

# Actuadores

- Salidas
- Sonido
- Visualización
- Motor

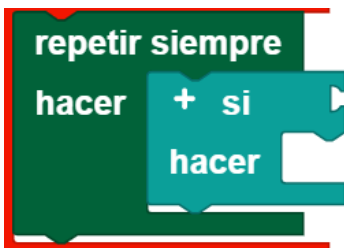
# Salidas

## Arranque de cada bloque

El **arranque de cada bloque** ofrece la posibilidad de ejecutar un programa si se cumple una condición. Funciona de manera similar a la distinción de casos, pero no se ejecuta solo una vez, sino cada vez que se cumple la condición durante todo el transcurso del programa. **Arranque de cada bloque:**



Es una abreviatura del siguiente constructo:



Se pueden utilizar todas las condiciones de esta categoría en el **Arranque de cada bloque** de la categoría salidas.

**Nota:** La sección del programa dentro de Arranque de cada bloque debe ser corta y no contener llamadas de bloqueo ni bucles sin fin, de modo que esta parte del programa se pueda procesar rápidamente.

## Luces LED



## Configuración

Con los bloques **configuración LED ...** y **configuración del brillo LED ...** se puede encender y apagar el LED o configurar su brillo en un valor determinado (desde 0 hasta 512).

## Acceso

Con el bloque **obtención de brillo LED** se puede acceder al brillo de un LED y procesarlo como valor.

## Consulta

Con los bloques **el LED está ...** y **es el brillo del LED ...**, se puede configurar la actividad o el brillo de un LED como un estado. En el ejemplo, el brillo del LED se establece en 512, a menos que ya tenga ese brillo.



## Motores

El icono en los bloques del motor representa todos los motores que no son codificadores ni servomotores.

### Configuración

Con el bloque **configuración de la velocidad del motor en [] ...** se puede configurar la velocidad de un motor en un valor determinado (desde 0 hasta 512).

### Acceso

Con el bloque **obtención de la velocidad del motor** se puede acceder a la velocidad del motor y procesarla como valor.

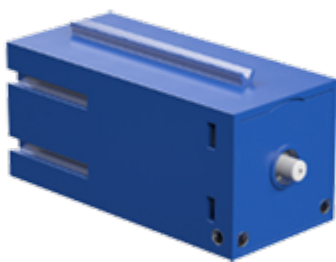
### Consulta

Con los bloques **el motor está funcionando** y **la velocidad del motor es ...** se puede configurar la actividad o la velocidad de un motor como un estado.

### Detención

Con el bloque **detención del motor ...** es posible parar un motor.

## Compresor



### Configuración

Con el bloque **configuración del compresor []**, se puede activar o desactivar el compresor.

### Consulta

Con el bloque **el compresor está []**, se puede configurar la actividad de un compresor como un estado.

## Válvula magnética



## Configuración

Con el bloque **configuración de la válvula magnética** [], se puede activar o desactivar la válvula magnética. «Activado» significa que la válvula está abierta y «desactivado» que la válvula está cerrada.

## Consulta

Con el bloque **la válvula magnética está** [], se puede configurar la actividad de una válvula magnética como un estado.

# Sonido

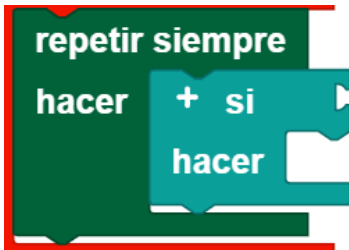
El TXT 4.0 Controller tiene un altavoz incorporado y, por tanto, permite reproducir sonido.

## Arranque de cada bloque

El **arranque de cada bloque** ofrece la posibilidad de ejecutar un programa si se cumple una condición. Funciona de manera similar a la distinción de casos, pero no se ejecuta solo una vez, sino cada vez que se cumple la condición durante todo el transcurso del programa. **Arranque de cada bloque:**



Es una abreviatura del siguiente constructo:



Se pueden utilizar todas las condiciones de esta categoría en el **Arranque de cada bloque** de la categoría Sonido.

**Nota:** La sección del programa dentro de Arranque de cada bloque debe ser corta y no contener llamadas de bloqueo ni bucles sin fin, de modo que esta parte del programa se pueda procesar rápidamente.

## Reproducción de sonido

### Archivos de audio preinstalados

Con el siguiente bloque puede reproducir uno de los 29 sonidos preinstalados. El archivo de audio deseado se puede seleccionar utilizando el menú desplegable (triángulo pequeño). También es posible reproducir el sonido en bucle continuo. Para hacer esto, ha de marcar la casilla que hay detrás del icono de bucle.



### Archivos de audio propios

Si se desea reproducir un sonido propio, se puede utilizar el bloque



. Para incluir su propio archivo de audio en el bloque, tiene que:

1. Estar conectado al controlador
2. Introducir la dirección IP del controlador en el navegador (la IP que también se utilizó para realizar la conexión al controlador tiene que seleccionarse aquí)
3. Introducir USER: ft, PASSWORD: fischertechnik en la página a la que se ha accedido
4. Abrir la carpeta de sonidos y utilizar el signo + para cargar el archivo de audio deseado en el controlador (importante: el archivo de audio ha de estar en formato wav)
5. Especificar en el bloque ROBO Pro Coding la ruta «./dateiname.wav»

Aquí también existe la opción de reproducir el sonido en bucle continuo.

## Consulta

Para consultar si se está reproduciendo un archivo de audio, se utiliza el bloque **reproducir sonido**. Este se puede utilizar como estado en el programa.

## Detención

Para detener un sonido, se utiliza el bloque **detención de la reproducción del sonido** en el programa.

# Visualización

Con los bloques de la categoría Pantalla, se puede configurar y optimizar la pantalla del TXT 4.0 Controller. Esto se realiza en dos pasos:

1. Configurar, es decir
  - Abrir un nuevo archivo en la categoría Pantalla mediante el icono de Página utilizando el signo + que se encuentra en la parte superior izquierda
  - Arrastrar los elementos deseados hasta la zona de la cuadrícula (representa la parte configurable de la pantalla)
  - Ajustar las especificaciones, si es necesario.
2. Programar, es decir
  - En el programa principal, utilizar los bloques de la categoría Pantalla para programar el efecto de interacción con la pantalla.

## Bloques

### Consulta de evento

El bloque **Evento** [] permite acceder al valor de retorno de un elemento. Este bloque solo se puede utilizar en los programas de eventos. En estos programas de eventos, el bloque hace referencia de manera automática al evento del programa en que se utiliza. El tipo adecuado de valor de retorno se puede seleccionar utilizando el menú desplegable (triángulo pequeño):



### Campo de etiquetado

Con el campo de etiquetado de elementos puede colocar texto en la pantalla. El icono en el configurador de pantalla es la etiqueta. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del campo de etiquetado en píxeles,
- la posición del campo de etiquetado en píxeles (la esquina superior izquierda del campo de texto se encuentra en el punto especificado),
- el nombre del campo de etiquetado y
- el contenido del campo de etiquetado (este texto aparece cuando se enciende la pantalla)

Con el bloque **configuración del texto del campo de etiquetado ...** el texto mostrado se puede cambiar mientras se esté ejecutando el programa.

## Entradas

El elemento **Entrada** permite que los usuarios inserten texto utilizando el controlador. El icono correspondiente en el configurador de pantalla es la letra «T». Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del campo de entrada en píxeles,
- la posición del campo de entrada en píxeles (la esquina superior izquierda del campo de entrada se encuentra en el punto especificado),
- el nombre del campo de entrada y
- el contenido del campo de entrada (este texto aparece cuando se enciende la pantalla)

Con el bloque **configuración del texto del campo de entrada ...** el texto mostrado se puede cambiar mientras se esté ejecutando el programa.

## Programa de entrada

El programa de entrada se ejecuta cuando se ha completado una entrada. Se escribe de manera independiente al programa principal. Las variables funcionan a nivel global en ambos programas. El programa de entrada se ejecuta en el bloque **en la entrada completa**. El bloque de **Eventos []** se configura en «texto», en el programa de entrada. En este ejemplo, la variable **Nombre** se configura a partir del texto introducido; posteriormente se usa en el programa principal para generar el texto **especificado** :



## Instrumento de medición

La función del instrumento de medición puede mostrar valores (no inferiores a 1). El icono correspondiente en el configurador de pantalla es el escalado. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del instrumento de medición en píxeles,
- la posición del instrumento de medición en píxeles (la esquina superior izquierda del medidor se encuentra en el punto especificado),
- el nombre del instrumento de medición,
- la orientación del instrumento de medición
- el rango de valores representado por el instrumento de medición, y
- el valor del instrumento de medición que se muestra cuando se enciende la pantalla

Con el bloque **configuración del instrumento de medición en el valor ...** el instrumento de medición se puede configurar según el valor introducido. Este valor debe estar en el rango de valores definido previamente. Si el valor se encuentra fuera del rango de valores, se muestra uno de los límites del rango de valores, dependiendo de si el valor es demasiado grande o demasiado pequeño.

## Indicador de estado



El indicador de estado muestra la actividad de un elemento. Dependiendo del estado, se enciende («activo») o no («inactivo»). El icono en el configurador de pantalla es un diodo incandescente. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del indicador de estado en píxeles,
- la posición del indicador de estado en píxeles (la esquina superior izquierda del indicador de estado se encuentra en el punto especificado),
- el nombre del indicador de estado,
- el color del indicador de estado y
- si el indicador de estado debe estar activo o inactivo al principio

Con el bloque **configuración del indicador de estado en activo** [] se puede activar o desactivar el indicador de estado. En el menú desplegable (triángulo pequeño) puede seleccionar si el indicador de estado debe configurarse como activo o inactivo.

## Control deslizante

El control deslizante arroja valores según su posición. El usuario puede cambiar la posición utilizando la pantalla táctil. Se puede acceder al valor a través del bloque **Evento** [], en el momento en que el control deslizante esté inactivo. El valor obtenido es un número decimal. Si desea que el valor del control deslizante sea un número entero, ha de utilizar el **bloque** circular. El icono correspondiente del control deslizante es la línea con el círculo. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del control deslizante en píxeles,
- la posición del control deslizante en píxeles (la esquina superior izquierda del control deslizante se encuentra en el punto especificado)
- el nombre del control deslizante,
- la actividad del control deslizante,
- la orientación del control deslizante,
- el rango de valores cubierto por el control deslizante y
- el valor en el que se encuentra el controlador cuando se enciende la pantalla

Con el bloque **Configuración del valor del control deslizante ...** puede mover el control deslizante a un valor diferente.

Mediante la **configuración del control deslizante en activado** [] se puede cambiar la actividad utilizando el menú desplegable (triángulo pequeño).

## Programa del control deslizante

El programa del control deslizante se ejecuta después de mover el control deslizante. Se escribe de manera independiente al programa principal. Las variables funcionan a nivel global en ambos programas. El programa del control deslizante se ejecuta en el bloque **cuando se mueve el control deslizante**. El bloque de **Eventos** [] se configura en «value», en el programa del control deslizante. En este ejemplo la velocidad del motor se controla mediante el control deslizante. El valor del control deslizante ha de redondearse, ya que el motor solo acepta números enteros para el número de revoluciones:




# Botón

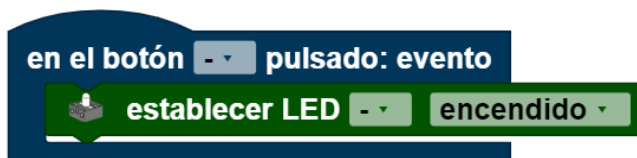
El botón es un campo etiquetado que se puede pulsar. Si pulsa el botón, el programa de botones se ejecuta en cuanto se suelta el botón. El icono asociado al botón es el cuadrado con la etiqueta «OK». Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del botón en píxeles,
- la posición del botón en píxeles (la esquina superior izquierda del botón se encuentra en el punto especificado),
- el nombre del botón,
- el texto que aparece en el botón y
- la actividad del botón

Con el bloque **configuración del botón en activado**  se puede cambiar la actividad utilizando el menú desplegable (triángulo pequeño).

## Programa de botones

El programa de botones se ejecuta cuando se deja de pulsar el botón. Se escribe de manera independiente al programa principal. Las variables funcionan a nivel global en ambos programas. El programa de botones se ejecuta en el bloque **cuando se hace clic en el botón**. El bloque **Eventos**  no se puede utilizar con el programa de botones, ya que el botón no tiene valor de retorno. En este ejemplo se activa el LED cuando se pulsa el botón.



## Interruptor

El interruptor tiene dos posiciones y siempre se encuentra en una de ellas. Según la posición, muestra **verdadero** o **falso**. El icono correspondiente del interruptor es el óvalo con el punto. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño del interruptor en píxeles,
- la posición del interruptor en píxeles (la esquina superior izquierda del interruptor se encuentra en el punto especificado),
- el nombre del interruptor,
- el texto que aparece al lado del interruptor,
- la actividad del interruptor y
- el estado en el que se debe encontrar el interruptor al iniciar el programa

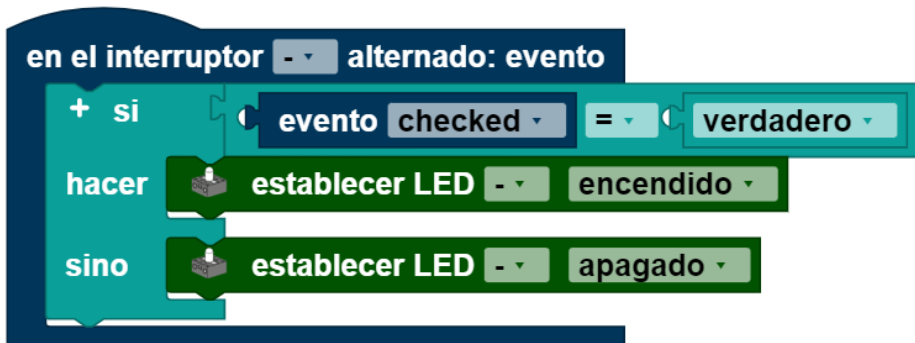
El bloque



desempeña dos funciones. Se puede establecer la actividad (seleccione enabled en el menú desplegable) o el estado (seleccione checked en el menú desplegable) como **verdadero** o **falso**.

## Programa del interruptor

El programa del interruptor se ejecuta cada vez que se activa el interruptor. Se escribe de manera independiente al programa principal. Las variables funcionan a nivel global en ambos programas. El programa del interruptor se ejecuta en el bloque **cuando se conecta el interruptor**. El bloque de **Eventos []** se establece en «checked» en el programa del interruptor, y muestra **verdadero** o **falso**. Este programa de ejemplo enciende el LED cuando se acciona el interruptor; de lo contrario, el LED se apaga:



## Casilla de verificación

La casilla de verificación puede adoptar dos estados y siempre se encuentra en uno de ellos. Según el estado, muestra **verdadero** o **falso**. El icono de la casilla de verificación es el cuadrado con el tick. Si arrastra este icono hasta la zona de la cuadrícula, se abre una ventana a la derecha. Aquí, en Inspector, se puede determinar

- el tamaño de la casilla de verificación en píxeles,
- la posición de la casilla de verificación en píxeles (la esquina superior izquierda de la casilla de verificación se encuentra en el punto especificado),
- el nombre de la casilla de verificación,
- el texto que aparece al lado de la casilla de verificación,
- la actividad de la casilla de verificación y
- el estado en el que se debe encontrar la casilla de verificación al iniciar el programa

El bloque siguiente tiene dos funciones. En el menú desplegable (triángulo pequeño) se puede seleccionar cuál se va a utilizar. Se puede establecer la actividad (seleccione enabled en el menú desplegable) o el estado (seleccione checked en el menú desplegable) como **verdadero** o **falso**.



## Programa de la casilla de verificación

El programa de la casilla de verificación se ejecuta cada vez que se pulsa dicha casilla de verificación. Se escribe de manera independiente al programa principal. Las variables funcionan en ambos programas. El programa de la casilla de verificación se ejecuta en el bloque **cuando se activa la casilla de verificación**. El bloque de **Eventos []** se establece en «checked» en el programa del interruptor, y muestra **verdadero** o **falso**. Este programa de ejemplo enciende el LED cuando se marca la casilla de verificación; de lo contrario, el LED se apaga.

en la casilla de verificación - alternada: evento

+ si

evento checked = verdadero

hacer

establecer LED encendido

sino

establecer LED apagado

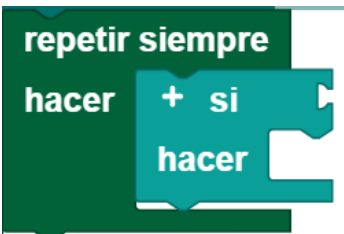
# Motor

## Arranque de cada bloque

El **arranque de cada bloque** ofrece la posibilidad de ejecutar un programa si se cumple una condición. Funciona de manera similar a la distinción de casos, pero no se ejecuta solo una vez, sino cada vez que se cumple la condición durante todo el transcurso del programa. **Arranque de cada bloque:**



Es una abreviatura del siguiente constructo:



Se pueden utilizar todas las condiciones de esta categoría en el **Arranque de cada bloque** de la categoría Motor.

**Nota:** La sección del programa dentro de Arranque de cada bloque debe ser corta y no contener llamadas de bloqueo ni bucles sin fin, de modo que esta parte del programa se pueda procesar rápidamente.

## Motor

El icono en los bloques del motor representa todos los motores que no son codificadores ni servomotores.

## Configuración

Con el bloque **configuración de la velocidad del motor en [] ...** se puede configurar la velocidad de un motor en un valor determinado (desde 0 hasta 512). En el menú desplegable (triángulo pequeño) se puede seleccionar el sentido de rotación.

## Acceso

Con el bloque **obtención de la velocidad del motor** se puede acceder a la velocidad del motor y procesarla como valor.

## Consulta

Con los bloques **el motor está funcionando** y **la velocidad del motor es ...** se puede configurar la actividad o la velocidad de un motor como un estado.

## Detención

Con el bloque **detención del motor** [] es posible parar un motor. Para ello, el bloque **detención del motor** [] ofrece opciones para detener un motor directamente o de esperar a que se agote. La opción deseada se puede seleccionar utilizando el menú desplegable (triángulo pequeño):



## Servomotor



## Configuración

Con el bloque **configuración de la posición en ...** se puede configurar la posición de un servomotor en un valor determinado (desde 0 hasta 512). 0 y 512 son los valores de la desviación máxima hacia la derecha y hacia la izquierda. Por consiguiente, con el valor 256 el servomotor se encuentra en la parte central.

## Acceso

Con el bloque **acceder a la posición** se puede acceder a la posición de un servomotor y procesarla como valor.

## Motor codificador



El motor codificador tiene las mismas funciones que un motor normal, pero también ofrece la opción de contar las revoluciones y de controlar diferentes motores de manera sincronizada. Una revolución se divide en ~ 64 pasos.

## Configuración

Con el bloque

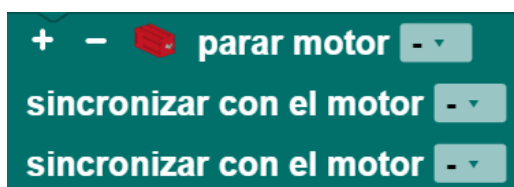


se puede configurar la velocidad de un motor en un valor determinado (de 0 a 512). En el menú desplegable (triángulo pequeño) se puede seleccionar el sentido de rotación. Además, puede introducir el número de pasos que debe recorrer el motor. En este ejemplo, el motor gira 100 pasos, es decir, una revolución y un tercio. Como se puede ver en el ejemplo, este bloque tiene un signo +, y a través de él se pueden controlar diferentes motores de forma sincronizada. Es posible sincronizar motores en el maestro o en una extensión pero no se puede realizar una sincronización general entre los motores del maestro y una extensión.

**Nota:** las llamadas de sincronización que se suceden rápidamente, como las que permite un bucle (véase el ejemplo), pueden perjudicar la sincronización o incluso impedirla por completo.

## Detención

El motor se puede parar utilizando el bloque **detención del motor** .... Si se desea detener más de un motor al mismo tiempo, se pueden añadir hasta tres motores adicionales mediante el signo + que se encuentra a la izquierda del bloque.



## Consulta

El bloque **se ha obtenido la posición** se utiliza para usar la posición alcanzada como condición. En este caso, posición hace referencia a la posición final de un motor codificador después de haber completado la amplitud de paso.