que podrás reconocer que es lo que el texto quiere decir.

¿Robot, el ser humano artificial?

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

Existen innumerables tipos diferentes de robots. Algunos se parecen un poco a un humano, otros sólo están constituidos de uno o varios brazos. ¿Que es entonces lo que hace que un robot sea lo que es?

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



Computing, (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 "Computing".

Con el "ROBO TX Training Lab" 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 y a continuación a través de una conexión USB o Bluetooth transmitirlos al ROBO TX Controller. El Controller "controla" y comanda entonces el modelo de acuerdo a la programación que tu has creado.

Explicaciones sobre elementos

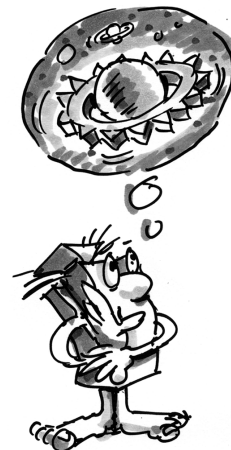
Todo está dentro del kit de construcción.

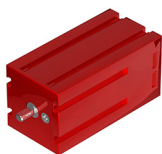
Primero encontrarás numerosos elementos fischertechnik, además de motores, lámparas y sensores, así como unas coloridas instrucciones de construcción de los diferentes modelos.

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.

Actuadores

Se denominan actuadores a todos los elementos 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 codificadores

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 se 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 hembra 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 caso 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 codificadores fischertechnik generan por rotación del árbol del motor, 3 impulsos. Además, como los motores codificadores tienen un engranaje con una [relación de transmisión de 25:1](#) (dígase "25 a 1"), una vuelta del árbol que proviene del engranaje, 75 impulsos del codificador.

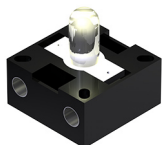


Motor XS

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.

Los dos engranajes, que asimismo se adjuntan al kit de construcción, se adaptan exactamente al motor XS.

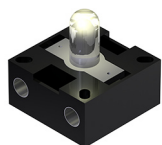
El motor XS está dimensionado para una tensión de alimentación de 9 Voltios y un consumo de corriente de máximo 0,3 Amperios.



Bombilla

Dos bombillas incandescentes están contenidas en el kit de construcción. Esta pueden ser empleadas de forma muy versátil, por ejemplo como luces de señalización en un semáforo, o también como luz intermitente en un robot.

Las bombillas están dimensionadas para una tensión de 9 Volt y consumen aprox. 0,1 Amperios de corriente.



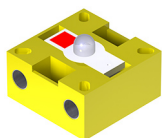
Lámpara de lente

En esta lámpara se ha incorporado una lente, que reúne la luz en un haz. Presenta un aspecto similar a una bombilla globular. Tienes que cuidar, que no las confundas. Para una mejor diferenciación el zócalo enchufable de esta lámpara es de color gris, mientras que la bombilla globular posee un zócalo blanco. Necesitas la lámpara de lente para montar una [barrera de luz](#).

Las lámpara de lente está dimensionadas como las bombillas para una tensión de 9 Voltios y consumen aprox. 0,15 Amperios de corriente.

Sensores

Los sensores son en cierta medida las contrapiezas de los actuadores. 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.



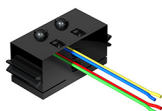
Fototransistor

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.

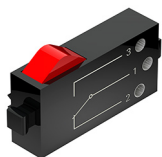
Atención:

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



Sensor de pista

El sensor de pista [infrarrojo](#) es un sensor infrarrojo digital para reconocimiento de una pista negra sobre un sustrato blanco a una distancia de 5 - 30 mm. Está constituido de dos elementos de transmisión y dos de recepción. Como conexión necesitas dos entradas digitales y la alimentación de tensión de 9 Voltios (polo positivo y negativo) en el [ROBO TX Controller](#).



Pulsador

El pulsador también se 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.

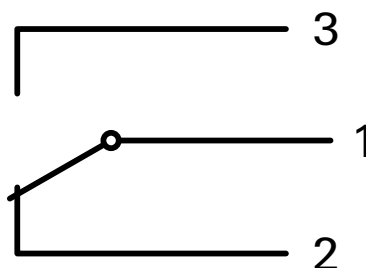
Pulsador no oprimido: no fluye ninguna corriente

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 elemento se trata de un sensor de calor, con los que se pueden medir temperaturas. A 20°C su [resistencia eléctrica](#) es de 1,5kΩ (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 [ROBO TX 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 que se interrumpa una barrera de luz.



Software ROBO Pro 2.x

ROBO Pro es una superficie de programación, con la que puedes crear los programas para el [ROBO TX 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. Un ejemplo para un programa de estas características lo encontrarás en la imagen a la derecha.

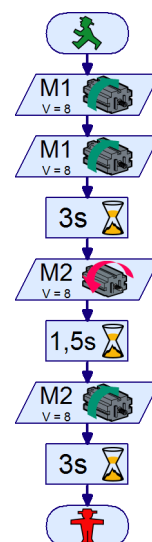
Como se crea exactamente un programa así, está descrito extensamente en los capítulos 3 y 4 de la [ayuda en línea ROBO Pro](#).

La instalación de ROBO Pro y el controlador para el ROBO TX Controller está descrito en las instrucciones de instalación, que se encuentran en el kit de construcción.

El software se encuentra en el mismo CD que este cuaderno adjunto.

Cuando hayas terminado con la instalación de ROBO Pro, puedes iniciar ROBO Pro inmediatamente, porque como siguiente necesitamos la ayuda en línea del software.

Lo mejor es que leas ahora los dos primeros capítulos de la [ayuda en línea ROBO Pro](#). En este caso aprendes enseguida a conocer un poco el software, de manera tal que a continuación podemos iniciar directamente con la experimentación.





ROBO TX Controller

El ROBO TX Controller es el núcleo de este kit de construcción Computing. Porque él controla los [actuadores](#), y evalúa las informaciones de los [sensores](#).

Para esta tarea el ROBO TX Controller dispone de numerosas conexiones a las que puedes conectar los elementos. Que elementos se pueden conectar a que conexiones y que cuales son las funciones de las conexiones, está descrito en el manual de instrucciones del ROBO TX Controller.

Una golosina especial es la interfaz Bluetooth integrada. A través de ella puedes conectar sin cables tu PC con el ROBO TX Controller. O también varios Controller con el PC y entre sí.

Como el Controller se maneja con los elementos 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 ya sabes, muchos de los elementos de ROBO TX Training Lab funcionan con corriente, necesitas naturalmente también una alimentación de corriente.

Para ello se adapta especialmente el acumulador fischertechnik Accu Set. Este no está contenido en el ROBO TX Training Lab.

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 ROBO TX Controller con ayuda de la prueba de interfaz de ROBO Pro. Como funciona esto exactamente está explicado en la [ayuda en línea ROBO Pro](#) en capítulo 2.4.



Primeros pasos

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

Como para experimentar, además de los elementos de fischertechnik también trabajarás con el [software ROBO Pro](#), como primera medida debes familiarizarte con él y aprender como se crea un programa.

Como esto está realmente explicado de forma excepcional en los capítulos 3 y 4 de la [ayuda en línea ROBO Pro](#), 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.



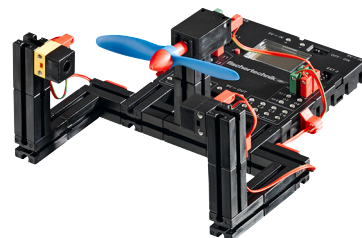
Modelos iniciales

Después que hayas leído los capítulos 3 y 4 de la [ayuda en línea ROBO Pro](#), puedes ahora programar ya algunos modelos del kit de construcción. Por esta razón nosotros también queremos poner manos a la obra. Siempre que hayas terminado de construir y cablear un modelo, comprueba con ayuda de la [prueba de interfaz](#), si en el [ROBO TX Controller](#) todas las entradas y salidas están correctamente conectadas y los [sensores](#), [motores](#) y [lámparas](#) funcionan correctamente.

El secador de manos

En tu escuela se han instalado en los aseos nuevos secadores de manos junto al 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:

- 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 "1" ([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 de inicio](#) 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:

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



Tarea 2:

- 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:

El semáforo

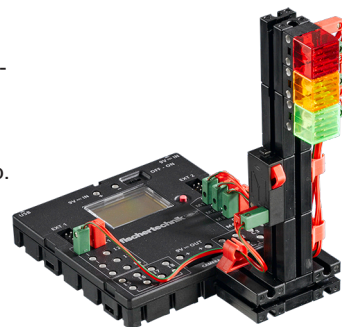
Delante de vuestra casa se instala un semáforo. Dado que el montador de la empresa instaladora está con mucha prisa, tu te ofreces asumir para él, la programación del control del semáforo.

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



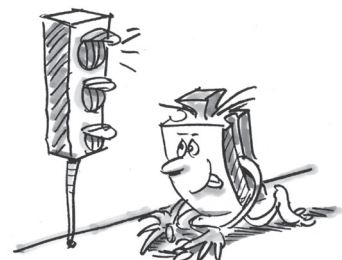
Tarea 1:

- El semáforo en principio debe estar en verde. Cuando un peatón oprime el [pulsador I1](#), el semáforo 3 segundos más tarde tiene que pasar a ámbar y tras 4 segundos a rojo. La fase roja debe durar 10 segundos, la fase roja-ámbar que sigue a continuación, 3 segundos. Entonces debe volver a estar verde.



Sugerencias de programación:

- Las diferentes lámparas perteneces a las siguientes salidas de la interfaz:
 - Rojo – M1
 - Ámbar – M2
 - Verde – M3
- 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:



Tarea 2:

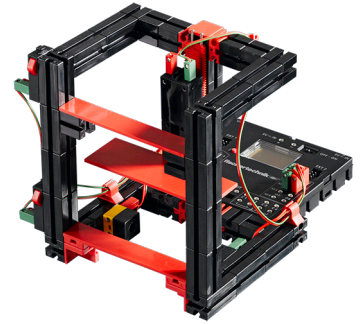
- Al día siguiente te llama el montador de la empresa instaladora del semáforo. Él ha olvidado mencionarte, que en la caja de distribución sobre la acera se encuentra un interruptor I2, que debe conmutar el semáforo a una luz ámbar intermitente en el momento de ser accionado. Tu le aseguras al montador, de integrar aún está función rápidamente en tu programa.

Sugerencias de programación:

- Fija en tu modelo de semáforo un segundo [pulsador](#) y conéctalo a la [entrada I2](#).
- Consulta con una segunda derivación la [entrada I2](#). Cuando se oprime el [pulsador I2](#), la ejecución se deriva a la luz intermitente. En caso contrario el control del semáforo se ejecuta como en la tarea 1.
- La luz intermitente la obtienes conectando y desconectando la [lámpara M2](#) en un intervalo de 0,5 segundos. Emplea para ello un subprograma. Como se crea un subprograma, puedes leerlo en el capítulo 4 de la [ayuda en línea ROBO Pro](#).
- El programa de ejemplo no encontrarás como es habitual haciendo clic sobre el símbolo. Intenta sin embargo antes de consultar, de descubrir tu mismo la solución. ¡Mucho éxito!

El elevador

Tu vecino ha instalado en su taller un elevador de cargas, para no necesitar más cargar sus envases pesados por la escalera hasta el primer piso. Ahora sólo necesita el control, que naturalmente tú le programarás con gusto.



Tarea 1:

- Programa el elevador de tal manera, que inicialmente se desplace hacia abajo a su posición de partida. En esta posición la **barrera de luz** en I3 está interrumpida. Cuando se oprime uno de ambos **pulsadores** (I1 en la planta baja o I2 en el primer piso) el elevador tiene que desplazarse en cada caso a la otra planta.

Sugerencias de programación:

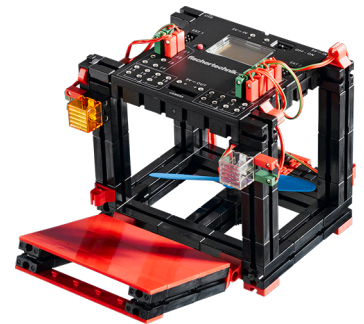
- Cuando la **barrera de luz** está interrumpida, el elevador se encuentra abajo, cuando no está interrumpida tu partes de la base de que el elevador se encuentra en el primer piso.
- Hasta donde se desplaza hacia arriba, lo controlas a través del tiempo en que el motor estás conectado.
- Aún si seguramente esto lo logras sin ayuda, para todos los casos tenemos nuevamente una propuesta de solución:

Antes que te atrevas con las siguientes tareas de programación, deberías abrir nuevamente la [ayuda en línea ROBO Pro](#). Estudia allí cuidadosamente el capítulo 5. Conmuta en ROBO Pro al nivel 3. Lentamente las tareas de programación se tornan algo más exigentes. Emplearemos entradas analógicas, elementos de mando, operadores y variables. Pero si lees atentamente la ayuda en línea ROBO Pro, más tarde te resultará fácil trabajar con ello.

El lavavajillas

Como siguiente quieres atreverte con la programación de un lavavajillas. El lavavajillas debe tener las siguientes funciones:

- Pulsador** para conectar y desconectar (I1)
- Pulsador**, que detecta si la puerta está cerrada (I2)
- Función de lavado (hélice en M1)
- Función de secado (lámpara **roja** en M2)
- Indicación que la máquina está conectada (lámpara **naranja** en M3)
- Indicación en que estado de servicio se encuentra la máquina (lámpara **transparente** en M4)
 - Parpadeo rápido: la máquina lava
 - Parpadeo lento: la máquina seca
 - Luz continua: la máquina ha terminado



Tarea 1:

- Confecciona un programa de lavado, que recién comience cuando la puerta está cerrada y se oprime el botón de arranque I1. Como primero se lava, a continuación se seca. El estado de servicio tiene que ser indicado a través de ambas lámparas M3 y M4.

Sugerencias de programación:

- Aquí nuevamente nuestra propuesta de solución:


Tarea 2:

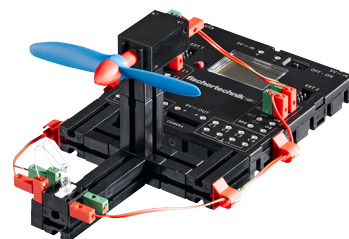
- Cuando la puerta se abre, el procedimiento de lavado debe ser interrumpido. Tras el cierre de la puerta el programa continua en aquel punto en donde fue interrumpido.
- Muestra adicionalmente el estado de servicio del lavavajillas en el display del [ROBO TX Controllers](#).

Sugerencias de programación:

- Hay que reconocer que esta tarea la tiene consigo. Si la puedes solucionar sólo: ¡Fantástico! Pero para el caso que después de todos los razonamientos no consigues avanzar, ningún problema. Mira simplemente una vez en nuestra propuesta de solución.

Regulador de temperatura

En vuestra casa se ha instalado un nuevo equipo de aire acondicionado. Naturalmente le has preguntado inmediatamente al instalador, como funciona el regulador de temperatura. El te ha explicado gustoso, que un sensor de temperatura mide permanente la temperatura actual. En el momento que se sobrepasa el valor límite superior, se conecta la refrigeración. Si por el contrario 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.

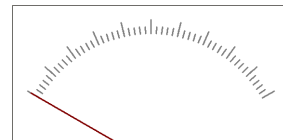

Tarea 1:

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 I2](#).

- 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.
- El valor actual de la entrada analógica debe ser emitido a un instrumento de medición y a una indicación de texto en ROBO Pro, así como en el display del [ROBO TX Controllers](#).

Sugerencias de programación:

- **Observa:** El valor de resistencia de la [resistencia NTC](#) descende con el aumento de la temperatura. El valor límite superior de temperatura es entonces el menor valor de I2. A este valor límite se debe conectar el soplador. El valor límite inferior de temperatura es el mayor mayor de I2. A este valor límite se debe conectar la calefacción.
- Que valor posee I2 a temperatura ambiente, lo puedes descubrir con la [prueba de interfaz](#). Tú conectas la lámpara M2 y observas, hasta donde descende 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.
- Muestra el valor de la entrada analógica en tu programa con una indicación de texto y/o con un instrumento de medición (consulta también [ayuda en línea ROBO Pro](#) capítulo 8.1).



Temp. = 0

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

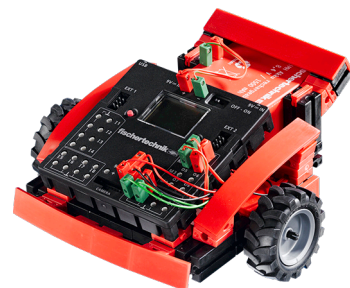


Robot – el siguiente desafío

Modelo básico

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.



- Monta el robot tal como está descrito 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 elementos con el [ROBO TX 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 elementos conectados al [ROBO TX 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.



Tarea 1: Simplemente recto (nivel 1)

- 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:

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

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. ¿Como funciona esto? Muy sencillo:



Tarea 2: Circular también en una curva (nivel 1)

- 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 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 la imagen a la derecha:



Tarea 3: Recorrer una figura (nivel 2)

- 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:

¿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. Aún 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 codificadores](#).



Tarea 4: Utilizar motores codificadores

- Repite las últimas tres tareas, y emplea en lugar de los [elementos de salida del motor](#) y de [tiempo de espera](#) los [elementos del motor codificador](#). Como se emplean está explicado en la [ayuda en línea ROBO Pro](#) en capítulo 11.6.

Sugerencia de programación:

- Con el [elemento de motor 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:

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 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.

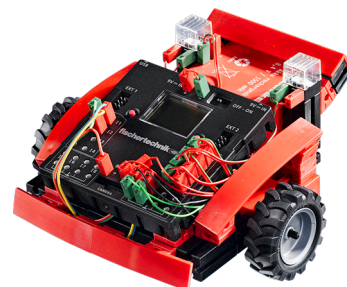
El rastreador



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

Pero un robot en realidad debería en lo posible reaccionar de manera autónoma. Por esta razón le queremos dar algo, a lo cual pueda reaccionar: Una línea negra en el suelo como marcación.

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 modificar tu modelo básico a un rastreador. Como se hace estos, se encuentra naturalmente en las instrucciones de construcción.

Después que has finalizado la modificación de tu modelo, debes comprobar con la [prueba de interfaz](#), si todos los elementos están correctamente conectados al [ROBO TX Controller](#) y funcionan. Puedes comprobar el [sensor de pistas](#), sujetándolo sobre la pista de recorrido y lo mueves lateralmente. Las señales en las entradas, en las que están conectadas el [sensor de pistas](#), en este caso tendrían que modificarse.

Recuerda ajustar las entradas en [prueba de interfaz](#) a "Digital 10V (Sensor de pista)".


Tarea 1: Reconocer una pista (nivel 2)

- 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 y ambas lámparas deben parpadear 3 veces en cada caso. Emplea para esta tarea el recorrido 1a del kit de construcción.

Sugerencias de programación:

- Primero consultas ambas entradas del [sensor de pista](#). Si ambas entradas reciben la señal "0", el robot se encuentra sobre la pista negra. Puedes ponerlo en marcha.
- La función intermitente la ubicas en un subprograma.
- Emplea para el desplazamiento en línea recta nuevamente el [elemento de motor codificador](#), no obstante sin introducción de distancia.
- El programa terminado lo encontrarás aquí:

Ahora tu robot puede reaccionar. Sin embargo, la función aún está bastante restringida. Sería mejor, si el robot en lugar de detenerse corrigiera su dirección para seguir la pista.


Tarea 2: Seguir la pista (nivel 2)

- Amplia tu programa en la función, que el robot al abandonar la pista corrija adecuadamente su dirección y siga la pista. Prueba esta tarea primero con el recorrido 1a y luego con el recorrido 1b.

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:

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 (nivel 2)

- Escribe un subprograma "buscar", con el que el robot busque una pista, 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.

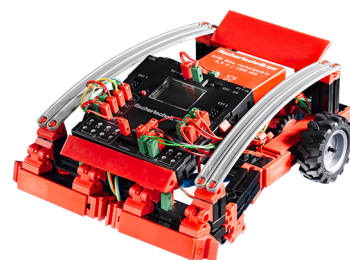
Sugerencias de programación:

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

La cortadora de césped

¿Robots pueden cortar el césped? Naturalmente. Sólo se le tiene que decir, como deben contornar los obstáculos y donde termina el césped. Así ya se puede transferir el trabajo al robot y pasar la tarde en la piscina.

Pero antes que lleguemos a ello, primero tienes que montar el robot cortadora de césped según las instrucciones de construcción. A continuación debes comprobar nuevamente con la [prueba de interfaz](#), si todo funciona como debe.



Entonces puedes pasar a la programación. El césped está simbolizado por la superficie blanca de nuestro estadio de fútbol de recorrido 1b, que está limitada por un borne negro. Este borde no puede ser atravesado por la cortadora de césped (por que en caso contrario por ejemplo terminaría en la pista de atletismo de Tartan). En caso que en el césped se presentara un obstáculo, el robot debe desviarse si se desliza contra él. Además el mecanismo cortador debe ser desconectado cuando se detecta un obstáculo.



Tarea 1: Reconocer límites y obstáculos y desviarse (nivel 2)

- Programa la cortadora de césped de tal manera, que desde su posición de partida dentro de los límites) avance en línea recta hasta que tropieza con un obstáculo o alcanza el límite del césped (línea negra).
- Cuando la cortadora de césped tropieza con un obstáculo (parachoques delantero), debe detenerse inmediatamente, parar el mecanismo cortador, retrocede un pequeño tramo, girar hacia la izquierda, nuevamente desplazarse hacia delante y volver a conectar el mecanismo cortador. Esta función la logras en el subprograma "desviar".
- Cuando la cortadora de césped alcanza la limitación del césped, debe asimismo detenerse inmediatamente e iniciar el subprograma "desviar".

Sugerencias de programación:

- Nuestra propuesta de solución:

Dependiendo del tamaño de tu "césped" podrían producirse los siguientes problemas: Según como se ajuste la duración del "viraje", el robot sólo se desliza a lo largo del borde, o eventualmente siempre se vuelve a detener en la misma área del "césped". Por esta razón, el robot al desviarse, siempre debe comportarse un poco diferente.



Tarea 2: La casualidad (nivel 3)

- Modifica el programa de la cortadora de césped, de manera que esta cada vez al desviarse asuma otro ángulo para el giro. Esta debe girar entonces una vez más y una vez menos. Además debe desviarse de un obstáculo que reconoce con el parachoques derecho, hacia la izquierda y viceversa.

Sugerencias de programación:

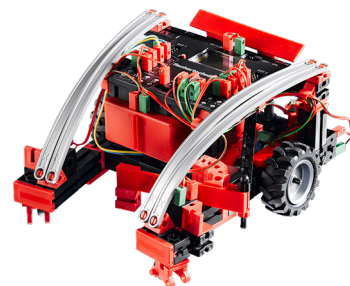
- Para poder utilizar variables en ROBO Pro, tienes que ajustar el software a "nivel 3".
- Para ello necesitas un **generador aleatorio**. Esto lo puedes ajustar por ejemplo, haciendo contar el valor de una variable por un **bucle de conteo** siempre de nuevo desde "0" a un determinado valor. Para "desviar" en el subprograma insertas tras el comando para el giro de la cortadora de césped un "Aguardar a..." Elemento, que aguarda hasta que la variable se ponga a "0". Debido a que posiblemente la variable cada vez que se consulta tiene otro valor, la duración es siempre diferentemente larga, hasta que nuevamente sea "0". De ello resultan diferentes tiempo al girar el robot.
- Debemos reconocer, que esto no es del todo sencillo. En caso que después de todos los razonamientos no llegues a una solución, naturalmente hay una propuesta de solución de nuestra parte:

El robot de fútbol

¿Haz 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 informaciones las encontrarás por ejemplo en la página principal de Robo-Cup <http://www.roboocup.org>.

En las instrucciones de construcción encontrará una propuesta para el montaje de un robot de fútbol. Él es tan maniobrable con nuestros otros robots, pero tiene adicionalmente aún una barrera de luz para reconocer un balón y un "mecanismo de disparo". Procede a montarlo, a continuación 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 puede emplear por ejemplo una bola de tenis (no contenida en el kit de construcción) Según que tipo de bola emplees, deberás quizás adaptar algo el mecanismo de disparo.




Tarea 1: "Ha atrapado el balón, y dispara" (nivel 2)

- En el primer paso nuestro artista electrónico del balón tiene que aprender a reconocer el balón y reaccionar a ello. Prográmalo de tal manera, que dispare el balón, en el momento que este sea 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.
- Entonces debe aprender a "disparar penaltis". Coloca el balón sobre el punto de penalti del recorrido 1a. Coloca el robot en el inicio de la línea negra. Él debe entonces tomar velocidad a lo largo de la línea y en el momento que reconoce el balón, disparar este a la portería. Al final de la línea debe parar y virar.

Sugerencias de programación:

- 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 entrenado no siempre se lo tiene fácil. En caso que tu jugador robot no te haga caso, quizás lo puedas convencer con nuestra propuesta de programa:

Pero como un verdadero artista del balón tiene que sabre más cosas que sólo disparar penaltis, queremos ampliar un poco aún las habilidades de nuestro robot de fútbol.


Tarea 2: El que busca, encuentra. (nivel 2)

- El robot de fútbol debe ahora desplazarse dentro del estadio (recorrido 1b) y en ese caso no salirse de las líneas de limitación. Si encuentra el balón, debe dispararlo, naturalmente en lo posible a la portería.

Sugerencias de programación:

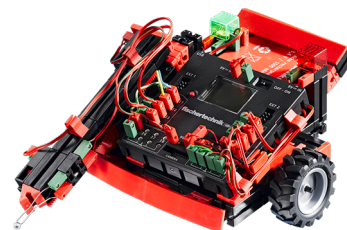
- La mayoría de las funciones para este programa ya las has programado en la [cortadora de césped](#). Puedes entonces emplear el programa de la cortadora de césped, guardarlo bajo otro nombre y ampliarlo con las funciones del robot de fútbol.
- Probablemente no lo necesites, pero a pesar de todo también aquí hay nuevamente una propuesta de solución:

El robot de medición

¿Hace más calor debajo de tu cama que debajo del escritorio? ¿Cuanto calor emana de una vela? ¿Se puede enfriar la habitación con un cubo de hielo?

Para estas (y otras) preguntas, podrás encontrar respuesta con ayuda del robot de medición. Él está equipado con un [sensor de temperatura \(NTC\)](#) y puede medir e indicar la temperatura en diversos puntos. Adicionalmente el robot de medición también tiene un [sensor de pista](#), de manera tal que puedes especificarle su trayecto a través de una línea negra.

Monta primero el robot de medición como se describe en las instrucciones de construcción y comprueba sus funciones con la [prueba de interfaz](#).


Tarea 1: Control y medición de temperatura (nivel 3)

- Toma el programa del [rastreador](#) y amplíalo con un subprograma para controlar el brazo de medición. El robot debe desplazarse sobre el recorrido 1b y en determinados intervalos, medir la temperatura. La temperatura debe ser indicada en el display del [ROBO TX Controllers](#).

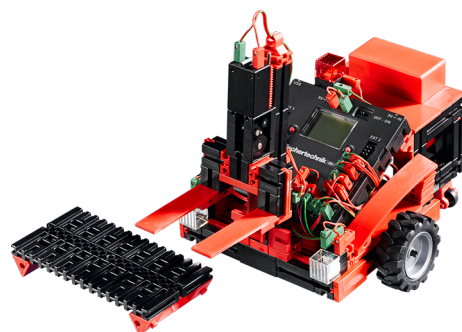
Sugerencias de programación:

- Para indicar el valor de resistencia del [sensor de temperatura](#) utiliza una [entrada analógica](#), una [salida de cuadro de mando](#) y un [elemento de mando "Indicación"](#).
- En la entrada analógica se indica el valor de resistencia del [sensor de temperatura](#), no la temperatura. Para convertir este valor a una indicación de temperatura, puedes emplear un subprograma.
- Nuestra completa propuesta de solución para esta tarea la encontrarás aquí:

La carretilla elevadora

Un área, en que el robot se emplea preferentemente en la industria, es la logística. O sea, en todo lugar donde deben moverse objetos, de "A" hacia "B", como se acostumbra a decir.

Estas tareas de transporte las puedes reproducir fantásticamente con el modelo de carretilla elevadora. Monta el modelo tal como está descrito en las instrucciones de construcción. Comprueba a continuación con la [prueba de interfaz](#), si los elementos funcionan todos correctamente.


Tarea 1: Arriba y abajo (nivel 3)

- Escribe para las funciones "elevar" y "descender" un subprograma para cada caso.
- Debido a que durante la marcha la horquilla no puede estar demasiado arriba o demasiado abajo, necesitas aún un subprograma "posición de desplazamiento".

Sugerencias de programación:

- Al "elevar" la horquilla debe ser llevada hacia arriba (motor M3 gira hacia la izquierda), hasta que se active el final de carrera superior I4.
- Al "descender" la horquilla debe ser llevada hacia abajo (motor M3 gira hacia la derecha), hasta que el engranaje en el final de carrera inferior I3 haya pasado y el interruptor esté nuevamente abierto. Para ello empleas un [elemento "Aguardar a"](#) y lo ajustas a "1 -> 0 (descendiente)".
- Para la "posición de desplazamiento" la horquilla debe desplazarse tanto hacia arriba, hasta que esta se encuentre apenas sobre el final de carrera inferior.
- Como siempre hay por nuestra parte una propuesta de solución para la tarea:

Después que toda la mecánica ahora funciona correctamente, también la queremos utilizar. Finalmente una carretilla elevadora debe transportar objetos y no sólo poder mover su horquilla.


Tarea 2: De "A" hacia "B" (nivel 3)

- En el kit de construcción hay un recorrido 2a para la carretilla elevadora. Esta tiene que partir desde su posición de arranque, tomar una paleta que se encuentra en el campo "A", transportarla sobre la pista al campo "B" y allí descargarla.

Sugerencias de programación:

- Programa la carretilla elevadora de tal manera, que siempre cuando la pista ha finalizado, la paleta carga o descarga. Utiliza para estos dos procedimientos un subprograma en cada caso.
- Desplaza la carretilla elevado tan despacio, que bajo ningún concepto pierda involuntariamente la pista. Sino el considera que ha llegado al final y que tiene que cargar o descargar la paleta.
- Nuestra propuesta de solución:



Tarea 3: Reconocer bifurcaciones

Ahora se torna interesante. La carretilla elevadora tiene que reconocer una bifurcación. Utiliza para ello el recorrido 2b.

- La carretilla elevadora arranca desde su posición inicial. Ella tiene que desplazarse al campo "A", allí recoger una paleta y transportarla al campo "B". A continuación debe desplazarse de retorno a su posición inicial. Mientras que ella sigue la pista, tienen que brillar sus dos reflectores. Si gira o marcha en reversa, debe brillar la lámpara roja.

Sugerencias de programación:

- La interrupción de la pista la puede reconocer la carretilla elevadora como derivación. Una vez que ha alcanzado este punto, puedes decidir como debe continuar su marcha: Puede desplazarse en línea recta hasta que encuentre una pista, doblar hacia la izquierda y a la vez buscar la pista o doblar hacia la derecha hasta que encuentre nuevamente la pista. Crea para estos 3 casos un subprograma para cada uno.
- Para volver a encontrar la pista al girar hacia la izquierda o hacia la derecha, el robot debe primero desplazarse un pequeño tramo en línea recta. Sino girar por un lado de la pista.
- En caso que hayas perdido la pista durante la programación, mira una vez nuestra propuesta:



Tarea 4: Transportar sin fin

- Amplia el programa de la tarea 3 de tal manera, que la carretilla elevadora tras el retorno a su posición inicial realice una pequeña pausa y luego busque la paleta del campo "B", la traiga de vuelta a "A" y a continuación se desplace nuevamente al inicio. La carretilla debe repetir permanentemente el desarrollo completo como en una cinta de montaje.

Sugerencias de programación:

- Crea un bucle sin fin llevando una línea de retorno desde el último elemento de programa hasta el inicio del mismo.
- Aún cuando la tarea seguramente no sea un problema para ti, aquí está nuestra propuesta de solución:

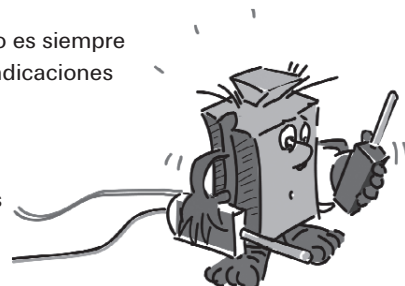


Si no funciona inmediatamente ...

... en la mayoría de los casos esto tiene una solución sencilla. Sólo que esta no es siempre 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 elementos individuales con ayuda de la [prueba de interfaz](#) en [ROBO Pro](#).



Cables y cableados

Cuando un elemento eléctrico no funciona de manera alguna, comprueba primero el cable, con el que los has conectado al [ROBO TX Controller](#). Conecta para ello el [acumulador](#) con un cable a una [bombilla](#). 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 elemento, 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 el contacto entre los extremos de cable [desaislados](#) en el conector y el conector mismo. Posiblemente sea suficiente apretar algo más firmemente los tornillos en el conector.

Cortocircuitos

Tienes un cortocircuito, cuando una conexión positiva y una negativa se tocan. Tanto el [acumulador](#), como también el [ROBO TX 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. Cuando el LED rojo en el cargador deja de parpadear y brilla permanentemente, el acumulador está completamente cargado.

Error en el programa

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

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

En modo en línea, o sea cuando el [ROBO TX 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.

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.

- **El comodín del público**

Puedes visitarnos también en Internet bajo <http://www.fischertechnik.de>. Allí entre otros existe un foro, donde seguramente se te podrá ayudar. Además puedes hacerte socio sin cargo del fischertechnik Fanclub.

¿Y como sigue ahora?

¿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 "Computing".

Lo que te hemos mostrado aquí es sólo una mínima parte de las posibilidades que tienes con el [ROBO TX Controller](#) y los elementos fischertechnik. A partir de ahora está todo en tus manos. Puedes desplegar libremente tu fantasía y sencillamente construir con toda libertad.

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.

Por ejemplo en una [carretilla elevadora](#) en lugar de la horquilla se podría también 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 desplazarte por las figuras, sino hasta delinearlas. De la carretilla elevadora ahora se ha hecho una máquina de delinear.

Además, si tus amigos también tienen un [ROBO TX Controller](#), las cosas se ponen aún más interesantes. Porque a través de la interfaz Bluetooth no sólo tu PC se puede comunicar con el [ROBO TX Controller](#), sino también varios Controller entre sí. Así pueden por ejemplo programar 2 robots, que reaccionan entre sí. O bailen juntos. En el capítulo 7 de la [ayuda en línea ROBO Pro](#) encontrarás algunas informaciones interesantes sobre este tema.

¿Tiene tu ordenador una interfaz Bluetooth? En caso afirmativo, puedes conectar entonces el [ROBO TX Controller](#) con el ordenador también a través de Bluetooth en lugar de un cable USB. En caso negativo, puedes comprarte un USB-Bluetooth-Stick y a través de él comunicar el [ROBO TX Controller](#) de forma inalámbrica con tu PC. Como funciona esto lo puedes consultar en el manual de instrucciones del [ROBO TX Controllers](#) y en <http://www.fischertechnik.de>.

¿Entonces, a que 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.

