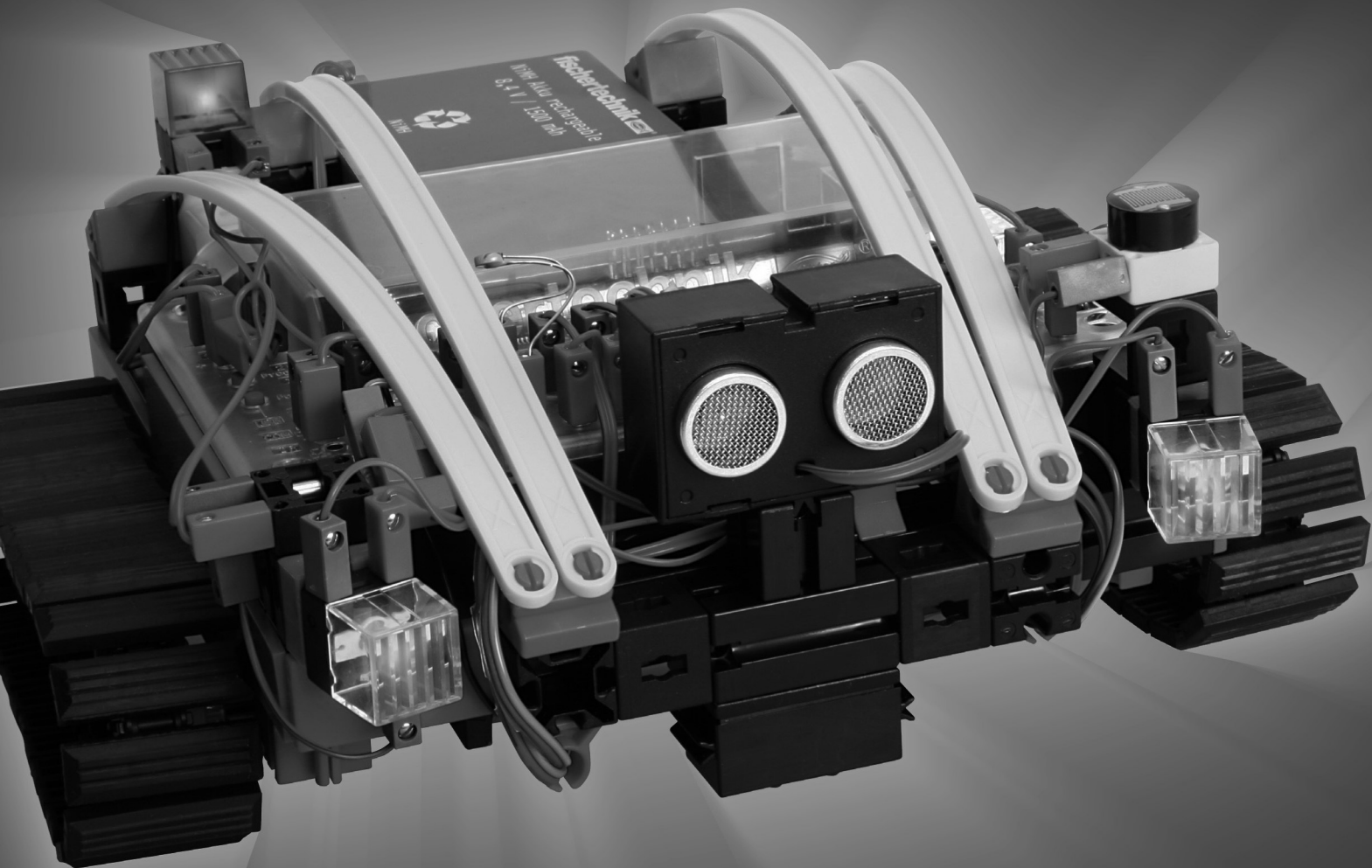


fischertechnik 

COMPUTING

Begleitheft
Activity booklet
Manuel d'accompagnement
Begeleidend boekje
Cuaderno adjunto
Folheto



ROBO EXPLORER
6 MODELS

(D) Seite 1–16

ROBO EXPLORER

Das Begleitheft zum Baukasten

Beschreibt genau, wie du vorgehen musst.

Enthält zahlreiche Programmieraufgaben.

(GB+USA) Page 17–32

ROBO EXPLORER

**The Activity Booklet for the
Konstruktion Kit**

Describes exactly what you must do.

Contains numerous programming tasks.

(F) Page 33–48

ROBO EXPLORER

**Le manuel d'accompagnement
du jeu de construction**

Vous explique en détail comment procéder.

Propose un grand nombre de commandes
à programmer.

(NL) Blz. 49–64

ROBO EXPLORER

**In het begeleidend boekje van de
bouwdoos**

Staat precies hoe je te werk moet gaan.

Bevat verschillende
programmeeropgaven.

(E) Página 65–80

ROBO EXPLORER

**El cuaderno adjunto para el kit de
construcción**

Describe exactamente cómo tienes que
proceder. Contiene numerosas tareas de
programación.

(P) Página 81–96

ROBO EXPLORER

O auxiliar do kit

Descreve detalhadamente como você
deve proceder. Contém numerosas
tarefas de programação.

Inhalt



Fahrzeuge mit Raupenantrieb	S.2
Die Lenkung	S.2
Explorer-Modell aus fischertechnik	S.3
Aktoren	S.3
Sensoren	S.3
ROBO Interface	S.5
Stromversorgung	S.5
Software ROBO Pro	S.5
Interface-Erweiterungen	S.5
Vorüberlegungen	S.6
Grundlagen zum Interface	S.6
Das Basismodell	S.7
Basisprogramm	S.7
Impulssteuerung	S.8
Unterprogramme	S.8
Autonome Raupenfahrzeuge	S.9
Spursucher	S.9
Tunnelroboter	S.12
Farberkenner	S.13
Explorer	S.14
RoboCupJunior – Rescue Robot	S.16
Wichtige Tipps	S.16

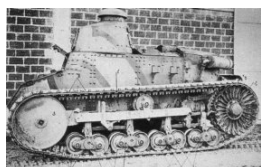
Fahrzeuge mit Raupenantrieb

■ Wozu benötigt man autonome Fahrzeuge mit Raupenantrieb? Die Erfindung des Raupenantriebs für Fahrzeuge wurde notwendig, um auch unwegsames Gelände zu überwinden. Wo ein Reifenantrieb nicht mehr funktionierte, z. B. in der Wüste kam der Raupenantrieb zum Einsatz. So wurden im 1. Weltkrieg die ersten raupengetriebenen Lastkraftwagen und Panzer gebaut und eingesetzt.



Je nach Gelände konnte man die Fahrzeuge von Reifen- auf Raupenantrieb umrüsten.

Auch in der zivilen Nutzung waren Raupenfahrzeuge im Einsatz. Wie du schon anhand der Bilder erkennen kannst, waren eigentlich immer radangetriebene Fahrzeuge Grundlage für Raupenfahrzeuge.



Aber schon bald stellte man einen Schwachpunkt fest: Die lenkbaren Vorderräder. Deshalb ging man dazu über, den Raupenantrieb auf alle Achsen zu erweitern.

Die Lenkung



■ Aber wie funktionierte dann die Lenkung? Ganz einfach, durch Verlangsamen oder Beschleunigen einer der beiden Raupen. Wollte man eine Rechtskurve fahren, verlangsamte man durch einen Steuerknüppel (pro Raupe einer) die rechte Raupe. Dadurch drehte sich diese langsamer und somit wurde das Fahrzeug nach rechts bewegt.

Auch heute, natürlich auf dem neuesten Stand der Technik, findest du viele raupengetriebene Fahrzeuge. Vom Kleinbagger bis zu Riesenkolossen im Übertagebau für Braunkohle.



In der Cheops-Pyramide in Ägypten wollte man mit Hilfe eines Miniroboters weiteren Geheimnissen auf die Spur kommen.

Die Forscher schickten den Roboter von der Größe einer Spielzeuglokomotive durch einen dunklen, schmalen Schacht. Er führt hinauf von einer Kammer im Zentrum der 4500 Jahre alten Pyramide und endet vor einer geheimnisvollen Steintür.



■ Unbekannte Räume erkunden, Abstände messen, Spuren folgen, Fahrtrichtungen durch Blinksignale anzeigen, Farben erkennen, Temperaturen messen, berührungslos Hindernissen ausweichen, Tag und Nacht erkennen, Scheinwerfer automatisch ein- und ausschalten, Alarm auslösen, etc. Das alles – und noch viel mehr – ermöglichen die Sensoren des ROBO Explorer. Im Einzelnen ist das: der NTC-Widerstand, der Fotowiderstand, der Ultraschall-Abstandssensor, der optische Farbsensor sowie der speziell entwickelte Infrarot-Spursensor. Dank zweier Power-Motoren und des Raupenantriebs kann auch unwegsames Gelände erforscht und befahren werden. Mit dem enthaltenen Modell Rescue Robot ist der Baukasten eine ideale Grundlage zur Teilnahme am RoboCup-Junior.

Bevor du nun loslegst, solltest du dich etwas mit den wichtigsten Bauteilen beschäftigen. Diese werden nachfolgend beschrieben:

Powermotor

Um die Raupen und somit deine Modelle anzutreiben und zu lenken, benötigst du 2 Powermotoren. Diese werden am Interface an den Ausgängen M1 bis M4 angeschlossen.

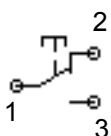
Kugellampe

Hier handelt es sich um eine Glühlampe für eine Spannung von 9 V. Diese kann als Meldesignal für die Fahrtrichtung oder einfach als Beleuchtung eingesetzt werden. Sie wird wie der Powermotor an den Interfaceausgängen M1 bis M4 angeschlossen.

Summer

Der Summer dient z. B. dazu, erkannte Hindernisse oder Farben akustisch zu melden. Angeschlossen wird er ebenfalls an den Interfaceausgängen M1 bis M4.

Taster



2 Taster zählen zu den Berührungssensoren. Betätigst du den roten Knopf, wird im Gehäuse ein Kontakt mechanisch umgelegt und es fließt ein Strom zwischen den Kontakten 1 und 3. Gleichzeitig unterbricht die Schalterstrecke zwischen den Anschlusspunkten 1 und 2.

Taster oder Schalter werden auf zwei verschiedene Arten verwendet:

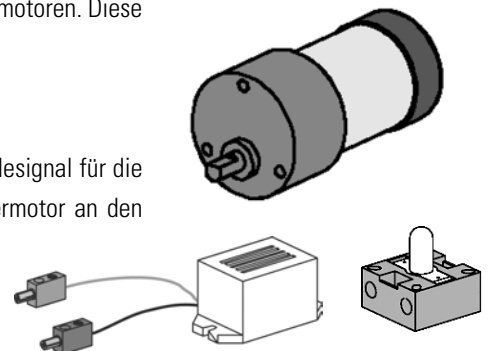
Taster als „Schließer“

Die beiden Schaltbilder zeigen dir den Versuchsaufbau. Eine Spannungsquelle (9V) wird an den Kontakt 1 des Tasters, eine Lampe an den Kontakt 3 des Tasters und an den Minuspol der Spannungsquelle angeschlossen.

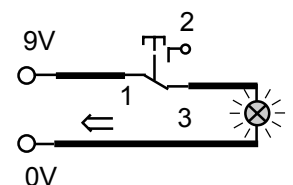
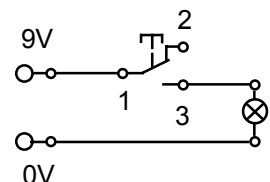
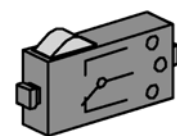
Bei nicht betätigtem Taster ist die Lampe aus. Drückt man den Taster wird der Stromkreis über Kontakt 1 und Kontakt 3 geschlossen, die Lampe leuchtet.

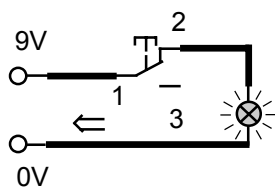
Explorer-Modell aus fischertechnik

Aktoren



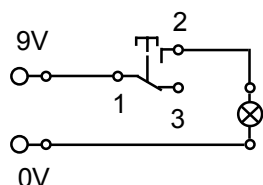
Sensoren





Taster als „Öffner“

Auch hier zeigen dir die beiden Schaltbilder die Funktionsweise. Kontakt 1 wird mit der Spannungsquelle verbunden. Kontakt 2 mit der Lampe und die Lampe wieder mit dem Minuspol der Spannungsquelle.



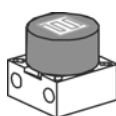
Die Lampe leuchtet. Betätigst du jetzt den Schalter, wird der Strom unterbrochen, die Lampe verlischt.

In deinen Modellen setzt du die Taster zusammen mit dem Impulsrad als Zähltaster ein.



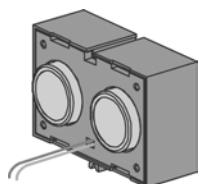
NTC-Widerstand (Negativ Temperatur Coeffizient)

Hier handelt es sich um ein Bauteil, mit dem du verschiedene Temperaturen messen kannst. Man spricht auch von einem Wärmesensor. Bei ca. 20 Grad hat der NTC-Widerstand einen Wert von 1,5 kOhm. Steigt die Temperatur, nimmt der Widerstandswert ab. Diese Information kann in das Interface an den analogen Eingängen AX und AY eingelesen werden und steht in ROBO Pro als Zahlenwert von 0 - 1023 zur Verfügung.



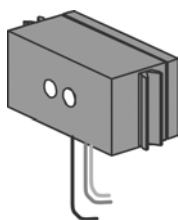
Fotowiderstand

Der LDR 03, ein analoger Helligkeitssensor für die Eingänge AX und AY (Widerstandsmessung), reagiert auf Tageslicht und verändert dabei seinen Widerstandswert. Dieser ist ein Indikator für die Lichthelligkeit.



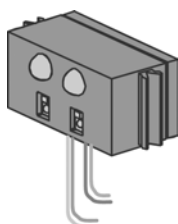
Ultraschall-Abstandssensor

Ein Abstandssensor ist ein technisches Bauteil, das in der Lage ist, den Abstand zwischen sich und einem Objekt zu messen. Abstandssensoren arbeiten mit Licht, Infrarotstrahlung, Radiowellen, oder **Ultraschall** und verwenden unterschiedliche Messmethoden. Schall breitet sich als Welle aus. Ein Echo wird zur Ultraschallquelle reflektiert, welches nun als Signal wieder gefangen und ausgewertet wird. Die Zeitdifferenz zwischen Aussenden und Empfangen des Signals gibt Aufschluss über die Distanz zwischen Hindernis und Sensor. Als Anschlüsse sind D1/D2 am Interface vorgesehen. Die Reichweite des Sensors beträgt bis zu 4 m. Der ausgegebene Zahlenwert entspricht dem Abstand in Zentimeter.



Optischer Farbsensor

Farbsensoren werden meist in der Automatisierungstechnik eingesetzt. Dabei soll z. B. die Farbe oder ein Farbaufdruck kontrolliert werden, um sicher zu gehen, dass die richtigen Bauteile eingebaut werden. Der fischertechnik Farbsensor sendet rotes Licht aus, das von verschiedenen Farbflächen unterschiedlich stark reflektiert wird. Die Menge des reflektierten Lichts wird über den Fototransistor gemessen und als Spannungswert zwischen 0 V und 10 V ausgegeben. Der Messwert ist abhängig von der Umgebungshelligkeit sowie vom Abstand des Sensors zur Farbfläche. Diesen Wert kannst du über die Eingänge A1 und A2 einlesen und als Zahlenwert von 0 - 1000 in deinem Programm verarbeiten.



Spursensor

Der IR-Spursensor ist ein digitaler Infrarotsensor zur Erkennung einer schwarzen Spur auf weißem Untergrund im Abstand von 5 - 30 mm. Er besteht aus zwei Sende- und zwei Empfängerelementen. Als Anschluss benötigst du zwei digitale Eingänge und die 9 V-Spannungsversorgung (plus und minus).

■ Das wichtigste Bauteil, um ein Raupenfahrzeug aufzubauen, ist das ROBO Interface, welches fest in die verschiedenen Modelle eingebaut wird. An ihm schließt du je nach Bedarf deine Sensoren und Aktoren an. Die Grundverdrahtung entnimmst du der beliegenden Bauanleitung.

■ Bei den ROBO Explorer-Modellen handelt es sich um autonome Fahrzeuge, die sich frei im Raum bewegen. Als Stromversorgung verwendest du deshalb das fischertechnik Accu Set.

■ ROBO Pro ist eine einfache grafische Programmieroberfläche mit der du deine Programme schreiben kannst. Der Vorteil ist, dass du keine Programmiersprache lernen musst. Eigentlich kannst du sofort loslegen.

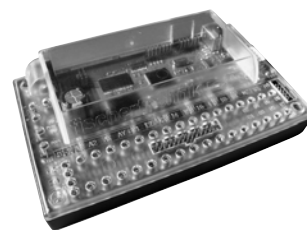
Für den Baukasten ROBO-Explorer benötigst du die ROBO Pro Version 1.2.1.30. Falls du eine ältere Version der Software besitzt, kannst du sie kostenlos updaten. Entweder über das Hilfe-Menü in ROBO Pro – Neue Version downloaden oder unter

www.fischertechnik.de/robopro/update.html

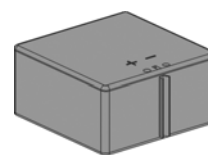
ROBO RF Data Link

Die Funkschnittstelle ROBO RF Data Link ersetzt das Schnittstellenkabel zwischen dem PC und dem Interface. Mit einer Funkübertragung werden die Daten auf das Interface gespielt. Der Vorteil ist, dass das oft lästige Kabel entfällt. Ein weiterer Vorteil ist, dass du Programme über Funk im Onlinemodus betreiben kannst. So lassen sich Fehler leichter finden als im Downloadbetrieb. Und noch einen Vorteil hat diese Methode – mobile Roboter lassen sich online über den Bildschirm steuern und können Messwerte kabellos an den Rechner übermitteln. Sind mehrere Roboter mit einem Data Link ausgestattet, können sie sogar untereinander Daten austauschen, sozusagen miteinander „reden“.

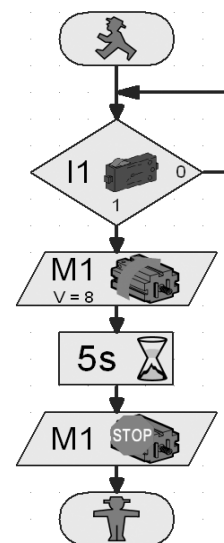
ROBO Interface



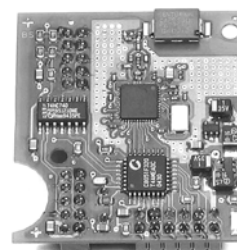
Stromversorgung



Software ROBO Pro



Interface-Erweiterungen



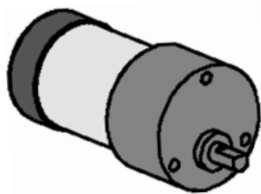
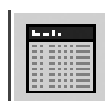
Vorüberlegungen

■ Wie bei allen anderen fischertechnik-Robotern wirst du dich beim ROBO Explorer schrittweise in die Faszination der Technik und Programmierung einarbeiten. Du beginnst mit einem einfachen Modell und arbeitest dich bis zu immer umfangreicheren Systemen mit faszinierenden Möglichkeiten vor. Im Vordergrund steht bei allen Modellen ein sorgfältiger Aufbau und eine sorgfältige Inbetriebnahme.

Grundlagen zum Interface

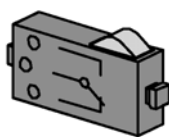
■ Bevor du dich an die einzelnen Modelle heranwagst, solltest du dich mit Hilfe von einigen Experimenten mit dem Interface vertraut machen. Lege dazu das Handbuch der Software ROBO Pro bereit um bei Problemen nachschlagen zu können.

Nachdem du die Software installiert hast, kannst du das Interface über das mitgelieferte Kabel an deinem PC anschließen. Starte nun das Programm ROBO Pro und öffne „Interface testen“.



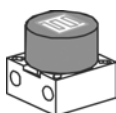
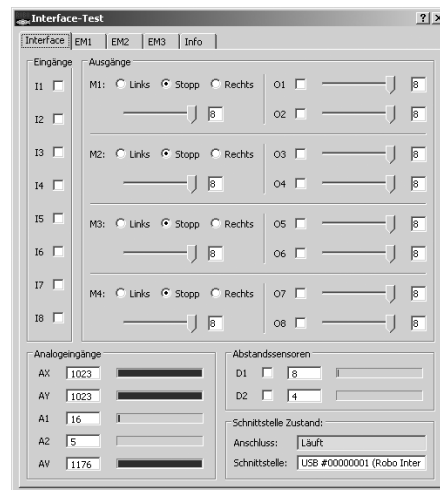
Powermotor

Verbinde die Anschlüsse des Powermotors mit dem Anschluss M1. Klicke mit der Maus auf die Auswahl „Links“ oder „Rechts“. Der Motor läuft mit seiner Maximalgeschwindigkeit an. Durch Betätigen des Reglers kannst du die Umdrehungsgeschwindigkeit einstellen. Über „Stopp“ wird der Vorgang beendet.



Taster

Schließe als nächstes einen Taster (Schließer) am Digital Eingang I1 an und beobachte was an der Testanzeige geschieht, wenn du den Taster betätigst. Eine optische Anzeige (Häkchen) zeigt dir den Zustand des Tasters an. Schließe den Taster als Öffner an und betrachte das Ergebnis. Es erscheint zuerst die Zustandsanzeige für „Schalter geschlossen“. Betätigst du den Schalter, verlischt die Anzeige.



Fotowiderstand

Schließe den mitgelieferten Fotowiderstand an dem Anschluss AX oder AY an.

Verändere die Lichtstärke des Fotowiderstands durch langsames Abdecken mit einem schwarzen Papierstreifen. Was passiert? Du wirst sehen, es verändert sich der blaue Leuchtbalken und der Zahlenwert des benutzen Eingangs.

Der Test des Interfaces wird im Handbuch im Kapitel 2 sehr gut erläutert. Ebenfalls findest du Hilfe, falls es Probleme zwischen deinem Rechner und dem Interface und der Software gibt – reinschauen lohnt sich!

■ Jetzt geht es los. Nachdem du mit dem Interface und der Programmierung vertraut bist, kannst du nun die erste Aufgabe lösen. Zuerst wird das Basismodell anhand der Bauanleitung aufgebaut.



Aufgabe 1 - ROBO Pro Level 1:

Dein Raupenfahrzeug soll 6 Sekunden lang geradeaus fahren, anschließend soll es sich 3 Sekunden lang nach links drehen und dann stehen bleiben.

Bei deinem ersten Programm wollen wir dich noch etwas unterstützen. Klicke als Erstes auf den Button „Datei-Neu“. Dein Programm beginnt mit einem grünen Ampelmännchen für Programmstart.



Anschließend benötigst du 2 Motorsymbole. Setze das erste Symbol so unter den Programmstart, dass die Verbindung automatisch gezogen wird. Bewege die Maus auf das Motorsymbol und schalte das Eigenschaftsfenster ein (rechte Maustaste). Dort stellst du den Motorausgang „M1“ und bei Aktion die Drehrichtung „Links“ ein. Dann bestätigst du mit OK. Füge in gleicher Weise das zweite Motorsymbol ein und wiederhole den Vorgang für den Motorausgang „M2“.

Als nächstes soll das Programm eine bestimmte Zeit warten. Dazu verwendest du das Wartezeit-Symbol. Dieses platzierst du unter den zweiten Motor und stellst die Zeit auf 6 Sekunden ein.

Anschließend soll sich das Raupenfahrzeug 3 Sekunden drehen. Dazu fügst du nochmal zwei Motorsymbole für M1 und M2 ein. M1 soll sich nach links und M2 nach rechts drehen. Da beide Motoren 3 Sekunden arbeiten sollen, füge anschließend das Wartezeit-Symbol ein und setze den Zeitwert auf 3 Sekunden.

Danach musst du beide Motoren stoppen. Dies geschieht wie vor der Drehung mit dem Einfügen der beiden Motorsymbole und der Parametereinstellung „Stopp“.

Zum Schluss musst du noch das Symbol für Programmende „Rotes Ampelmännchen“ einfügen. Jetzt ist dein erstes Programm fertig und du kannst es speichern. Teste es dann im Onlinemodus. Klicke dazu auf den Button „Start“.

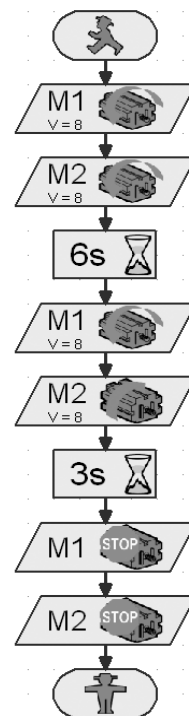
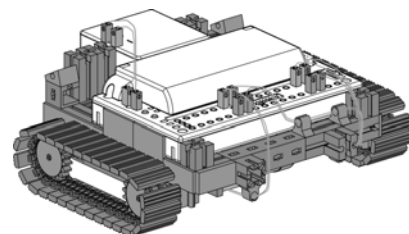
Hast du alles richtig gemacht, kannst du das Programm auf das Interface laden. Dazu klickst du auf den Button „Programm zum Interface herunterladen“. Übernehme die Einstellung des Download-Fensters. Gleich nach dem Download fährt das Modell los. Leider hängt es aber noch am USB-Kabel. Lade das Programm nochmals, aktiviere aber „Programm über Taster am Interface starten“. Ist das Programm übertragen, kannst du das Kabel abziehen. Drücke zum Starten des Programms am Interface die „Prog-Taste“.

Das fertige Programm findest du unter:

C:\Programme\Robopro\Beispielprogramme\Robo_Explorer\Basismodell_1.rpp

Das war eigentlich zu einfach für dich. Aber wie schon erwähnt, sollst du dich ja langsam vorarbeiten. Dazu wird die Aufgabe etwas erweitert.

Das Basismodell

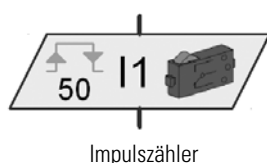
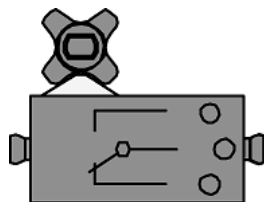


Programm im Online-Modus starten



Programm zum ROBO-Interface herunterladen

Impulssteuerung



■ Das Basismodell enthält bereits 2 Sensoren (Taster) zur Wegmessung sowie jeweils ein Impulsrad, das mit der Motorachse verbunden ist. Dieses Impulsrad betätigt den Schalter bei jeder Umdrehung vier Mal. Prüfe das Modell mit „Interface testen“ auf Funktionsfähigkeit.

Aufgabe 2 - ROBO Pro Level 1:

Dein Raupenfahrzeug soll 50 Impulse vorwärts fahren. Anschließend soll es 11 Impulse nach links fahren und dann stoppen. Informationen zum Impulszähler findest du im Handbuch im Kapitel 3.6.3.



Lösung

Schalte zuerst beide Motoren ein. Füge anschließend den Impulszähler hinzu. Ändere in den Eigenschaften die Anzahl der Impulse auf 50 um. Füge nochmals 2 Motorensymbole ein. M1 dreht weiter nach „Links“, M2 schaltest du auf „Rechts“. Dieses Drehen soll 11 Impulse lang dauern. Zum Schluss musst du nochmals 2 Motorensymbole einfügen und mit „Stopp“ belegen. Übertrage das Programm und teste es an deinem Fahrzeug.

Wenn du im Eigenschaftsfenster des Impulszählers den Impulstyp „0->1“ oder „1->0“ einstellst, erhältst du acht anstelle von vier Impulsen pro Umdrehung des Impulsrades. D. h. die Anzahl der Impulse pro zurückgelegter Wegstrecke verdoppelt sich und die Genauigkeit der Wegmessung steigt.

Das fertige Programm findest du unter:

C:\Programme\Robopro\Beispielprogramme\Robo_Explorer\Basismodell_2.rpp

Unterprogramme



Neues Unterprogramm erzeugen

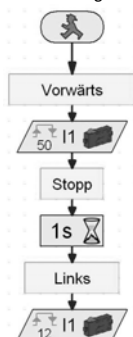


Aktuelles Unterprogramm kopieren



Aktuelles Unterprogramm löschen

Lösung



Sicher hast du beim Erstellen des Programms festgestellt, dass man für jeden Richtungswechsel oder Stopp jedesmal die Motoren setzen muss. Große Programme werden dann sehr schnell unübersichtlich und die Fehlersuche wird oft zu einem Geduldsspiel.

Hier bietet dir ROBO Pro als elegante Lösung das Arbeiten mit „Unterprogrammen“ an. Lese dazu im Handbuch das Kapitel 4.1 durch. Wichtig ist, dass du in ROBO Pro auf **Level 2** umschaltest.



Aufgabe 3 - ROBO Pro Level 2:

Dein Raupenfahrzeug soll ein Quadrat abfahren. Verwende wie in der 1. Aufgabe die Parameter 50 und 11. Erstelle für jede Fahrtrichtung und für den Stopp-Befehl ein Unterprogramm.

Erstelle zuerst das Unterprogramm „Vorwärts“ (siehe ROBO Pro Handbuch Kapitel 4). Markiere die Programmteile und kopiere sie in den Zwischenspeicher. Anschließend erstellst du das Unterprogramm „Links“ und „Stopp“. In beide fügst du aus dem Zwischenspeicher die Programmteile für „Vorwärts“ ein und änderst die Parameter entsprechend. Für das Drehen verwendest du eine geringere Umdrehungsgeschwindigkeit. Als kleine Hilfe haben wir dir einen Teilausschnitt aus der Aufgabe aufgezeigt.

Die nachfolgende Tabelle soll dir auf einen Blick zeigen, wie du die Motoren für die Fahrtrichtungen programmieren musst.

Fahrtrichtung	Drehrichtung Motor 1	Drehrichtung Motor 2
Vorwärts	Links	Links
Rückwärts	Rechts	Rechts
Links	Links	Rechts
Rechts	Rechts	Links
Stopp	Stopp	Stopp

Anhand dieser Tabelle sind alle Motoren in den Beispielprogrammen programmiert.

Fertiges Programm:

C:\Programme\Robopro\Beispielprogramme\Robo_Explorer\Basismodell_3.rpp

Aufgabe 4 - ROBO Pro Level 3:

Der Roboter fährt kein genaues Quadrat ab. Prüfe folgende Fragen: Was ist die Ursache? Wie kann das Ergebnis verbessert werden?

Tipp:

Synchronisiere die Motoren M1 und M2 über die Taster I1 und I2 so, dass das Modell exakt geradeaus fährt. Dazu haben wir dir ein fertiges Unterprogramm „gerade_sync“ mitgeliefert. Hier muss nur noch die Anzahl der Impulse als Konstante am orangen Unterprogrammeingang eingegeben werden (siehe Basismodell_4.rpp). Genauso kannst du im Unterprogramm „Drehen_sync“ die Anzahl der Impulse für den Drehwinkel eingeben.



Fertiges Programm:

C:\Programme\Robopro\Beispielprogramme\Robo_Explorer\Basismodell_4.rpp

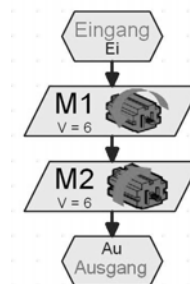
■ Nachdem du mit dem Basismodell genügend experimentiert hast, soll nun dein Roboter auf verschiedene Signale von außen reagieren.

Damit dein Raupenfahrzeug seine Umgebung erkennt und bestimmte Aufgaben erfüllt, musst du es mit Sensoren ausstatten. Die folgenden Modellvorschläge zeigen dir verschiedene Varianten von Raupenfahrzeugen mit unterschiedlichen Sensoren. So sollen verschiedene Wegstrecken, Licht oder Farben aber auch Wärmequellen oder Abstände erkannt werden. Die einzelnen Programme findest du im Verzeichnis: C:\Programme\Robopro\Beispielprogramme\Robo_Explorer\

Sicher sind dir aus dem Fernsehen Filme von menschenleeren Fabrikhallen, in denen wie von Geisterhand gesteuerte Transportfahrzeuge fahren, bekannt. Zum Teil werden solche Systeme mit im Boden eingelassenen Datenleitungen oder auf dem Boden aufgezeichneten Wegemarkierungen gesteuert.

Grundlage deiner Programmierung soll sein, dass der Roboter an einer schwarzen Linie entlang fährt.

Bevor du mit dem Programmieren beginnst, baue zuerst anhand der Bauanleitung den Spurensucher auf. Einen Experimentier-Parcours mit aufgedruckter schwarzer Linie findest du im Baukasten. Die Linie, an der der Spurensucher entlangfahren soll, sollte zuerst einmal eine Gerade sein.



Unterprogramm „Links“

Autonome Raupenfahrzeuge

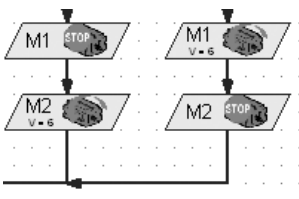
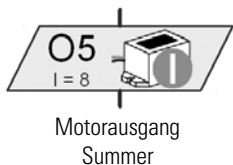
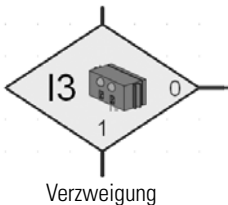
Spursucher





■ Wie soll nun das Modell funktionieren?

Der Roboter soll auf einem weißen Untergrund eine schwarze Linie finden und dieser dann folgen. Um dies zu ermöglichen, hast du in deinem Modell den IR-Spursensor eingebaut. Der Baustein sendet ein Licht im Infrarot-Bereich auf den Fahrbahnnuntergrund. Dieses wird, je nach Untergrund reflektiert und von Fototransistoren gemessen. Für deine Programmierung bedeutet das: Heller/weißer Untergrund reflektiert das Licht und du erhältst den Wert 1. Bei einem schwarzen Untergrund wird das Licht nicht reflektiert und du bekommst den Wert 0. Haben beide Transistoren den Wert 0 hat dein Roboter den Fahrweg (schwarze Linie) gefunden und muss nun diesem folgen.



Aufgabe 1 - ROBO Pro Level 2:

Dein Raupenfahrzeug soll auf eine gerade schwarze Spur gestellt werden und daran entlangfahren. Verliert es die Spur oder ist diese zu Ende soll es stehen bleiben und ein akustisches Signal eine Sekunde lang abgeben.

Gliedere das Gesamtprogramm in ein Hauptprogramm für die Sensorabfrage und jeweils ein Unterprogramm für die Fahrt geradeaus, das Summersignal und für Stopp.

Ein paar kleine Tipps:

Prüfe mit dem Interfacetest die Spurerkennung des Sensors. Sollte die schwarz-weiß-Erkennung nicht richtig funktionieren, können störende Lichtquellen (z. B. Sonne) die Ursache sein. Gegebenenfalls muss der Sensor etwas näher an die Spur positioniert oder mit einer Bauplatte abgeschirmt werden.

Fertiges Programm: **Spursucher_1.rpp**

■ Sicherlich wirst du mit deiner ersten Lösung nicht zufrieden sein – denn dein Roboter fährt zwar eine kleine Strecke entlang der Linie. Da er aber noch nicht nachjustieren kann, verlässt er die Markierung, bleibt stehen und signalisiert dir dies.

Aufgabe 2 - ROBO Pro Level 2:

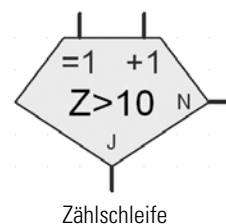
Erweitere dein Hauptprogramm in den Abfragezweigen der Fototransistoren so, dass der Roboter erkennt, wenn er nicht mehr exakt auf der Spur fährt. Dann soll er seine Fahrtrichtung entsprechend korrigieren. Einen Tipp findest du im linken Programmausschnitt.

Jetzt sieht es schon besser aus. Dein Roboter bleibt genau auf der vorgegebenen Spur. In einer Industriehalle würden jetzt andere Roboter am Ende der Spur die transportierte Ladung entnehmen oder den Roboter mit einer neuen Ladung versehen. Diese könnte dann zum Ausgangspunkt der Fahrt zurückbefördert werden.

Fertiges Programm: **Spursucher_2.rpp**

Aufgabe 3 - ROBO Pro Level 2:

Dein Roboter ist in den vorherigen Aufgaben an einer bestimmten schwarzen Linie entlanggefahren. In dieser Aufgabe soll er eine Linie suchen. Dazu soll er sich einmal im Kreis drehen. Findet er dabei keine Spur, soll er eine kleine Wegstrecke geradeaus fahren und dann erneut suchen. Hat dein Roboter eine Spur gefunden, soll er der Spur folgen. Endet diese oder verliert er sie, soll er mit dem Suchen erneut beginnen.

**Tipp:**

Erinnere dich an die ersten Aufgaben im Basismodell. Hier sollte der Roboter sich um 90 Grad drehen. Dies hast du mit einer Wegesteuerung bewerkstelligt. Auch hier kann dir diese Technik von Nutzen sein.

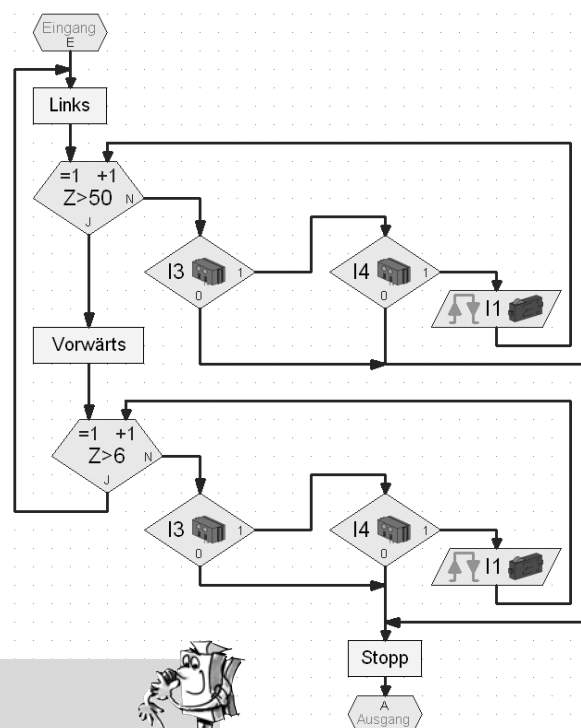
Für die Spurensuche legst du ein eigenes Unterprogramm mit dem Namen „Spursuche“ an. Im Bild siehst du, wie wir es vorschlagen würden.

Schreibe noch ein Unterprogramm mit dem Namen Links. Vorwärts und Stopp hast du ja schon angelegt.

Somit sind das Hauptprogramm und alle Unterprogramme für den Spursucher fertig. Deine Aufgabe ist es nun, diese so in das Programm einzubinden, dass die Aufgabe 3 erfüllt werden kann.

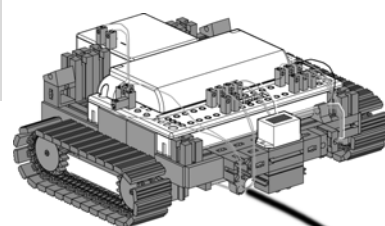
Fertiges Programm: **Spursucher_3.rpp**

Die bisherigen Fahrstrecken waren immer eine Gerade. Es kommt aber auch vor, dass Fahrstrecken mit Kurven ausgestattet sind. Dies findet man z. B. in Industrieanlagen in denen Material oder Werkstücke von einer Maschine zur nächsten transportiert werden müssen.

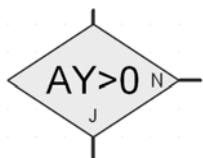
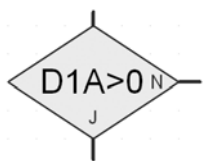
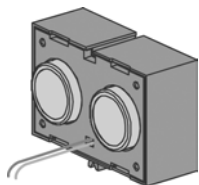
**Aufgabe 4 - ROBO Pro Level 2:**

Der Experimentier-Parcours enthält verschiedene Kurven mit unterschiedlichen Radien.

Experimentiere bei deinem Rundkurs auch mit unterschiedlichen Geschwindigkeiten von M1 und M2. Mit welchen Einstellungen schafft der Roboter den Parcours am schnellsten? Halte in einer kleinen Tabelle die Ergebnisse fest.



Tunnelroboter, Abstands- und Temperatursensor



- Mit dem Sensor zur Abstandsmessung und dem NTC-Widerstand bieten sich dir weitere Möglichkeiten, dein Modell zu einem Profi-Roboter auszubauen.

Abstände, Temperaturen messen und evtl. Hilfsmaßnahmen einleiten. Was meinst du, wo könnten Roboter mit solchen Fähigkeiten eingesetzt werden? Sicher fallen dir hierzu verschiedene Einsatzgebiete ein. Wir wollen uns mit dem Bereich des Feuerschutzes und der Feuerbekämpfung in Auto- und Eisenbahntunnels befassen.



Die Aufgabe eines solchen Roboters ist es, sich an den Brandherd heranzutasten, Temperaturen im Tunnel zu messen und einer Leitzentrale Daten zu melden. Meist sind die Roboter mit einer mobilen Löscheinrichtung ausgestattet, die je nach Bedingung eingesetzt werden kann.

Baue auch hier gewissenhaft nach der Bauanleitung das Modell „Tunnelroboter“ auf. Der Abstandssensor wird an den Anschlüssen D1 angeschlossen.

Aufgabe 1: ROBO Pro Level 2

Ähnlich dem Spurensucher, der an einer Linie entlang fährt, soll dein Roboter an einer Wand, in einem bestimmten Abstand (ca. 20 cm) eine gewisse Strecke fahren.



Fertiges Programm: **Tunnel_1.rpp**

- Lass uns für die nächste Aufgabe noch einmal auf einen Feuerlöschroboter eingehen. Damit er wie dein Roboter an der Wand entlang fährt, benutzt er Abstandssensoren. Um aber den Brandherd zu erkennen, setzt er Wärmesensoren ein. Dieser Wärmesensor ist für dein Modell der NTC-Widerstand. Die physikalische Eigenschaft dieses Bauteils ist, dass sich bei steigender Temperatur der Widerstandswert verringert. Diese Änderung kannst du wieder mit dem Interface-Test ausprobieren. Schließe den NTC-Widerstand an dem Anschluss AY an. Halte eine Wärmequelle an den NTC und beobachte den blauen Balken von AY.

Aufgabe 2: ROBO Pro Level 2

Erweitere das Programm, bei welchem der Roboter an der Tunnelwand entlang fährt. Messe zusätzlich über den AY-Anschluss die aktuelle Wärme. Steigt diese auf einen bestimmten Wärmewert, soll dein Roboter anhalten und ein Warnsignal über den Summer ausgeben. Gleichzeitig mit dem Summer soll die rote Warnleuchte ein Blinksignal abgeben.

Nach diesem simulierten Löschvorgang soll dein Roboter wenden und zum Ausgangspunkt zurückkehren.



Fertiges Programm: **Tunnel_2.rpp**

Tipp:

Da dein Roboter nur einen Abstandssensor besitzt, benötigt er zur Rückkehr zum Ausgangspunkt eine zweite Wand an der er entlang fahren kann.

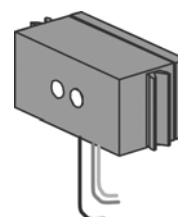
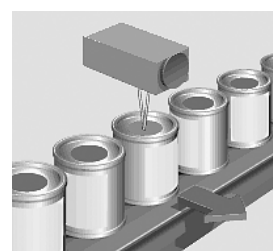
Solltest du in deiner fischertechniksammlung noch einen Motor und eine Luftschraube haben, kannst du auch noch das Löschen des Feuers mit in dein Programm einbeziehen.

■ Als weiteren Sensor wirst du den Farbsensor kennenlernen. Das nebenstehende Bild soll dir ein Beispiel aus dem industriellen Einsatz geben. Wie du erkennen kannst, werden Farbdosen, die falsch eingereiht wurden, aussortiert.

Das vom Tastgut reflektierte Licht wird empfangen, digitalisiert und mittels eines Computers und einer Software aufbereitet. Aufgabe des Sensors ist es, die verschiedenen Farben zu erkennen und Messdaten an das Interface zu senden.

In dem Modell Farberkenner ist der Farbsensor eingebaut. Angeschlossen wird er mit dem schwarzen Draht an A1, dem roten Draht an + und dem grünen Draht an -. Für die ersten Testprogramme verwendest du die auf dem Parcours aufgedruckten Farbflächen.

Farberkenner



Aufgabe 1 - ROBO Pro Level 2:

Überprüfe zuerst die Werte, die das Interface für die verschiedenen Farben im Interface-Test ausgibt. Verwende neben den 3 vorgegebenen Farben noch Schwarz und Weiß.

Erstelle eine kleine Tabelle und trage die Werte, die du gemessen hast ein. Beobachte auch Veränderungen, wenn sich der Abstand zur Farbfläche oder das Umgebungslicht ändert.



Aufgabe 2 - ROBO Pro Level 2:

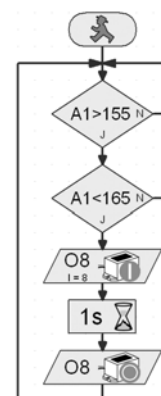
Schreibe ein kleines Programm, mit dem der Sensor die grüne Farbfläche erkennt. Liegt der gemessene Wert im vorgegebenen Wertebereich, wird der Summer eine Sekunde aktiviert. Anschließend springt das Programm zum Start.

Farbe	Wert
Weiß	
Schwarz	
Blau	
Rot	
Grün	

Fertiges Programm: **Farberkenner_2.rpp**

Tipp:

Für die nächste Aufgabe benötigst du die drei Lampen mit verschiedenfarbigen Leuchtkappen, die in dem Modell bereits eingebaut sind.





Aufgabe 3 - ROBO Pro Level 3:

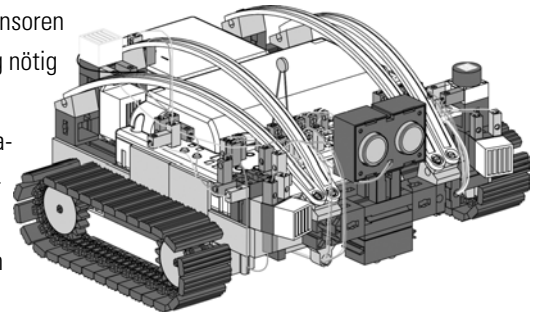
Schreibe ein Programm, welches deinen Roboter eine bestimmte Strecke geradeaus fahren lässt. Auf dieser Strecke befinden sich drei Farbflächen. Erfasst der Sensor eine Farbe, soll der Roboter für 3 Sekunden stehen bleiben. In dieser Zeit schaltet er die Lampen mit der entsprechenden Farbe ein und gibt über den Summer ein Akustiksignal aus. Danach fährt er zur nächsten Fläche und wiederholt seine Arbeit. Anschließend fährt er zur letzten Fläche, meldet das Ergebnis und bleibt dort stehen.

Fertiges Programm: **Farberkenner_3.rpp**

Gesamtmodell Explorer

■ Im Modell „Explorer“ sind alle Aktoren und Sensoren enthalten, die für ein autonomes Roboterfahrzeug nötig sind.

Jetzt sind dir keine Grenzen mehr gesetzt um einfache wie schwierige Aufgaben zu lösen. In den vorangegangenen Baustufen hast du meist nur einen Sensor benutzt um die Einsatzmöglichkeit kennen zu lernen.



Aufgabe 1 - ROBO Pro Level 2:

Programmiere deinen Roboter so, dass er auf seiner Fahrt auf ein vorhandenes Hindernis zufährt. In einem Abstand von ca. 60 cm soll er seine Geschwindigkeit um die Hälfte reduzieren. Bei einem Abstand von 40 cm hält er an. Bewegt sich das Hindernis weiter auf deinen Roboter zu, soll dieser ab einem Abstand von 20 cm langsam und ab 10 cm schnell rückwärts fahren.

Fertiges Programm: **Explorer_1.rpp**

Aufgabe 2 - ROBO Pro Level 2:

Jetzt geht dein Roboter auf seine Erkundungsreise. Erstelle ein Programm für den Einsatz von 2 Sensoren - Spursensor und Abstandsensoren. Zuerst soll der Roboter auf dem Experimentier-Parcours der schwarzen Linie folgen. Auf der Strecke platzierst du ein Hindernis. Er soll ca. 10 cm vor dem Hindernis anhalten und einen Zentimeter zurückfahren. Danach soll er umdrehen und in der anderen Richtung der Spur folgen.



Fertiges Programm: **Explorer_2.rpp**

Aufgabe 3 - ROBO Pro Level 2:

Das Programm aus Aufgabe 2 soll um drei Sensoren erweitert werden – den Farberkenner, den Temperatursensor und den Fotowiderstand zur Helligkeitsmessung.

Entlang der Spur befinden sich verschiedene Farbflächen. Diese meldet der Roboter durch verschiedene akustisches Zeichen. Wird während der Fahrt die Umgebungstemperatur zu hoch, soll die rote Warnlampe blinken. Sobald es im Raum dunkel wird, schaltet dein Roboter seine 2 Scheinwerfer ein. Wird es wieder hell, schalten die Scheinwerfer wieder aus.



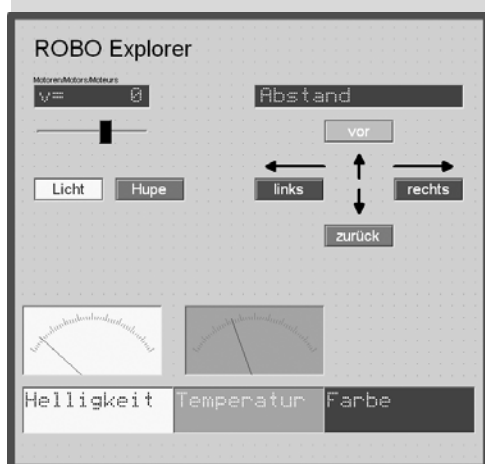
Fertiges Programm: **Explorer_3.rpp**

■ Im Jahre 2003 war der Mars der Erde so nah wie selten. Das nutzten Europäer und Amerikaner, um jeweils eigene Erkundungsfahrzeuge hinaufzuschicken. Deren Auftrag: Leben nachweisen.

Auch unser Explorer lässt sich als ferngesteuerter Roboter programmieren um unbekannte Welten zu entforschen. Dazu wird zusätzlich der RF Data Link benötigt.

**Aufgabe 4 - ROBO Pro Level 3:**

Für diese Aufgabe ist dein Computer die Steuerzentrale für eine Mars-expedition. Auftrag ist, dass Messwerte von einer Marslandschaft zur Bodenstation übermittelt werden. Die Steuerung für deinen Roboter wird in ROBO Pro im Bedienfeld erzeugt (siehe Handbuch Kapitel 8).



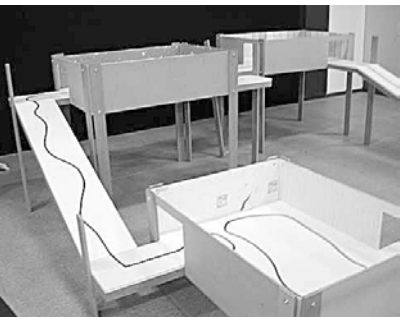
Dein Roboter soll so programmiert werden, dass er Messwerte von Bodenfarbe, Temperatur, Helligkeit und Hindernissen übermittelt. Gesteuert wird der Roboter von Hand über das Bedienfeld des Hauptprogramms in ROBO-Pro.

Fertiges Programm: **Explorer_4.rpp**

Der Mars - Aufbruch zum vierten Planeten



RoboCup Junior Rescue Robot



Jetzt hast du alle Aufgaben erfüllt, dich in den Roboterbau und die Programmieretechnik eingearbeitet – nun könntest du die Früchte deiner Arbeit ernten und mit dem Modell „Rescue Robot“ in der Rescue-Liga des RoboCupJunior antreten.

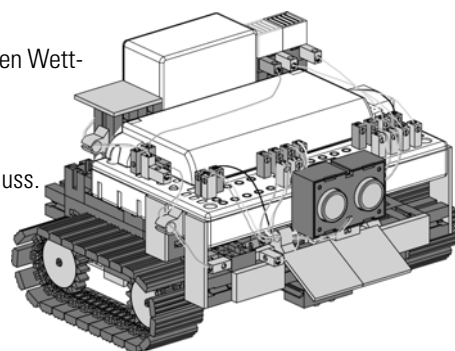
■ RoboCupJunior ist eine weltweite, projektorientierte Bildungsinitiative, die regionale, nationale und internationale Roboter-Veranstaltungen für junge Menschen fördert. Das Ziel besteht darin, Kindern und Jugendlichen Roboter und ihre Anwendung vorzustellen.

Anhand der Aufgabenstellung, die du aus dem Internet unter

www.robocupjunior.de

beziehen kannst, fällt es dir sicher leicht, deinen Rescue Robot für diesen Wettbewerb zu programmieren.

Das linke Bild zeigt dir einen Parcours, den dein Roboter durchfahren muss. Auf dieser Strecke muss er verschiedene Aufgaben erledigen, z. B. einer Linie entlang fahren, verschiedenfarbige Figuren am Boden suchen oder durch eine Tür fahren usw. Wäre das nicht etwas für dich?



Wichtige Tipps

■ Der Spaß am Thema Robotik kann sehr schnell verloren gehen wenn der Roboter nicht so funktioniert wie man es gerne hätte.

Oft lassen sich mit einfachen Mitteln Fehler erkennen und beheben.

Kabel

Hier solltest du genau vorgehen. Zuerst werden die Kabel auf die vorgeschriebene Länge zugeschnitten und dann die Enden abisoliert und mit den Steckern fest verbunden. Überprüfe mit Hilfe eines Leuchtsteins (38216) mit Kugelstecklampe (37869) und dem Akku-Pack die Funktionsfähigkeit.

Stromversorgung

Oft ist ein fast leerer Akku Ursache von Fehlverhalten deines ROBO-Explorers. Sinkt die Spannung unter 5 V, wird dies mit einer roten LED am Interface angezeigt. Gleichzeitig schaltet das Interface automatisch ab. Ein Fehlverhalten kann auch auftreten, wenn der Akku noch nicht so leer ist. Auch dann muss der Akku geladen werden.

Programmierung

Sind alle mechanischen Probleme gelöst und der Roboter will immer noch nicht so recht, liegt es oft an einer fehlerhaften Programmierung. Hier bietet dir ROBO Pro den Online-Modus, in dem du den Programmfluss auf dem Bildschirm verfolgen kannst. Hier findest du meist die kleinen Programmfehler, die sich eingeschlichen haben.



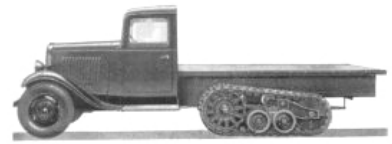
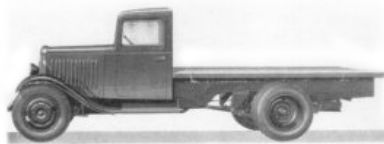
Contents



Vehicles with Crawler Drive	p. 18
The Steering	p. 18
Explorer Model from fischertechnik	p. 19
Actors	p. 19
Sensors	p. 19
ROBO Interface.	p. 21
Power Supply	p. 21
Software ROBO Pro	p. 21
Interface Expansions	p. 21
Preliminary Considerations	p. 22
Basics of the Interface	p. 22
The Basic Model	p. 23
Basic Program	p. 23
Pulse Control	p. 24
Subprograms	p. 24
Autonomous Tracked Vehicles	p. 25
Trail Searcher	p. 25
Tunnel Robot	p. 28
Color Detector	p. 29
Explorer	p. 30
RoboCupJunior – Rescue Robot	p. 32
Important Tips	p. 32

Vehicles with Crawler Drive

■ Why are autonomous vehicles with a crawler drive needed? The invention of the crawler drive for vehicles was necessary in order to be able to travel over very difficult terrain. The crawler drive was used in areas such as the desert where wheeled vehicles could not travel or at least only with extreme difficulty. For example, in the First World War, the first trucks and tanks with a crawler drive were built and employed.



Depending on the terrain, the vehicles could be converted from wheel drive to crawler drive.

Tracked vehicles were also used in the civilian area. As you can see from the pictures, wheel-driven vehicles were always the basis for tracked vehicles.



But it didn't take long to determine a weak point: The steerable front wheels. Therefore, the crawler drive was expanded and used for all axles.

The Steering



■ But then how did the steering work? Very simple, through the slowing or speeding up of one of the two tracks. If a curve to the right was to be made, the right track was slowed down by using the control stick, one per track. When this was done, this track moved slower and thus the vehicle moved to the right.

Even today, of course with the state of the art technology, you still find many track-powered vehicles. These range from small excavators to the gargantuans for the mining of brown coal in surface mining.



In the Cheops Pyramid in Egypt, a minirobot was to be used to help find the trail of more secrets.

The researchers sent the robot, which was about the size of a toy locomotive, through a dark narrow shaft. It led upwards from a chamber in the center of the pyramid, which was 4500 years old, and ended in front of a mysterious stone door.



■ Explore unknown areas, measure distances, follow trails, show driving directions by means of blinking signals, recognize colors, measure temperatures, avoid obstacles without touching them, recognize day and night, turn headlights on and off automatically and trigger an alarm etc. The ROBO explorer sensors can do all of this and lots more. More specifically this includes: The NTC resistor, the photoresistor, the ultrasonic distance sensor, the optical color sensor and the specially developed infrared trail sensor. Thanks to two power motors and the crawler drive even almost impassable terrain can be explored and traveled on. With the rescue robot, which is contained as a model, the construction set provides the ideal basis for participation in the RoboCup-Junior.

Now, before you get started, you should become familiar with the most important components. These are described in the following:

Power Motor

In order to drive and steer the tracks and thus your models, you need two power motors. These are connected on the interface to the outputs M1 to M4.

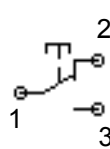
Domed Lamp

This is a light bulb for a voltage of 9 V. This can be used as an indicator signal for the direction of travel or simply for illumination. This is, as is the power motor, connected to the interface outputs M1 to M4.

Buzzer

The buzzer is used, for example, to acoustically report obstacles or colors, which are identified. It is also connected to the interface outputs M1 to M4.

Push Button Switch



Push button switches are in the category of contact sensors. If you push the red button, then a contact is mechanically moved in the casing and electrical current flows between the contacts 1 and 3. At the same time the switch circuit between the contact points 1 and 2 is interrupted.

Push button switches or switches are used in two different ways:

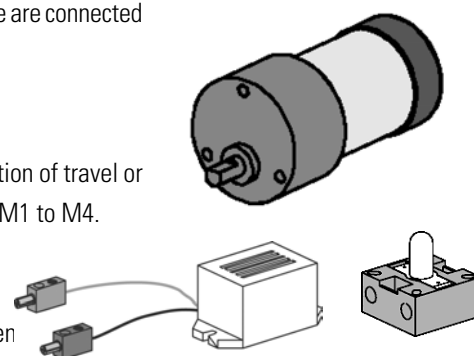
Push button switch as a „**normally open contact**.“

Both circuit diagrams show you the test setup. A voltage source (9 V) is connected to contact 1 of the push button switch, a lamp to contact 3 of the push button switch and to the negative pole of the voltage source.

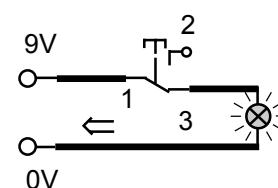
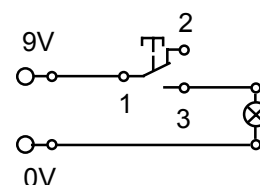
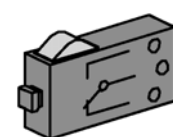
If the push button switch is not pushed then the lamp is off. If you push the push button switch then the circuit through contact 1 and contact 3 is closed, and the lamp lights up.

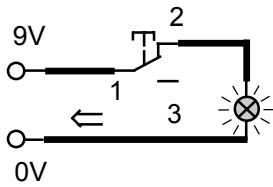
Explorer Model from fischertechnik

Actors



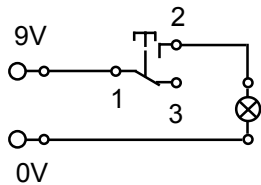
Sensors





Push button switch as a „**normally closed contact**.“

Both circuit diagrams here also show you how this works. Contact 1 is connected with the voltage source. Contact 2 with the lamp and the lamp with the negative pole of the voltage source.



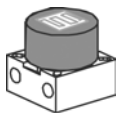
The lamp lights up. If you operate the switch now, the electrical current is interrupted and the lamp goes off.

In your models, you use the push button switch together with the pulse wheel as a counting push button switch.



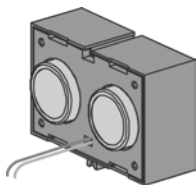
NTC Resistor(Negative Temperature Coefficient)

Here, this is a component, with which you can measure various temperatures. You can also call this a heat sensor. At about 20 degrees, the NTC resistor has a value of 1.5 kOhms. When the temperature increases, the resistance value decreases. This information can be read into the interface at the analog inputs AX and AY and is available to ROBO Pro as a numerical value from 0 to 1023.



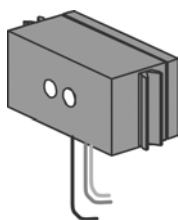
Photoresistor

The LDR 03, which is an analog brightness sensor for the inputs AX and AY (resistance measurement), reacts to daylight and changes its resistance value when it does this. This is an indicator for the brightness of the light.



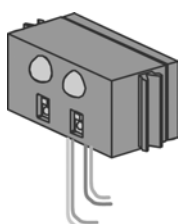
Ultrasonic Distance Sensor

A distance sensor is a technical component, which has the capability to measure the distance between itself and an object. Distance sensors work with light, infrared radiation, radio waves and **ultrasonic sound** and use various measuring methods. Sound is propagated as a wave. An echo is reflected back to the source of the ultrasonic sound and the echo is then received as a signal and evaluated. The time difference between the transmission and reception of the signal provides information about the distance between the obstacle and the sensor. D1 and D2 are available as connections on the interface. The range of the sensor is up to four meters. The numerical value, which is shown, corresponds to the distance in centimeters.



Optical Color Sensor

Color sensors are mostly used in automation engineering. When this is done, for example, the color or the color imprint is to be examined to insure that the correct components are installed. The fischertechnik color sensor transmits red light, which is reflected with a different strength from different colored surfaces. The quantity of the reflected light is measured by the phototransistor and outputted as a voltage value between 0 V and 10 V. The measured value is dependent on the ambient brightness and the distance of the sensor from the colored surface. You can read in this value through the inputs A1 and A2 and process the numerical value from 0 to 1000 in your program.



Trail Sensor

The IR trail sensor is a digital infrared sensor for the identification of a black trail on a white background at a distance of five to 30 mm. It consists of two transmission and two receiver elements. For connection, you need two digital inputs and the 9 V voltage supply (plus and minus).

■ The most important component for building a tracked vehicle is the ROBO Interface, which is installed as a fixed element in the various models. You can connect your sensors and actors to this as needed. You can find the basic wiring in the attached construction instructions.

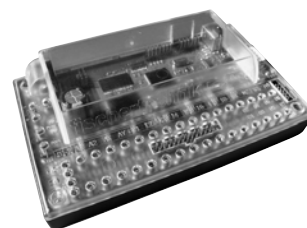
■ The ROBO Explorer models are autonomous vehicles, which move freely in the area. Therefore, you use the fischertechnik rechargeable battery set as the power supply.

■ ROBO Pro is a simple graphic programming interface, with which you can write your programs. The advantage is that you don't have to learn any programming language. Actually, you can start right now.

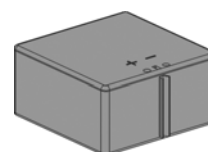
You need the ROBO Pro Version 1.2.1.30 for the construction set, ROBO Explorer. If you have an older version of the software, you can update this at no charge. Download this with the help menu in ROBO Pro—download new version—or at:

www.fischertechnik.de/robopro/update.html.

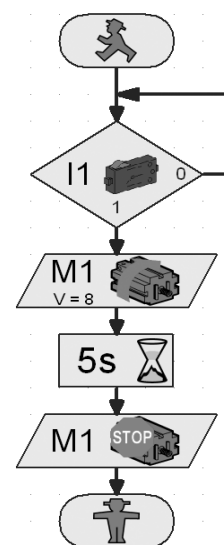
ROBO Interface



Power Supply



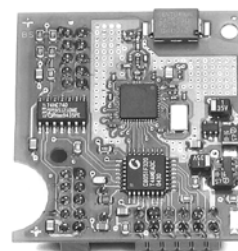
Software ROBO Pro



ROBO RF Data Link

The radio interface, ROBO RF Data Link, replaces the interface cable between the PC and the interface. The data are entered into the interface by means of radio transmission. The advantage is that the cable, which is often bothersome, is no longer needed. An additional advantage is that you can operate programs over radio in the online mode. In this way, you can find errors easier than in the download mode. And this method also has another advantage, mobile robots can be controlled online through the monitor and can transmit measured values to the computer without any cable. If several robots are equipped with a data link, they can even exchange data between each other, so to say „talk“ with each other.

Interface Expansions



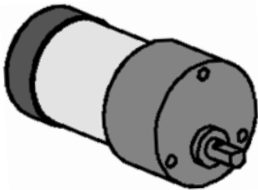
Preliminary Considerations

■ As for all other fischertechnik robots, with the ROBO Explorer, you will be introduced step-by-step to the fascination of technology and programing. You start with a simple model and work your way forwards to increasingly more comprehensive systems with fascinating possibilities. The primary focus for all models is the precise construction and precise placing in operation.

Basics of the Interface

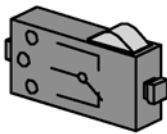
■ Before you start with the individual models, you should become familiar with the interface with the help of some experiments. For this purpose, put the handbook for the ROBO Pro software by your side so that you can look in the handbook if you have problems.

After you have installed the software, you can connect the interface using the cable, which is supplied, to your PC. Now start the program, ROBO Pro, and open the „Interface test.“



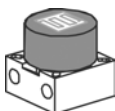
Power Motor

Connect the connections of the power motor with the connection M1. Click with the mouse on the selection „ccw“ or „cw.“ The motor starts with its maximum speed. By operating the control, you can adjust the RPMs. Use „Stop“ to stop the process.



Push Button Switch

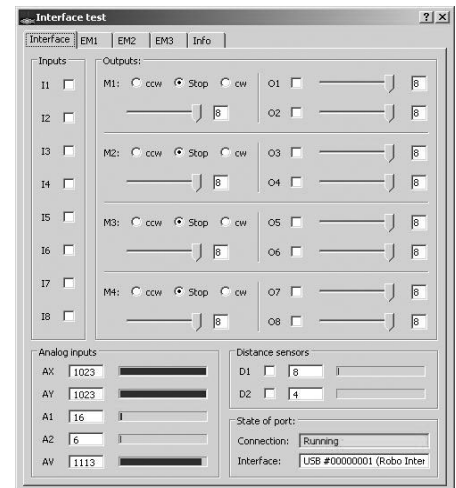
Now, connect a push button switch (normally open contact) to the digital input I1 and observe what happens in the test display when you push the push button switch. An optical display (check mark) shows you the status of the push button switch. Connect the push button switch as a normally closed contact and watch the result. First, the status display for „Switch Closed“ appears. If you push the switch, the information displayed disappears.



Photoresistor

Connect the photoresistor, which was supplied to you, to the connection AX or AY. Change the luminous intensity of the photoresistor by slowly covering with a black strip of paper. What happens? You see the blue light bar and the numerical value for the input used changes.

The test of the interface is explained very well in chapter 2 of the handbook. You can also find help if there are problems between your computer and the interface and the software, take a look, its worth it!



■ Now, let's get started. Now that you have found out how the interface and the programming work, you can solve the first task. First, the basic model is to be built using the construction instructions.



Task 1- ROBO Pro Level 1

Your tracked vehicle is to travel six seconds straight ahead and then it is to turn to the left for three seconds and then stop.

We want to give you some support with your first program. First, click on the button „New File.“

Your program starts with a little green traffic light figure for program start.



After this, you need two motor symbols. Place the first symbol under program start so that the connection is automatically made. Move the mouse to the motor symbol and turn on the properties window (right mouse button). There, you set the motor output to „M1“ and for action the direction of rotation, „Left.“ Then confirm with OK. Insert the second motor symbol in the same way and repeat the action for the motor output, „M2.“

Next, the program is to wait for a certain time. For this, use the waiting time symbol. You place this under the second motor and adjust the time to six seconds.

Following this, the tracked vehicle is to turn for three seconds. To do this, you again insert two motor symbols for M1 and M2. M1 is to rotate to the left and M2 to the right. Since both motors are to work for three seconds, now insert the waiting time symbol and set the time value to three seconds.

After this, you have to stop both motors. This done as before the rotation by inserting both motor symbols and the parameter setting, „Stop.“

At the end, you must then insert the symbol for the end of the program, the „red little traffic light figure.“ Now, your first program is finished and you can store it. Then test it in the online mode. To do this, click on the „Start“ button.

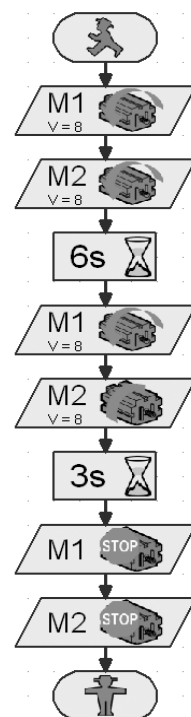
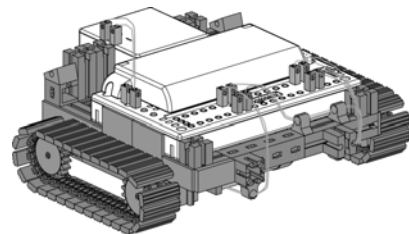
If you did everything right, then you can load the program onto the interface. To do this, click on the button „Download the Program to the Interface.“ Accept the setting of the download window. Right after the download, the model starts. Unfortunately, it is still connected with the USB cable. Load the program once again, but activate the „Start Program with Push Button Switch on Interface.“ Once the program has been transferred, you can pull the cable out. To start the program, press the „Prog Button“ on the interface.

You can find the finished program at:

C:\Programs\Robopro\SamplePrograms\Robo_Explorer\Basic_Model_1.rpp.

That was really to simple for you. But, as we already said, you should move forward slowly. For this purpose, the task will be expanded somewhat.

The Basic Model

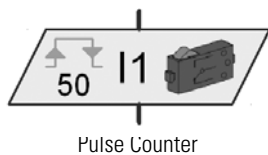
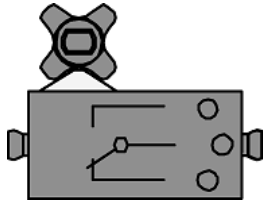


Start program in online mode



Download program to ROBO interface

Pulse Control



■ The basic model already contains two sensors (push button switches) for measuring distances and one pulse wheel for each, which is connected with the motor axle. This pulse wheel operates the switch four times for each revolution. Check the model with „Interface test“ for proper functioning.

Task 2 - ROBO Pro Level 1

Your tracked vehicle is to move 50 pulses forward. Following this, it is to travel 11 pulses to the left and then stop. You can find information about the pulse counter in the handbook in chapter 3.6.3.



Solution

First turn both motors on. Then add the pulse counter to these. In the properties change the number of pulses to 50. Again, insert two motor symbols. M1 continues to rotate to the „ccw,“ switch M2 to „cw.“ This turning is to last for 11 pulses. Finally, you must again insert two motor symbols and add „Stop.“ Transfer the program and test it on your vehicle.

When in the properties window for the pulse counter, you set the pulse type to „0->1“ or „1->0,“ you get eight instead of four pulses per rotation of the pulse wheel, which means the number of the pulses per distance traveled doubles and the precision of the measuring of the distance increases.

You can find the finished program at:

C:\Programs\Robopro\SamplePrograms\Robo_Explorer\Basic_Model_2.rpp.

Subprograms



Generate new subprogram



Copy current subprogram



Delete current subprogram

■ When you were making the program, you certainly noticed that you had to set the motors for every change of direction or stop. As a result, big programs become complicated very fast and the search for errors often becomes a test of patience.

Here, ROBO Pro offers „subprograms,“ which you can work with, as an elegant solution. Read all of chapter 4.1 in the handbook on this subject. It is important that you switch to level 2 in ROBO Pro.



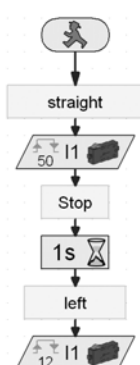
Task 3 - ROBO Pro Level 2

Your tracked vehicle is to move around a square. As for the first task, use the parameters 50 and 11. Make a subprogram for each direction of travel and for the Stop command.

First, make the subprogram „Forwards“ (see ROBO Prof Handbook chapter 4). Mark the program parts and copy them to the cache. Following this, prepare the subprogram, „Left“ and „Stop.“ Insert the program parts for „Forwards“ into both from the cache and change the parameters accordingly. For the turn, use a lower speed of rotation. As a little bit of help, we show you a part of the task.

The following table is to show you at a glance how you have to program the motors for the directions of travel.

Solution



Direction of Travel	Direction of Rotation Motor 1	Direction of Rotation Motor 2
Forwards	CCW	CCW
Backwards	CW	CW
Left	CCW	CW
Right	CW	CCW
Stop	Stop	Stop

Using this table, all motors in the example programs can be programmed.

Finished Program:

C:\Programs\Robopro\SamplePrograms\Robo_Explorer\Basic_Model_3.rpp

Task 4 - ROBO Pro Level 3

The robot does not travel in a precise square. Check the following questions: What is the cause? How can the result be improved?



Tip:

Synchronize the motors M1 and M2 using the push button switches I1 and I2 so that the model travels exactly straight ahead. For this purpose, we have supplied a finished program, „forward_sync“ for you. Here you only have to enter the number of pulses as a constant in the orange subprogram input (see Basic_Model_4.rpp). You can also enter the number of pulses for the angle of rotation in exactly the same way in the subprogram, „Turn_sync.“

Finished Program:

C:\Programs\Robopro\SamplePrograms\Robo_Explorer\Basic_Model_4.rpp

■ After you have experimented with the basic model enough, your robot is now to react to various signals from an external source.

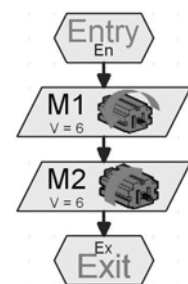
In order to allow your tracked vehicle to identify its environment and to complete certain tasks, you must equip it with sensors. The following model suggestions show you different variations of tracked vehicles with various sensors. This is to provide the capability to identify different paths, light and colors or even heat sources or distances. You can find the individual programs in the directory:

C:\Programs\Robopro\SamplePrograms\Robo_Explorer\

Of course you have seen factory buildings, which have no human beings in them, in TV movies where the transport vehicles in the buildings travel as if steered by the hand of a ghost. In some cases, such systems are controlled by data lines, which are embedded in the floor, or by markings on the floor to indicate the path.

The basis of your programming is to be that the robot travels along a black line.

Before you start with the programming, first build the trail searcher using the construction instructions. You can find an experiment course with a printed black line in the construction set. The line, which is to be followed by the trail searcher, is to be straight at first.



Subprogram, „Left“

Autonomous Tracked Vehicles

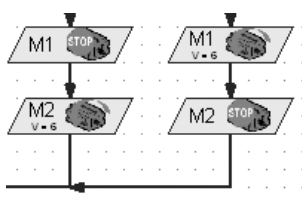
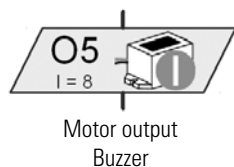
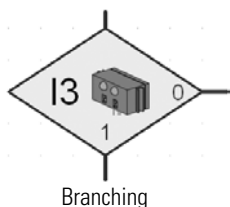
Trail Searcher





■ Now, how is the model to function?

The robot is to find a black line on a white background and then follow this line. To make this possible, you have installed an IR trail sensor in your model. The module transmits a light in the infrared range to the pavement surface. Depending on the background, this is reflected and measured by phototransistors. For your programming, this means: A bright and white background reflects the light and you receive the value 1. With a black background, the light is not reflected and you receive the value 0. If both transistors have the value 0, then the robot has found the lane (the black line) and must now follow this line.



Task 1 - ROBO Pro Level 2

Your tracked vehicle is to be placed on a straight black trail and then travel along this. If it loses the trail or if the trail ends, it is to stop and transmit an acoustic signal for one second.

Subdivide the entire program into a main program for the sensor interrogation and a subprogram for the travel straight ahead, one for the buzzer signal and one for stop.

A Few Little Tips:

Use the interface test to check the trail identification by the sensor. If the black-white identification does not function properly, then interfering light sources such as the sun could be the cause. If necessary, the sensor must be positioned somewhat closer to the trail or be shielded with a building board.

Finished Program: **Trail_searcher_1.rpp**

■ Of course, you will not be satisfied with your first solution because your robot can certainly travel a short stretch along the line. However, because it cannot yet readjust, it leaves the marking and stops and sends you a signal indicating this.

Task 2 - ROBO Pro Level 2

Expand your main program in the interrogation branches of the phototransistors so that the robot recognizes when it is not traveling exactly on the trail. Then, it is to correct the direction of travel accordingly. You can find a tip in the left program section.

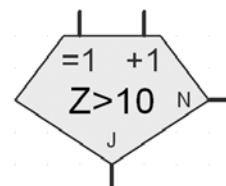


Now it looks better already. Your robot remains precisely on the prescribed trail. In an industrial building, other robots would remove the load, which was transported, at the end of the trail or would give the robot a new load. This load could then be transported back to the starting point of the trip.

Finished Program: **Trail_searcher_2.rpp**

Task 3 - ROBO Pro Level 2

Your robot traveled along a certain black line in the previous tasks. For this task, it is to search for a line. To do this, it is to turn in a circle one time. If it doesn't find a trail when it does this, it is to travel a short distance straight ahead and then search again. If your robot finds a trail, then it is to follow the trail. If the trail ends or it loses the trail, then it is to start the search again.



Counter loop

Tip:

Think about the first tasks with the basic model. Here, the robot was supposed to turn 90 degrees. You did this with a distance traveled control. This technology can be of use to you here as well.

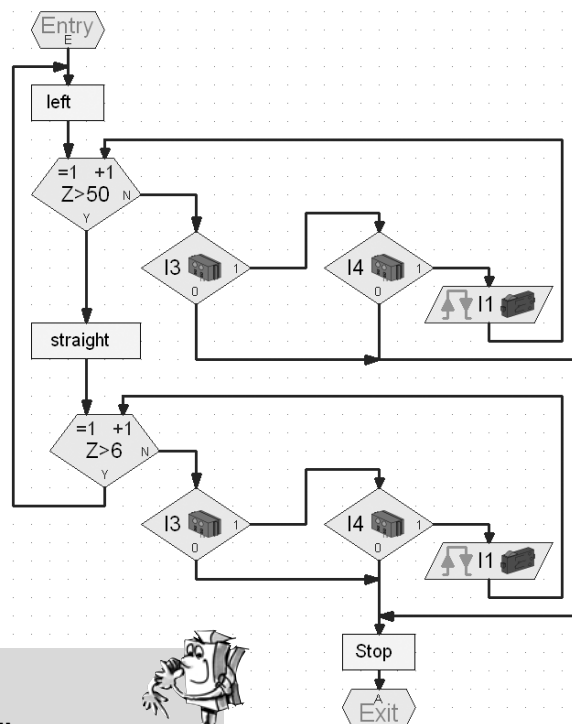
For the trail search, prepare your own subprogram with the name „track-seeking.“ In the diagram, you see how we suggest this be done.

Write another subprogram with the name, left. You have already created forwards and stop.

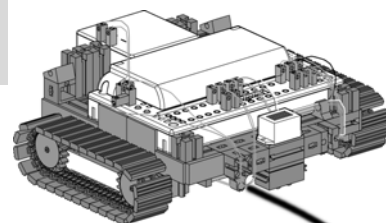
Thus, the main program and all subprograms for the trail searcher are finished. Now, your task is to integrate these into the program so that task 3 can be completed.

Finished Program: **Trail_searcher_3.rpp**

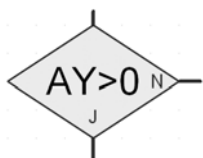
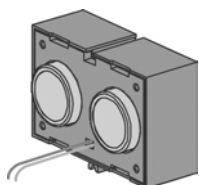
The previous routes were always straight. However, routes sometimes also have curves. You can find these, for example, in industrial facilities, in which material or workpieces must be transported from one machine to the next one.

**Task 4 - ROBO Pro Level 2**

The experiment course contains various curves with differening radiuses. Experiment with various speeds for M1 and M2 for your circuit. What settings allow the robot to complete the course the fastest? Record the results in a small table.



Tunnel Robot, Distance and Temperature Sensor



- The sensor for distance measurement and the NTC resistor offer you additional possibilities to expand your model to a professional robot.

Measuring distances and temperatures and initiating possible rescue measures. What do you think, where could robots with such capabilities be used? Of course, you can think of various areas, where they could be used. We want to examine the area of fire protection and fire fighting in auto and train tunnels.



The task for such a robot is to approach the source of the fire, measure temperatures in the tunnel and to report data to a control center. In most cases, the robots are equipped with mobile extinguishing equipment, which can be used depending on the conditions.

Now here conscientiously build the model, „tunnel robot,“ according to the construction instructions as well. The distance sensor is connected to the connections D1.

Task 1 ROBO Pro Level 2

Similar to the trail searcher, which travels along a line, your robot is to travel a certain route along a wall at a certain distance of about 20 centimeters.



Finished Program: **Tunnel_1.rpp**

- Let's examine a fire extinguisher robot once again for the next task. In order that it can travel along the wall like your robot, it uses distance sensors. However, in order to identify the source of the fire, it employs heat sensors. This heat sensor is the NTC resistor for your model. The physical characteristic of this component is that the resistance value decreases when the temperature increases. You can try out this change again with the interface test. Connect the NTC resistor to the connection, AY. Hold a heat source close to the NTC and observe the blue bar for AY.

Task 2 ROBO Pro Level 2

Expand the program, which allows the robot to travel along the tunnel wall. As an addition, measure the current heat with the AY connection. If this increases to a certain value then the robot is to stop and transmit a warning signal using the buzzer. At the same time as the buzzer, the red indicator light is to blink.

Following this simulated extinguishing action, your robot is to turn and return to the starting point.



Finished Program: **Tunnel_2.rpp**

Tip:

Since your robot only has one distance sensor, it needs a second wall, along which it can travel, to return to the starting point.

If you have another motor and a propeller in your fischertechnik collection, you can also integrate the extinguishing of a fire into your program.

■ You will get to know the color sensor as an additional sensor. The picture on the side gives you an example for industrial use. As you can see, the colored cans, which were placed in the wrong row, are being sorted out.

The light, which is reflected from the material being examined, is received, digitalized and processed using a computer and software. The task of the sensor is to identify the various colors and to transmit the measurement data to the interface.

The color sensor is installed in the color detector model. This is connected with the black wire to A1, the red wire to plus and the green wire to minus. For the first test programs, you are to use the colored surfaces imprinted on the course.

Task 1 - ROBO Pro Level 2

First, check the values, which the interface puts out for the various colors in the interface test. Use black and white in addition to the three colors prescribed.

Make a small table and enter the values, which you have measured. Also observe the changes when the distance to the colored surface or the ambient light changes.

**Task 2 - ROBO Pro Level 2**

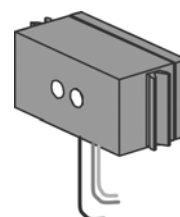
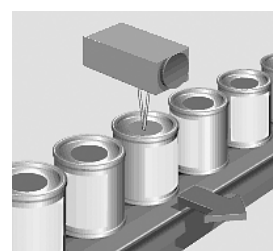
Write a small program, with which the sensor can identify the green colored area. If the measured value is within the prescribed range of values, the buzzer is to be activated for one second. Following this, the program jumps to start.

Finished Program: **Color_detector_2.rpp**

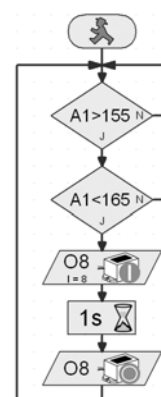
Tip:

For the next task, you need three lamps with different colored luminous caps, which are already installed in the model.

Color Detector



Color	Value
White	
Black	
Blue	
Red	
Green	





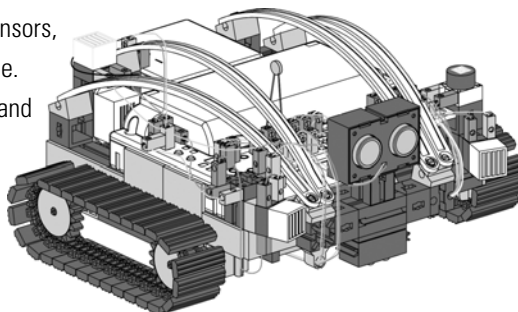
Task 3 - ROBO Pro Level 3

Write a program which allows your robot to travel straight ahead on a certain route. There are three colored areas on this route. If the sensor detects a color, the robot is to stop for three seconds. During this time, it turns on the lamp with the corresponding color and emits an acoustic signal with the buzzer. Following this, it travels to the next area and repeats its work. Following this, it travels to the last area, reports the result and remains there.

Finished Program: **Color_detector_3.rpp**

Explorer Model

■ The „Explorer“ model contains all actors and sensors, which are needed for an autonomous robotic vehicle. Now, you have no limits for the solution of simple and difficult tasks. In the preceding construction stages, you have only used one sensor in most cases to become familiar with the possible areas of employment.



Task 1 - ROBO Pro Level 2

Program your robot so that it travels up to an existing obstacle on its trip. At a distance of about 60 centimeters, it is to reduce its speed by half. At a distance of 40 centimeters, it is to stop. If the obstacle gets closer to your robot, the robot is to travel slowly at a distance of 20 centimeters and at 10 centimeters, it is to travel backwards fast.

Finished Program: **Explorer_1.rpp**

Task 2 - ROBO Pro Level 2

Now, your robot is going on an exploration trip. Prepare a program for the use of two sensors: trail sensor and distance sensor. First, the robot is to follow the black line on the experiment course. Place an obstacle on the route. The robot is to stop about 10 centimeters in front of the obstacle and travel one centimeter backwards. Following this, it is to turn around and follow the trail in the other direction.



Finished Program: **Explorer_2.rpp**

Task 3 - ROBO Pro Level 2

The program from task 2 is to be expanded with three sensors: the color sensor, the temperature sensor and the photoresistor for the measurement of brightness.

There are various colored areas along the trail. The robot reports these with various acoustic signals. If the ambient temperature becomes too high during the trip, the red indication light is to blink. As soon as the room becomes dark, the robot turns on its two headlights. If it gets bright again, the headlights are turned off.



Finished Program: **Explorer_3.rpp**

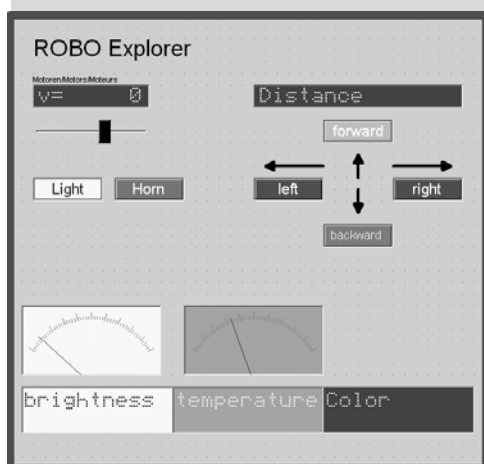
■ In 2003, Mars was closer to the Earth than almost ever before. The Europeans and Americans exploited this to send up their own exploration vehicles. Their mission: To prove life.

Our explorer can also be programed as a remote controlled robot to explore unknown worlds. The RF data link is needed additionally for this purpose.

Mars - Start to the Fourth Planet

**Task 4 - ROBO Pro Level 3**

For this task, your computer is the control center for a Mars expedition. The mission is to transmit the measured values of a Martian landscape to the ground station. The control of your robot is generated in ROBO Pro in the Panel (see handbook chapter 8).

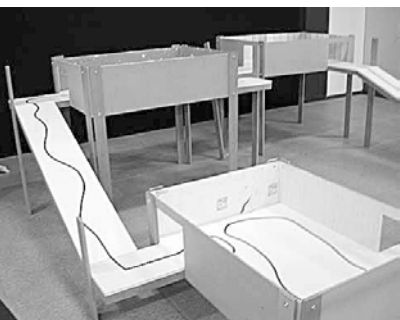


Your robot is to be programed so that it transmits measured values for the ground color, temperature, brightness and obstacles. The robot is to be controlled by hand through the "Panel" of the main program in ROBO Pro.

Finished Program: **Explorer_4.rpp**



RoboCup Junior Rescue Robot



Now, you have performed all tasks, became familiar with the construction of a robot and the programming technology, so now you could harvest the fruits of your labor and perform in the Rescue League of the RoboCupJunior with the „rescue robot“ model.

■ RoboCupJunior is a worldwide, project oriented education initiative, which promotes regional, national and international robot events for young people. The goal is to introduce children and young people to robots and their application.

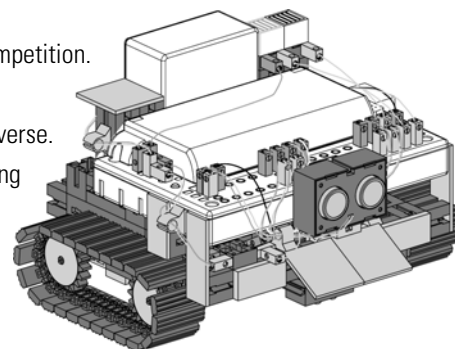
Using the tasks, which you can obtain from the Internet at

www.robocupjunior.de,

it will certainly be easy for you to program your rescue robot for this competition.

The picture on the left shows you a course, which your robot must traverse.

On this route, it must complete various tasks, for example, traveling along a line, searching for various figures on the floor or travel through a door and so forth. Wouldn't that be something for you?



Important Tips

■ The fun with the subject of robotics can be lost very fast if the robot doesn't function like you want it to function.

Often, simple means can be used to identify and eliminate errors.

Cables

Here, you must proceed precisely. First, the cables are cut to the prescribed length and then the ends are to be stripped of insulation and connected with plugs. Check for proper functioning using a plug in light holder (38216), a ball plug-in light (37869), and the rechargeable battery pack.

Power Supply

Often, a rechargeable battery, which is almost totally discharged, is the cause of erroneous behavior by your ROBO Explorer. If the voltage goes below 5 V, this is indicated by a red LED on the interface. At the same time, the interface is automatically turned off. Improper functioning can also occur even if the rechargeable battery is not so low. But then the rechargeable must still be charged.

Programing

When all mechanical problems have been solved and the robot is still not quite right, this is often due to errors in programming. Here, ROBO Pro offers you the online mode, which allows you to follow the program flow on the monitor. In most cases here, you can find the small program errors, which have crept in.



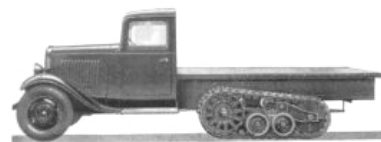
Sommaire



Véhicules avec traction à chenilles	Page 34
Le pilotage	Page 34
Maquette Explorer en fischertechnik	Page 35
Actionneurs	Page 35
Capteurs	Page 35
ROBO Interface	Page 37
Alimentation en courant électrique	Page 37
Logiciel ROBO Pro	Page 37
Extensions d'interface	Page 37
Réflexions préliminaires	Page 38
Principes de l'interface	Page 38
La maquette de base	Page 39
Programme de base	Page 39
Commande par impulsions	Page 40
Sous-programmes	Page 40
Véhicules sur chenilles autonomes	Page 41
Dépisteur	Page 41
Robots pour tunnels	Page 44
Identificateur de couleurs	Page 45
Explorer	Page 46
RoboCupJunior – Robot de sauvetage	Page 48
Tuyaux et astuces	Page 48

Véhicules avec traction à chenilles

■ A quoi servent les véhicules autonomes dotés d'une traction à chenilles ? L'invention de la traction à chenilles pour divers véhicules étaient indispensable pour surmonter les difficultés des terrains accidentés. La traction à chenilles s'emploie chaque fois que la traction à roues ne fonctionne plus, p. ex. dans un désert. Ainsi, les premiers camions et chars dotés d'une traction à chenilles ont été construits et utilisés durant la Première guerre mondiale.



La conversion de la traction à roues en traction à chenilles était possible chaque fois que le terrain à parcourir l'imposait.

L'emploi de véhicules à chenilles s'est avéré tout aussi utile dans le domaine privé. Les photos montrent cependant que les véhicules à chenilles étaient pratiquement toujours basés sur des véhicules à traction à roues.

Sauf que ce type de construction présentait un point faible évident : les roues avant orientables. Et c'est aussi pour cette raison que tous les essieux ont finalement été dotés d'une traction à chenilles.

Le pilotage



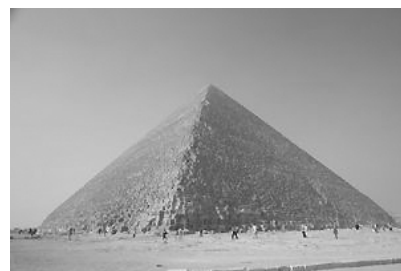
■ Mais comment le pilotage d'un tel véhicule fonctionne-t-il ? Tout simplement par ralentissement ou accélération d'une des deux chenilles. Il suffit de se servir de la manette de commande (une manette par chenille) pour ralentir la chenille de droite si on veut prendre un virage à droite. La chenille en question tourne plus lentement et le véhicule est déplacé vers la droite de ce fait.

Jadis comme aujourd'hui, vous trouverez d'innombrables véhicules dotés d'une traction à chenilles, toujours dans le respect des Règles les plus récentes de l'art, cela va de soi. De la petite pelle jusqu'aux colosses gigantesques pour l'extraction de la lignite à ciel ouvert.



On s'était par exemple aussi servi d'un robot miniature pour tenter de découvrir d'autres mystères de la pyramide de Cheops en Egypte.

Les chercheurs ont utilisé un robot de la taille d'une locomotive miniature pour explorer un puits sombre et étroit. Ce puits mène à une chambre dans le centre de la pyramide ancienne de 4500 ans et se termine face à une mystérieuse porte de pierres.



■ Découvrir des espaces inconnus, mesurer des distances, suivre des pistes, annoncer les orientations de marche par des signaux clignotants, identifier les couleurs, mesurer les températures, éviter les obstacles sans contact, identifier le jour et la nuit, mettre les phares automatiquement en circuit et hors circuit, déclencher l'alarme etc. Tout ceci – et bien plus – est possible grâce aux capteurs ROBO Explorer. Il s'agit notamment des pièces suivantes : résistance NTC, photorésistance, capteur d'espacement ultrasonique, capteur optique chromatique, de même que le dépisteur aux infrarouges spécialement conçu à cet effet. Deux moteurs électriques performants et la traction à chenille permettent même d'étudier des terrains accidentés et de s'y déplacer. La maquette du robot de sauvetage contenue transforme cette boîte de construction en base idéale pour la participation aux championnats RoboCup-Junior.

Nous vous recommandons cependant de vous faire une idée des principaux éléments de construction avant de vous lancer dans la construction. Ces éléments sont décrits ci-après :

Moteur électrique

Vous devez disposer de 2 moteurs électriques pour actionner et piloter les chenilles, et vos maquettes de ce fait. Ces moteurs sont raccordés aux sorties de M1 à M4 de l'interface.

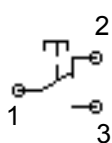
Lampe ronde

Il s'agit d'une lampe à incandescence d'une tension de 9 Volts, que vous pouvez utiliser comme signal d'état du sens de marche ou comme éclairage, tout simplement. Elle est raccordée aux sorties de M1 à M4 de l'interface, comme le moteur électrique.

Sonnerie

La sonnerie sert par exemple à la signalisation acoustique des obstacles ou couleurs détectés. Son raccordement se fait également aux sorties de M1 à M4 de l'interface.

Boutons



Les boutons forment partie du groupe des capteurs tactiles. L'actionnement du bouton rouge a pour effet de provoquer un renversement mécanique du contact dans le boîtier et d'assurer la conduction de courant entre les contacts 1 et 3. La ligne de commutation entre les points de raccordement 1 et 2 est simultanément interrompue.

Les boutons ou commutateurs sont utilisables de deux façons différentes :

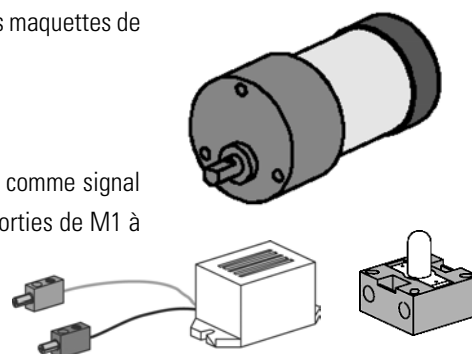
Bouton comme « **contact à fermeture** »

Les deux schémas de connexion montrent la structure utilisée pour le test. Une source de tension (9 Volts) est reliée au contact 1 du bouton, tandis qu'une lampe est reliée au contact 3 du bouton et au pôle négatif de la source de tension.

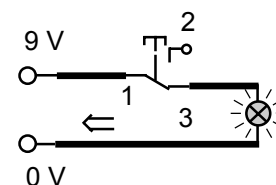
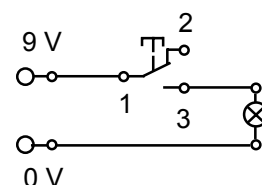
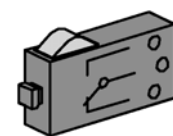
La lampe demeure éteinte tant que le bouton n'est pas actionné. L'actionnement du bouton a pour effet de fermer le circuit électrique entre les contacts 1 et 3 ; la lampe brille.

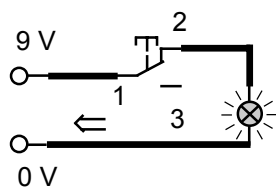
Maquette Explorer en fischertechnik

Actionneurs



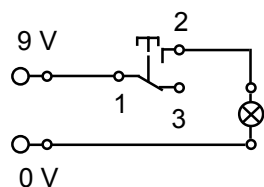
Capteurs





Bouton comme « **contact à ouverture** »

Les deux schémas de connexion montrent le fonctionnement. Le contact 1 est relié à la source de tension. Le contact 2 est relié à la lampe et la lampe est reliée au pôle négatif de la source de tension.



La lampe brille. L'actionnement du commutateur a alors pour effet d'interrompre le circuit électrique ; la lampe s'éteint.

Vous utilisez les boutons ensemble avec la roue à pulsations comme compteur dans vos maquettes.

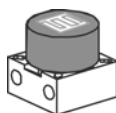
Résistance NTC (Négatif Température Coefficient ou coefficient de température négative)

Il s'agit d'un élément de construction permettant de mesurer différentes températures. Ce type d'élément est également appelé « capteur thermique ». La résistance NTC possède une valeur de 1,5 kOhm en présence d'une température d'environ 20 degrés Celsius. La valeur de résistance baisse dès que la température augmente. L'importation de cette information dans l'interface est possible au niveau des entrées analogiques AX et AY ; elle est aussi disponible dans ROBO PRO comme valeur exprimée en chiffres de 0 à 1023.



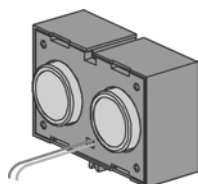
Photorésistance

Le LDR 03, un capteur de luminosité analogique pour les entrées AX et AY (mesure de la résistance électrique), réagit à la lumière du jour et modifie sa valeur de résistance en conséquence. Il assume donc la fonction d'un indicateur de luminosité.



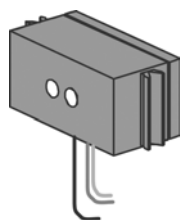
Capteur d'espacement ultrasonique

Un capteur d'espacement est un élément de construction technique capable de mesurer la distance entre soi et un objet quelconque. Les capteurs d'espacement fonctionnent à la lumière, au rayonnement infrarouge, aux ondes radioélectriques ou aux **ultrasons** et emploient différentes méthodes de mesure. Le son se propage comme onde. L'écho renvoyé à la source ultrasonique est capté sous forme d'un signal et interprété. La différence de temps entre l'expédition et la réception du signal permet de saisir la distance entre l'obstacle et le capteur. Les raccords respectifs D1/D2 sont prévus sur l'interface. Le capteur est doté d'une portée de jusqu'à 4 mètres. La valeur exprimée correspond à l'espacement en centimètres.



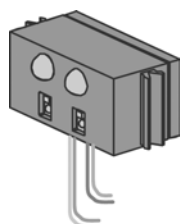
Capteur optique chromatique

Les capteurs chromatiques sont habituellement utilisés en technique d'automatisation. Il s'agit, par exemple, de contrôler la couleur ou les couleurs imprimées afin de garantir le montage des éléments de construction corrects dans un tout. Le capteur chromatique fischertechnik émet une lumière rouge, qui est réfléchiée avec une intensité différente en fonction des couleurs de la surface respective. Le volume de lumière réfléchi est mesuré par un phototransistor et exprimé sous forme d'une valeur de tension entre 0 Volt et 10 Volts. La valeur mesurée dépend de la luminosité environnante et de la distance du capteur vis-à-vis de la surface de couleurs. Vous pouvez importer cette valeur aux moyens des entrées A1 et A2 et traiter la valeur située entre 0 et 1000 dans votre programme.



Dépisteur

Le dépisteur IR est un capteur infrarouge numérique de détection d'une piste noire sur une surface blanche à intervalles de 5 à 30 mm. Il est composé de deux éléments d'émission et de deux éléments de réception. Vous devez disposer de deux entrées digitales pour son raccordement et d'une alimentation en tension de 9 Volts (positive ou négative).



■ Le principal élément pour construire un véhicule à chenilles est l'interface ROBO à intégrer dans les différentes maquettes dès la construction. C'est à cette interface que vous devez raccorder vos capteurs et actionneurs respectifs. Veuillez consulter les instructions de construction ci-jointes en ce qui concerne le câblage de base.

■ Les maquettes ROBO Explorer sont des véhicules autonomes, qui se déplacent librement dans la pièce. L'alimentation en courant électrique devrait donc se faire au moyen de l'Accu Set fischertechnik.

■ ROBO PRO est une simple interface de programmation graphique vous permettant d'écrire vos propres programmes. Vous ne devez maîtriser aucun langage de programmation et ceci est un des principaux avantages. Vous pouvez donc démarrer dans l'immédiat.

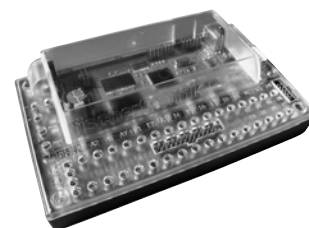
Vous devez disposer de la version ROBO Pro 1.2.1.30 pour utiliser la boîte de construction ROBO Explorer. Une mise à jour est disponible gratuitement, pour autant que vous disposiez d'une version du logiciel plus ancienne. Pour ce faire, veuillez passer par le menu de l'aide dans ROBO Pro – pour télécharger la nouvelle version ou adressez-vous directement à

www.fischertechnik.de/robopro/update.html

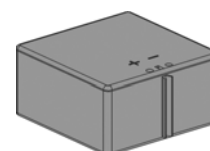
ROBO RF Data Link

L'interface radiotélégraphique ROBO RF Data Link remplace le câble d'interface entre le microordinateur et l'interface. La transmission radiotélégraphique transmet les données à l'interface. Ceci présente l'avantage de pouvoir renoncer au câble souvent onéreux. Un avantage supplémentaire réside dans l'utilisation radio des programmes en mode Online. Ainsi, la détection des défauts est plus facile qu'en mode de téléchargement. Sans oublier que cette méthode présente un avantage complémentaire – la commande de robots mobiles se fait virtuellement à l'écran et la transmission des valeurs mesurées à l'ordinateur s'effectue sans fil. L'équipement Data Link de plusieurs robots permet aussi de procéder à un échange des données entre les robots, qui peuvent donc « communiquer ou se parler entre-eux ».

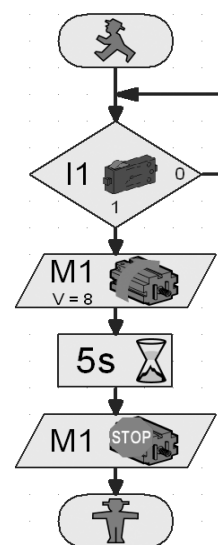
ROBO Interface



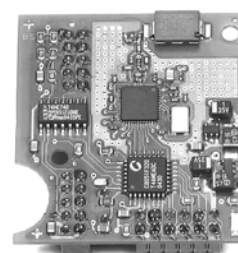
Alimentation en courant électrique



Logiciel ROBO Pro



Extensions d'interface



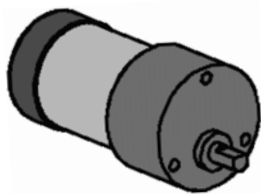
Réflexions préliminaires

Principes de l'interface

■ ROBO Explorer vous familiarisera progressivement avec l'univers fascinant de la technique et de la programmation, comme ceci est par ailleurs aussi le cas de tous les robots fischertechnik. Vous commencez par une simple maquette et progressez jusqu'à l'atteinte de systèmes de plus en plus vastes, dotés de possibilités passionnantes. Mais il va de soi qu'une construction méticuleuse et une mise en service attentionnée occupent toujours le premier plan, indépendamment de la maquette à réaliser.

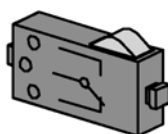
■ Nous vous recommandons de vous familiariser avec l'interface au moyen de quelques expérimentations, avant de passer à la construction des différentes maquettes. Pour ce faire, veillez à la disponibilité du manuel du logiciel ROBO Pro, afin que vous puissiez le consulter immédiatement en cas de problèmes.

Après l'installation du logiciel, vous pouvez raccorder l'interface à votre microordinateur au moyen du câble fourni. Démarrez maintenant le programme ROBO Pro et ouvrez le menu « Test de l'Interface ».



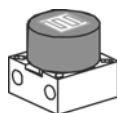
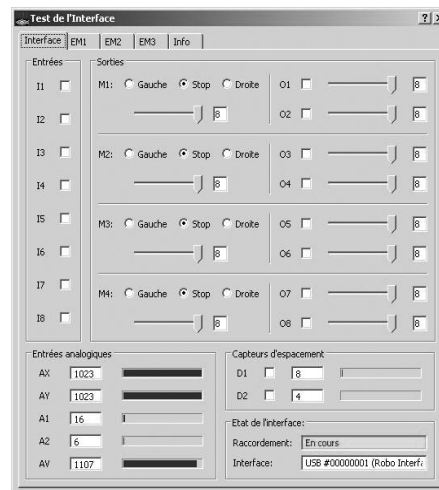
Moteur électrique

Reliez les raccords du moteur électrique au raccord M1. Cliquez sur la sélection « Gauche » ou « Droite » avec la souris. Le moteur démarre à sa vitesse maximale. Servez-vous du régulateur de la barre pour régler la vitesse de rotation. L'opération prend fin dès l'actionnement du bouton « Stop ».



Boutons

Raccordez ensuite un bouton (contact à fermeture) à l'entrée numérique I1 et observez ce qui se passe sur l'écran d'affichage du test si vous actionnez le bouton. Un affichage optique (crochet) signale l'état du bouton. Raccordez le bouton comme contact à ouverture et observez le résultat de cette opération. L'affichage d'état « Commutateur fermé » s'affiche en premier lieu. Cet affichage s'efface dès que vous actionnez le commutateur.



Photorésistance

Raccordez la photorésistance fournie aux raccords AX ou AY.

Modifiez la luminosité de la photorésistance en la couvrant progressivement au moyen d'une bande de papier noire. Que va-t-il se passer ? Vous constatez une modification de la colonne lumineuse bleue et de la valeur de l'entrée utilisée.

Le test de l'interface est très bien expliqué au chapitre 2 du manuel. Le manuel peut aussi vous être utile en cas de difficultés entre votre ordinateur et l'interface et le logiciel – n'hésitez pas à le consulter !

■ Maintenant, c'est parti. Vous êtes familiarisé avec l'interface et la programmation et pouvez vous lancer dans la résolution de la première tâche. Commencez par la construction de la maquette de base en vous servant des instructions de construction.



Tâche 1 – ROBO Pro niveau 1 :

Votre véhicule à chenilles doit se déplacer tout droit durant 6 secondes, virer vers la gauche par la suite après 3 secondes et s'immobiliser.

Et voici les étapes à suivre pour votre première programmation, que nous voulons encore encadrer un petit peu. Cliquez d'abord sur le bouton « Fichier – Nouveau fichier ». Votre programme commence par un bonhomme vert signalant le démarrage du programme.



Ensuite, vous devez disposer de 2 pictogrammes de moteur. Placez le premier pictogramme sous le démarrage du programme de façon à ce que la liaison soit établie automatiquement. Déplacez la souris sur le pictogramme du moteur et mettez l'écran des paramètres en circuit (bouton droit de la souris). C'est ici que vous réglez la sortie moteur sur « M1 » et le sens de rotation d'action sur « Gauche ». Validez par « OK ». Ajoutez le deuxième pictogramme de moteur de la même façon et répétez l'opération pour la sortie moteur « M2 ».

Le déroulement du programme doit patienter durant une certaine période par la suite. Veuillez utiliser le pictogramme du temps d'attente pour ce faire. Vous placez ce pictogramme sous le deuxième moteur et réglez le temps sur 6 secondes.

Ensuite, le véhicule à chenilles doit tourner durant 3 secondes. Pour ce faire, ajoutez à nouveau les deux pictogrammes de moteur pour M1 et M2. M1 doit tourner à gauche et M2 à droite. Ajoutez ensuite le pictogramme du temps d'attente, étant donné que les deux moteurs doivent travailler durant 3 secondes, et placez la valeur sur 3 secondes.

Vous devez arrêter les deux moteurs par la suite. Cette manoeuvre est effectuée, de la même façon que pour la rotation, par l'ajout des deux pictogrammes de moteur et du paramètre « Stop ».

Pour terminer, il ne vous reste plus qu'à ajouter le pictogramme du « bonhomme rouge » signalant la fin du programme. Votre premier programme est fin prêt et vous pouvez l'enregistrer. Testez ensuite le programme en mode Online. Pour ce faire, cliquez sur le bouton « Lancer le programme en mode Online ».

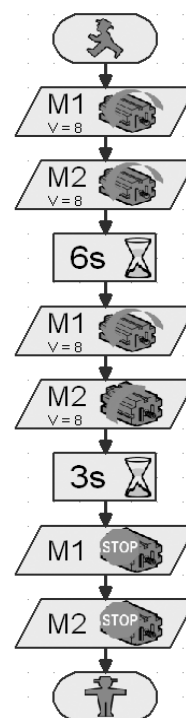
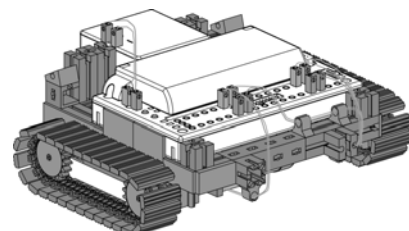
Vous pouvez charger le programme dans l'interface, si tout a été correctement programmé. Pour ce faire, cliquez sur le bouton « Télécharger le programme dans ROBO Interface ». Validez les paramètres de la fenêtre de téléchargement. La maquette se met en marche immédiatement après le téléchargement. Sauf qu'elle est encore reliée au câble USB. Chargez le programme une nouvelle fois, mais activez « Lancer le programme via la touche de l'interface ». Vous pouvez retirer le câble après l'achèvement de la transmission du programme. Appuyez sur la « Touche PROG » de l'interface pour démarrer le programme.

Le programme final est à votre disposition sous :

C:\Programmes\Robopro\Exemples de programmes\Robo_Explorer\Maquette de base_1.rpp

Mais tout ceci était certainement trop simple pour vous. Mais nous tenons à vous familiariser lentement avec le programme. Nous allons donc procéder à une petite extension de la tâche.

La maquette de base

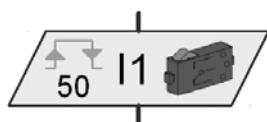
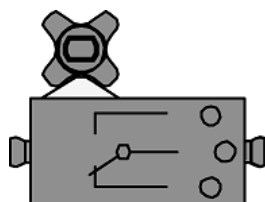


Lancer le programme
en mode Online



Télécharger le programme
dans ROBO Interface

Commande par impulsions



Compteur d'impulsions

■ La maquette de base comporte déjà 2 capteurs (boutons) de mesure du trajet, de même qu'une roue à impulsions reliée avec l'axe du moteur. Cette roue à impulsions actionne le commutateur quatre fois lors de chaque tour. Contrôlez le fonctionnement correct de la maquette au moyen du « Test de l'interface ».

Tâche 2 – ROBO PRO niveau 1 :

Votre véhicule à chenilles doit avancer de 50 impulsions. Ensuite, il devrait tourner vers la gauche durant 11 impulsions avant de s'immobiliser. Des Informations utiles quant au compteur d'impulsions figurent au chapitre 3.6.3 du manuel.



Solution

Mettez d'abord les deux moteurs en circuit. Ajoutez ensuite le compteur d'impulsions. Modifiez les paramètres pour un nombre d'impulsions de 50. Ajoutez encore 2 pictogrammes de moteur. M1 continue de tourner vers la « Gauche », tandis que vous commutez M2 sur « Droite ». Cette orientation devrait durer 11 impulsions. Pour terminer, vous devez à nouveau ajouter 2 pictogrammes de moteur et leur appliquer un « Stop ». Transmettez le programme et testez-le sur votre véhicule.

Vous obtenez huit au lieu de quatre impulsions par tour de la roue à impulsions, si vous réglez le programme sur le type d'impulsion « 0 -> 1 (croissant) » ou « 1 -> 0 (décroissant) » dans l'écran des paramètres du compteur d'impulsions. Ceci signifie que le nombre d'impulsions par trajet parcouru est doublé et que la précision de mesure du trajet s'accroît.

Le programme final est à votre disposition sous :

C:\Programmes\Robopro\Exemples de programmes\Robo_Explorer\Maquette de base_2.rpp

■ Vous avez certainement remarqué que chaque élaboration d'un programme pose pour condition de placer les moteurs lors de chaque nouvelle orientation ou d'un arrêt. Ceci a souvent pour effet d'accroître rapidement la confusion au sein de programmes plus volumineux et de transformer la recherche d'éventuels défauts en véritable jeu de patience.

ROBO PRO vous propose de faire appel à une solution plus élégante au moyen de « sous-programmes ». Veuillez lire le chapitre 4.1 du manuel dans ce contexte. Il est cependant important de commuter ROBO PRO sur le **niveau 2**.

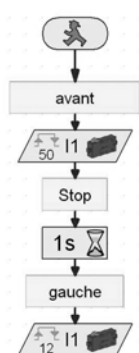
Tâche 3 – ROBO PRO niveau 2 :

Le véhicule à chenilles doit se déplacer sur les lignes d'un carré. Utilisez les paramètres 50 et 11 comme pour la première tâche. Elaborez un sous-programme pour chaque sens de marche et pour chaque arrêt.



Elaborez d'abord le sous-programme « En avant » (voir chapitre 4 du manuel ROBO PRO). Marquez les éléments du programme et copiez-les dans la mémoire temporaire. Ensuite, vous élaborez les sous-programmes « Gauche » et « Stop ». Intégrez les éléments du programme « En avant » de la mémoire intermédiaire dans le programme actuel et modifiez les paramètres en conséquence. Servez-vous d'une vitesse de rotation réduite pour l'orientation. Nous avons illustré un extrait de la tâche, afin de vous donner une meilleure idée des opérations. Le tableau ci-après est destiné à vous donner une vue d'ensemble de la programmation des moteurs en fonction du sens de marche.

Solution



Sous-programmes



Créer un nouveau sous-programme



Copier le sous-programme actuel



Effacer le sous-programme actuel

Sens de marche	Sens de rotation moteur 1	Sens de rotation moteur 2
En avant	Gauche	Gauche
En arrière	Droite	Droite
Gauche	Gauche	Droite
Droite	Droite	Gauche
Stop	Stop	Stop

Tous les moteurs des programmes de démonstration ont été programmés au moyen de ce tableau.

Programme final :

C:\Programmes\Robopro\Exemples de programmes\Robo_Explorer\Maquette de base_3.rpp

Tâche 4 – ROBO PRO niveau 3 :

Le robot ne suit pas les lignes du carré avec précision. Posez-vous les questions suivantes et vérifiez : Quelle est la cause de l'irrégularité ? Que puis-je faire pour améliorer le résultat ?



Astuce :

Synchronisez les moteurs M1 et M2 au moyen des boutons I1 et I2 de façon à ce que la maquette se déplace exactement tout droit. Nous vous avons fourni le sous-programme fin prêt « sync_droit » pour se faire. Il suffit d'entrer le nombre d'impulsions comme constante au niveau de l'entrée du sous-programme de couleur orange (voir maquette de base_4.rpp). Vous pouvez procéder de même dans le sous-programme « sync_orientation » et entrer le nombre d'impulsions de l'angle de rotation.

Programme final :

C:\Programmes\Robopro\Exemples de programmes\Robo_Explorer\Maquette de base_4.rpp

■ Vous venez de vous familiariser avec le fonctionnement de la maquette de base et votre robot est donc prêt pour réagir à différents signaux venus de l'extérieur.

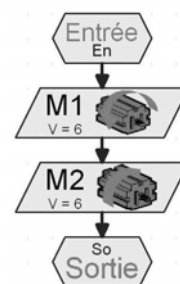
Vous devez équiper votre véhicule à chenilles de capteurs, si vous voulez qu'il identifie son environnement et exécute certaines tâches. Les suggestions de maquettes ci-après vous montrent les différentes variantes de véhicules à chenilles équipés de différents capteurs. Ils doivent notamment identifier différents trajets, la lumière ou les couleurs, mais également des sources de chaleur ou des espacements. Les différents programmes ont été consignés au répertoire :

C:\Programmes\Robopro\Exemples de programmes\Robo_Explorer\

Les halls industriels déserts où les véhicules de transport se déplacent comme par magie font toujours de l'effet dans les réalisations télévisées. Ces systèmes sont partiellement commandés par des lignes de transmission de données intégrées au sol ou des repères tracés sur le sol.

Votre propre robot devrait se déplacer le long d'une ligne noire, s'il a été correctement programmé.

Vous devez d'abord monter le dépisteur aux termes des instructions de construction avant de démarrer la programmation. Un parcours expérimental avec l'impression de la ligne noire est compris dans la boîte de construction. La ligne à longer par le dépisteur devrait être une droite pour commencer.



Sous-programme « Gauche »

Véhicules sur chenilles autonomes

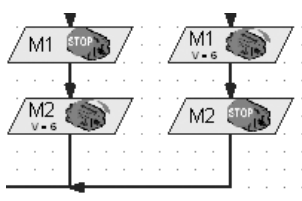
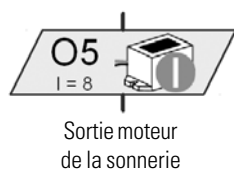
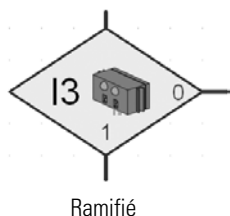
Dépisteur





■ Comment la maquette devra-t-elle fonctionner ?

Le robot doit trouver une ligne noire sur un fond blanc et longer cette ligne. Pour ce faire, vous avez monté le dépisteur IR dans votre maquette. L'élément de construction émet une lumière infrarouge sur la voie. Cette lumière est réfléchiée selon le revêtement respectif de la voie et mesurée par des phototransistors. Répercussions sur votre programmation : les revêtements clairs/blancs réfléchissent la lumière et vous obtenez la valeur 1. Un revêtement noir ne réfléchit pas la lumière, ce qui donne la valeur 0. Votre robot a trouvé la voie (ligne noire) et peut la suivre du moment que les deux transistors signalent la valeur 0.



Tâche 1 – ROBO PRO niveau 2 :

Votre véhicule à chenilles doit être placé sur une piste noire droite et la longer.

Un signal acoustique d'une durée d'une seconde doit retentir dès qu'il dévie de la piste ou que la fin de la piste est atteinte.

Veuillez diviser le programme en un programme principal d'interrogation des capteurs et un sous-programme pour le déplacement tout droit, le signal de la sonnerie et l'arrêt.



Encore quelques petites astuces :

Vérifiez la détection de la piste du capteur au moyen du test de l'interface. Une détection erronée du noir et blanc pourrait trouver son origine dans des sources lumineuses perturbatrices (p. ex. le soleil). Il se pourrait qu'il soit requis de positionner le capteur plus près de la piste ou de le protéger au moyen d'un panneau.

Programme final : **Dépisteur_1.rpp**

■ Nous sommes convaincus que votre première solution ne vous donnera pas satisfaction – bien qu'il soit évident que le robot longe la piste sur un petit trajet. Sauf que vous ne pouvez pas le réajuster, qu'il quittera la piste marquée avant de s'immobiliser et de vous signaler son arrêt.

Tâche 2 – ROBO PRO niveau 2 :

Vous devez étendre les ramifications d'interrogation du phototransistor du programme principal, de façon à ce que le robot puisse détecter qu'il ne suit plus la piste avec exactitude. Ensuite, il devrait corriger son sens de marche en conséquence. Une petite astuce à ce sujet vous est donnée dans l'extrait du programme à gauche.

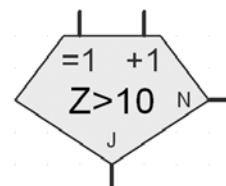


C'est déjà nettement mieux. Votre robot longe la piste voulue avec précision. D'autres robots assumeront le chargement à transporter en fin de piste ou rechargeront le robot, si nous reprenons l'exemple d'un hall industriel. La charge nouvellement assumée pourra être transportée en retour jusqu'au point de départ.

Programme final : **Dépisteur_2.rpp**

Tâche 3 – ROBO PRO niveau 2 :

Votre robot s'est déplacé le long d'une certaine ligne noire dans le cadre des tâches précédentes. La nouvelle tâche concerne la recherche d'une ligne. Pour ce faire, il devra d'abord tourner en rond. S'il ne trouve pas de piste, il devra avancer un petit peu et se mettre à nouveau à la recherche. Votre robot devra suivre la piste dès qu'il l'aura trouvée. Il doit redémarrer la recherche dès que la piste a pris fin ou s'il la perd.



Boucle de comptage

Astuce :

Rappelez-vous la première tâche de la maquette de base. Le robot devait tourner de 90 degrés dans ce cas. Vous avez résolu ce problème par une commande des voies à parcourir. Cette technique peut aussi présenter certains avantages dans le cas objet de la présente tâche.

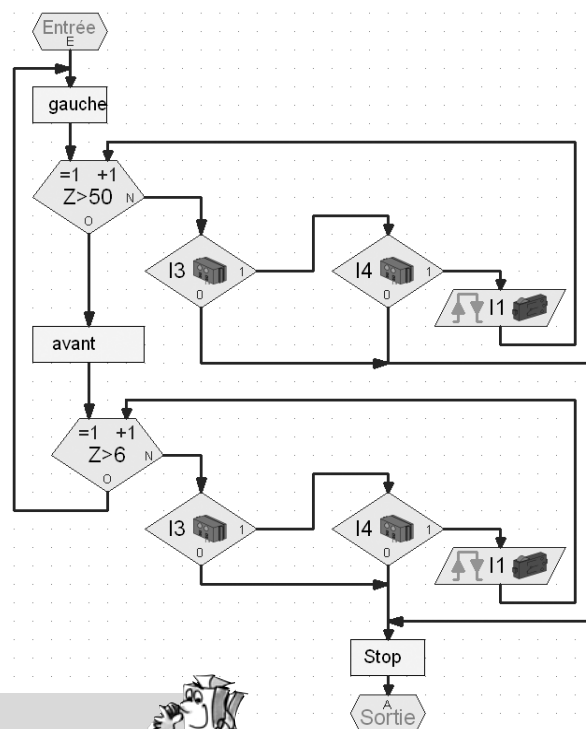
Vous devez créer un propre sous-programme appelé « Recherche de trace » pour la recherche de la piste. La figure est uniquement destinée à vous donner une idée de sa réalisation.

Ecrivez encore un sous-programme avec les dénominations à gauche. Les commandes « En avant » et « Stop » ont déjà été créées.

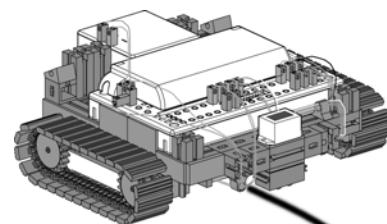
Le programme principal et tous les sous-programmes du dépisteur sont prêts. Vous avez maintenant pour mission de les intégrer dans le programme de façon à ce qu'ils puissent exécuter la tâche 3.

Programme final : **Dépisteur_3.rpp**

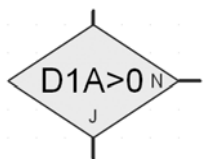
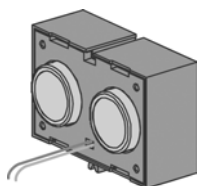
Les parcours à suivre jusqu'à présent étaient toujours composés d'une droite. Mais il se produit aussi que les parcours aient des virages. Ceci est notamment le cas dans des installations industrielles affectées au transport des marchandises ou pièces à usiner de machine en machine.

**Tâche 4 – ROBO PRO niveau 2 :**

Le parcours expérimental comprend différents virages dotés de rayons divergents. Pensez également à utiliser différentes vitesses de M1 et M2 dans vos expérimentations sur un parcours circulaire. Quels sont les paramètres permettant au robot de suivre le parcours le plus rapidement ? Prenez note des résultats sur un petit tableau.



Robot pour tunnel, capteur d'espacement et de température



■ Le capteur de mesure de l'espacement et la résistance NTC vous proposent des possibilités d'extension de votre maquette pour le transformer en robot professionnel.

Mesurer les espacements et les températures et engager des mesures auxiliaires si besoin est. Où pensez-vous que de tels robots peuvent faire preuve de leurs capacités ? Nous sommes certains que vous connaissez de nombreux domaines d'activités dans ce contexte. Notre exemple tient du domaine de la protection et de la lutte contre les incendies dans des tunnels automobiles et ferroviaires.



La mission d'un robot consiste à s'approcher du foyer de l'incendie, à mesurer les températures dans le tunnel et à signaler les données respectives à une centrale. Dans la plupart des cas, les robots sont équipés d'un équipement d'extinction mobile à utiliser en fonction des conditions existantes sur site.

Veuillez suivre les instructions de construction de la maquette « Robots pour tunnels » scrupuleusement. Le capteur d'espacement est à raccorder aux raccords D1.

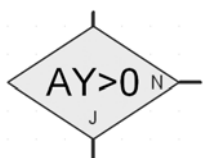
Tâche 1 : ROBO PRO niveau 2

La mission ressemble à celle du dépisteur longeant une ligne, sauf que le nouveau robot devra parcourir une certaine distance le long d'un mur en respectant un espacement défini (environ 20 cm).



Programme final : **Tunnel_1.rpp**

■ Mais revenons d'abord à un robot d'extinction des incendies pour la prochaine tâche. Il utilise des capteurs d'espacement comparables à ceux de votre propre robot pour longer le mur. Sauf qu'il se sert de capteurs thermiques pour détecter le foyer de l'incendie. Le capteur thermique correspond à la résistance NTC par rapport à votre maquette. La caractéristique physique de cet élément de construction est que la valeur de résistance baisse dès que la température s'accroît. Vous pouvez aussi tester cette modification dans le cadre du test de l'interface. Raccordez la résistance NTC au raccord AY. Tenez une source de chaleur devant la résistance NTC et observez la barre bleue de l'AY.



Tâche 2 : ROBO PRO niveau 2

Étendez le programme en demandant au robot de longer le mur du tunnel. Mesurez également la chaleur actuelle au moyen du raccord AY. Votre robot devrait s'arrêter et émettre un signal d'avertissement au moyen de la sonnerie dès le dépassement d'une température définie. Une lumière d'avertissement rouge devrait clignoter en même temps que la sonnerie.

Votre robot devrait faire demi-tour après cette extinction simulée et revenir au point de départ.



Programme final : **Tunnel_2.rpp**

Astuce :

Votre robot est équipé d'un seul capteur d'espacement et il doit donc disposer d'un deuxième mur à longer pour revenir au point de départ.

Vous pouvez intégrer l'extinction du feu dans votre programme, si vous disposez d'un moteur et d'une hélice supplémentaires dans votre collection fischertechnik.

■ Le prochain capteur à découvrir est le capteur chromatique. La figure ci-contre veut servir d'exemple pour un emploi industriel. Vous voyez que les flacons de peinture mal alignés sont rejetés.

La lumière réfléchiée par la marchandise à sonder est reçue, numérisée et interprétée au moyen d'un ordinateur et d'un logiciel. Le capteur a pour mission d'identifier les différentes couleurs et d'envoyer les données mesurées à l'interface.

Le capteur chromatique est intégré à la maquette de l'identificateur de couleurs. Son conducteur noir est raccordé à A1, son conducteur rouge à + et son conducteur vert à -. Veuillez utiliser les surfaces colorées imprimées sur le parcours pour le premier test du programme.

Tâche 1 – ROBO PRO niveau 2 :

Veuillez contrôler les valeurs sorties par l'interface pour les différentes couleurs dans le test de l'interface. Utilisez aussi le noir et le blanc en dehors des 3 coloris déterminés.

Elaborez un petit tableau et prenez note des valeurs mesurées. N'oubliez pas d'observer les modifications dues à la distance par rapport à la surface colorée et la lumière environnante.

**Tâche 2 – ROBO PRO niveau 2 :**

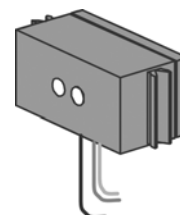
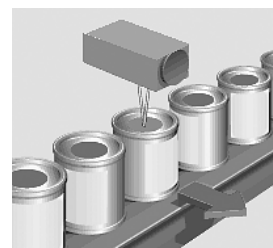
Créez un petit programme permettant au capteur de détecter la surface verte. La sonnerie est activée durant une seconde si la valeur mesurée se situe dans la plage de valeurs définie. Le programme bascule dans le démarrage par la suite.

Programme final : **Identificateur de couleurs_2.rpp**

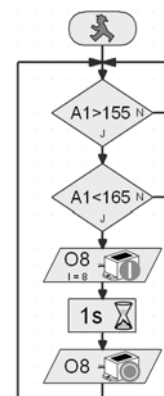
Astuce :

Servez-vous des trois lampes avec les capuchons lumineux de différentes couleurs, déjà montées dans la maquette, pour la mise en oeuvre de la prochaine tâche.

Identificateur de couleurs



Couleur	Valeur
Blanc	
Noir	
Bleu	
Rouge	
Vert	





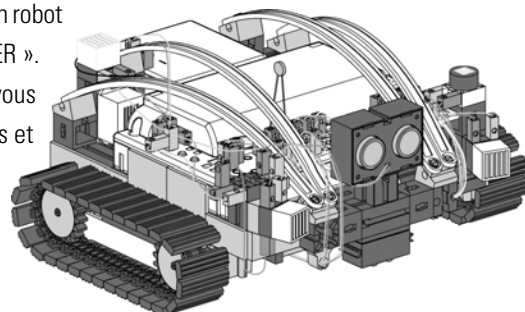
Tâche 3 – ROBO PRO niveau 3 :

Ecrivez un programme permettant à votre robot de se déplacer tout droit en suivant un parcours défini. Ce parcours sera doté de trois surfaces de couleurs. Le robot devrait s'immobiliser durant 3 secondes dès que le capteur a identifié une couleur. Durant cette période, le robot devra commuter les lampes correspondantes à la couleur identifiée et déclencher un signal acoustique au moyen de la sonnerie. Il se déplacera sur la prochaine surface et redémarrera son travail. Pour terminer, il se rendra sur la dernière surface, signalera le résultat de son travail et s'immobilisera.

Programme final : **Identificateur de couleurs_3.rpp**

Maquette générale Explorer

■ Tous les actionneurs et capteurs nécessaires pour un robot autonome sont compris dans la maquette « EXPLORER ». Vous disposez donc du matériel, qu'il vous faut pour vous lancer dans la solution sans limites de tâches simples et compliquées. Les niveaux de construction précédents étaient restreints à l'emploi d'un seul capteur pour se familiariser avec les possibilités d'utilisation.



Tâche 1 – ROBO PRO niveau 2 :

Programmez votre robot de façon à ce qu'il se déplace vers un obstacle existant. Il devrait réduire sa vitesse d'approche de 50 % dès l'atteinte d'un espacement d'environ 60 cm. Il devra s'immobiliser à une distance de 40 cm. Pour autant que l'obstacle approche de votre robot, ce dernier devra ralentir à partir d'une distance de 20 cm et reculer rapidement à partir d'une distance de l'obstacle de 10 cm.

Programme final : **Explorer_1.rpp**

Tâche 2 – ROBO PRO niveau 2 :

Votre robot part à la découverte de son environnement. Elaborez un programme pour l'utilisation de 2 capteurs – le dépisteur et le capteur d'espacement. Le robot devra d'abord suivre une ligne noire sur un parcours expérimental. Veuillez placer un obstacle sur ce parcours. Le robot devrait s'arrêter environ 10 cm devant l'obstacle et reculer d'un centimètre. Ensuite, il devra faire demi-tour et suivre la piste dans le sens inverse.



Programme final : **Explorer_2.rpp**

Tâche 3 – ROBO PRO niveau 2 :

Extension sur trois capteurs pour le programme de la tâche 2 – identificateur de couleurs, capteur de température et photorésistance de mesure de la luminosité.

Différentes surfaces de couleurs se situent le long de la piste. Le robot les signale par des signaux acoustiques distincts. La lampe d'avertissement rouge devrait clignoter si la température ambiante est trop élevée en cours de marche. Votre robot met ses 2 phares en circuit dès que la pièce s'assombrit. Les phares se débranchent dès la luminosité est à nouveau suffisante.



Programme final : **Explorer_3.rpp**

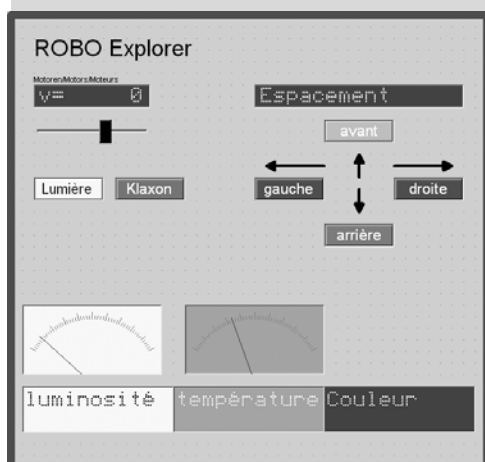
■ En 2003, la planète Mars était si proche de la Terre que rarement auparavant. Les Européens et Américains ont profité de cette occasion pour y expédier leurs propres véhicules d'exploration. La mission : trouver des traces de vie.

Notre Explorer est un robot télécommandé que vous pouvez programmer pour partir à la découverte d'univers inconnus. Vous devez cependant disposer de l'équipement supplémentaire RF Data Link.

Mars – en route à la découverte de la quatrième planète

**Tâche 4 – ROBO PRO niveau 3 :**

Votre ordinateur assume le rôle de la centrale de commande de votre expédition sur Mars. La mission consiste à transmettre les valeurs mesurées d'un paysage de la planète Mars à la station au sol. Dans ROBO Pro, la commande de votre robot est réalisée au niveau du panneau de commande (voir chapitre 8 du manuel).



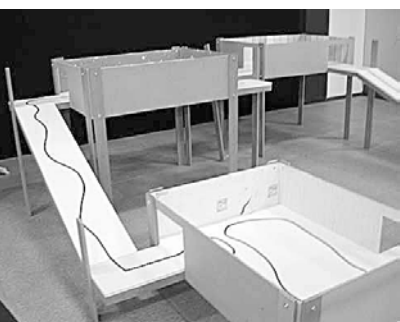
Votre robot devrait être programmée de manière à ce qu'il puisse transmettre les valeurs mesurées propres à la couleur du sol, à la température, à la luminosité et aux obstacles. La commande du robot s'opère manuellement au moyen du panneau de commande du programme principal dans ROBO Pro.

Programme final : **Explorer_4.rpp**



RoboCup Junior

Robot de sauvetage



Vous venez d'achever toutes les tâches de familiarisation avec la construction des robots et la technique de programmation – et le moment est certainement venu de récolter les fruits de votre travail et de vous présenter avec la maquette du « robot de sauvetage » dans la ligue de sauvetage du RoboCupJunior.

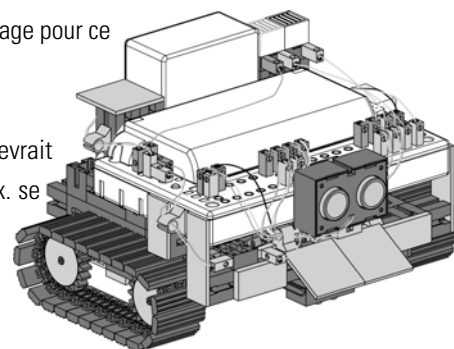
■ RoboCupJunior est une initiative mondiale d'enseignement orientée à des projets et consacrée à la promotion de manifestations régionales, nationales et internationales de robotique pour jeunes gens. RoboCupJunior a pour objectif de présenter l'univers des robots et leur utilisation aux enfants et adolescents.

Les tâches expliquées sur Internet sous

www.robocupjunior.org

vous permettront certainement de programmer votre robot de sauvetage pour ce concours.

La photo à gauche montre l'exemple d'un parcours que votre robot devrait assumer. Il doit accomplir différentes missions sur ce parcours, p. ex. se déplacer le long d'une ligne, chercher différentes figures sur le sol, traverser une porte etc. Est-ce que cela vous tenterait ?



Tuyaux et astuces

■ Le plaisir de la robotique chute rapidement si les robots ne fonctionnent pas comme souhaité.

Bien qu'il soit parfois facile de détecter les petits défauts et d'y remédier.

Câble

Procédez avec un maximum de précision. Les câbles doivent être coupés à la longueur prescrite, avant de dénuder leurs extrémités et de les relier fermement au moyen de connecteurs. Contrôlez le fonctionnement correct au moyen de la pierre lumineuse (38216) avec la lampe sphérique à emboîter (37869) et l'Accu Set.

Alimentation en courant électrique

Un accumulateur presque vide est souvent à l'origine d'un dysfonctionnement de votre ROBO EXPLORER. L'interface signale une chute de la tension sous 5 Volts par une DEL rouge. Simultanément, l'interface se débranche automatiquement. Un dysfonctionnement peut aussi se présenter bien que l'accu ne soit pas encore épuisé. Une recharge de l'accu s'impose également dans un tel cas.

Programmation

Une programmation erronée est souvent à l'origine d'un dysfonctionnement, si le robot ne fonctionne pas correctement bien que tous les problèmes mécaniques aient été résolus. ROBO PRO vous propose de passer par le mode Online, qui permet de suivre le déroulement du programme à l'écran. Vous pourrez y découvrir tous les petits défauts éventuellement intervenus en cours de programmation.



Inhoud



Voertuig met rupsaandrijving	P.50
De besturing	P.50
Onderzoeksmodel van fischertechnik	P.51
Schakelaars	P.51
Sensoren	P.51
ROBO Interface	P.53
Voeding	P.53
Software ROBO Pro	P.53
Interface-uitbreidingen	P.53
Toelichting	P.54
Uitgangspunten m.b.t. de interface	P.54
Het basismodel	P.55
Basisprogramma	P.55
Impulsbesturing	P.56
Subprogramma's	P.56
Autonoom rupsvoertuig	P.57
Spoorzoeker	P.57
Tunnelrobot	P.60
Kleurherkenning	P.61
Explorer	P.62
RoboCupJunior – Rescue robot	P.64
Belangrijke tips	P.64

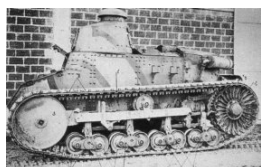
Voertuig met rupsaandrijving

■ Waarvoor heeft men een autonoom voertuig met rupsaandrijving nodig? De uitvinding van een rupsaandrijving voor voertuigen was nodig om ook op onbegaanbaar terrein te kunnen rijden. Als gewone banden niet meer werken, bijv. in een woestijn, kan een rupsaandrijving worden gebruikt. Zo werden in de 1e wereldoorlog de eerste rupsaangedreven vrachtwagens en pantservoertuigen gebouwd en ingezet.



Afhankelijk van het terrein kon men de voertuigen met banden ombouwen tot voertuigen met rupsaandrijving.

Ook door gewone burgers werden soms rupsvoertuigen gebruikt. Zoals je op de foto's kunt zien, waren de wielaangedreven voertuigen vrijwel altijd uitgangspunt voor de rupsvoertuigen.



Maar al snel constateerde men een zwak punt: de bestuurbare voorwielen. Dus ging men ertoe over, de rupsaandrijving naar alle assen uit te breiden.

De besturing



■ Maar hoe werkte dan de besturing? Heel eenvoudig, door één van beide rupsen langzamer of sneller te laten draaien. Als men rechtsaf wilde slaan, vertraagde men met behulp van een stuurknuppel (voor elke rups één) de rechter rups. Daardoor draaide deze rups langzamer en dus werd het voertuig naar rechts gedraaid.

Ook nu, natuurlijk wel aangepast aan de huidige technieken, zijn er nog veel rupsaangedreven voertuigen. Van kleine baggervoertuigen tot ware kolossen in de bruinkoolindustrie.



In de Cheops-pyramide in Egypte wilde men met behulp van een minirobot nog meer geheimen ontdekken.

De onderzoekers stuurden robots ter grootte van een speelgoedlocomotief door een donkere, smalle tunnel. Hij reed naar een kamer midden in de 4.500 jaar oude pyramide en eindigde voor een geheimzinnige stenen deur.



■ Onbekende ruimten ontdekken, afstanden meten, sporen volgen, rijrichtingen door middel knippersignalen aangeven, kleuren herkennen, temperatuur meten, zonder contact hindernissen ontwijken, onderscheid kunnen maken tussen dag- en nacht, schijnwerpers automatisch in- en uitschakelen, een alarm activeren, enz. Dat alles – en nog veel meer – maken de sensoren van de ROBO Explorer mogelijk. Met name zijn dat: de NTC-weerstand, de fotoweerstand, de nieuwe ultrasoon-afstandssensor, de infrarood kleursensor alsmede de speciaal ontwikkelde infrarood-sporensensor. Dankzij twee Powermotoren en de rupsaandrijving kan ook onherbergzaam terrein onderzocht en bereden worden. Met behulp van het meegeleverde model Rescue Robot is de bouwdoos een ideale basis voor deelname aan de de RoboCup-Junior.

Voordat je echt begint, moet je eerst de belangrijkste onderdelen kennen. Deze worden hierna beschreven:

Powermotor

Om de rupsen en dus jouw model aan te drijven en te besturen, heb je 2 powermotoren nodig. Deze worden op de interface van de uitgangen M1 tot M4 aangesloten.

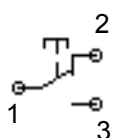
Kogellamp

Dit betreft een gloeilamp voor een spanning van 9 V. Hij kan als meldsignaal voor de rijrichting of gewoon als verlichting worden gebruikt. De lamp wordt net als de powermotor op de interface-uitgangen M1 tot M4 aangesloten.

Zoemer

De zoemer is bedoeld om bijv. hindernissen of kleuren te herkennen en dit akoestisch te melden. Ook deze wordt aangesloten op de interface-uitgangen M1 tot M4.

Voeler



Voelers behoren tot de aanraaksensoren. Druk je op de rode knop, wordt in de behuizing mechanisch een contact geschakeld en er vloeit een stroom tussen de contacten 1 en 3. Gelijktijdig wordt het schakeltraject tussen de aansluitpunten 1 en 2 onderbroken.

Voelers of schakelaars worden op twee verschillende manieren gebruikt:

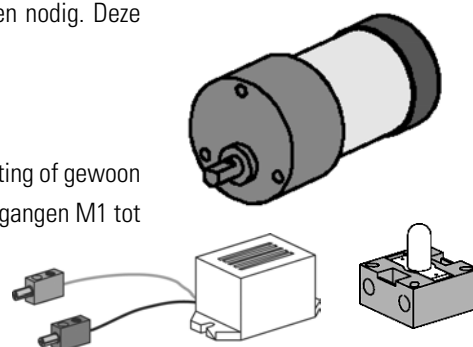
Voeler als „sluiter“

De beide schakelbeelden laten de opbouwopgave zien. Een spanningsbron (9V) wordt op contact 1 van de voeler, een lamp op contact 3 van de voeler en op de min-pool van de spanningsbron aangesloten.

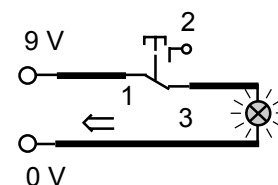
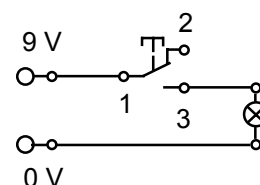
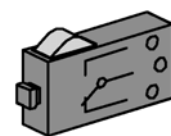
Als de voeler niet is ingedrukt, is de lamp uit. Druk je op de voeler, dan wordt het stroomcircuit via contact 1 en contact 3 gesloten en de lamp brandt.

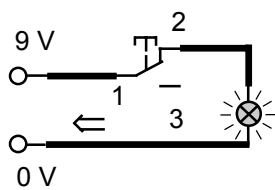
Onderzoeksmodel van fischertechnik

Schakelaars



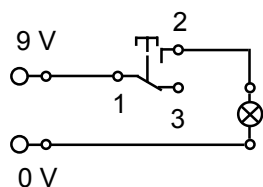
Sensoren





Voeler als „opener“

Ook hier kun je de werking op de beide schakelbeelden zien. Contact 1 wordt met de spanningsbron verbonden. Contact 2 met de lamp en de lamp weer met de min-pool van de spanningsbron.



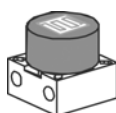
De lamp brandt. Druk je vervolgens op de schakelaar, dan wordt de stroom onderbroken en de lamp gaat uit.

Bij je eigen modellen gebruik je de voeler samen met het impuls wiel als telvoeler.



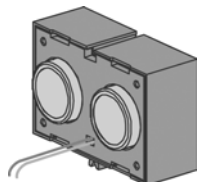
NTC-weerstand (Negatieve Temperatuur Coëfficiënt)

Het betreft hier een onderdeel, waarmee je verschillende temperaturen kunt meten. Dit wordt ook wel een warmtesensor genoemd. Bij ca. 20 graden heeft de NTC-weerstand een waarde van 1,5 kOhm. Stijgt de temperatuur, dan neemt de weerstandswaarde af. Deze informatie kan via de interface aan de analoge ingangen AX en AY worden ingelezen en is bij de ROBO Pro als getal tussen 0 - 1.023 beschikbaar.



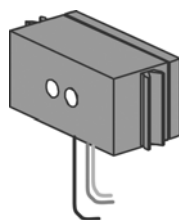
Fotoweerstand

De LDR 03, een analoge helderheidssensor voor de ingangen AX en AY (weerstandsmeting), reageert op daglicht en verandert daarbij van weerstandswaarde. Dit is een indicator voor de helderheid van het licht.



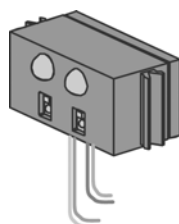
Ultrasoon afstandssensor

Een afstandssensor is een technisch onderdeel, dat in staat is, de afstand tussen zichzelf en een voorwerp te meten. Afstandssensoren werken met licht, infrarood stralen, radiogolven, of **ultrasone geluidsgolven** en worden voor verschillende meetmethoden gebruikt. Geluid breidt zich als een golf uit. Een echo wordt naar de bron van het ultrasone geluid gereflecteerd en daar als signaal opgevangen en geanalyseerd. Het tijdsverschil tussen het uitzenden en ontvangen van het signaal geeft informatie over de afstand tussen de hindernis en de sensor. Alle aansluitingen zijn als D1/D2 op de interface aanwezig. De reikwijdte van de sensor bedraagt max. 4 m. De uitgegeven getalswaarde komt overeen met de afstand in centimeters.



Optische kleursensor

Kleursensoren worden meestal in de automatiseringstechniek gebruikt. Daarbij moet bijv. de kleur of een kleurenafdruk gecontroleerd worden, om er zeker van te zijn dat het juiste onderdeel werd ingebouwd. De kleursensor van fischertechnik zendt rood licht uit, dat door de verschillende kleurvlakken niet altijd even sterk gereflecteerd wordt. De hoeveelheid gereflecteerd licht wordt via de fototransistor gemeten en als spanningswaarde tussen 0 V en 10 V uitgegeven. De gemeten waarde is afhankelijk van de helderheid van de omgeving en van de afstand tussen sensor en kleurvlak. Deze waarde kun je via de ingangen A1 en A2 inlezen en als getalswaarde tussen 0 - 1.000 in je eigen programma verwerken.



Sporensensor

De IR-sporensensor is een digitale infrarood sensor voor de herkenning van een zwart spoor op een witte ondergrond op een afstand van 5 - 30 mm. Hij bestaat uit twee zend- en twee ontvanger-elementen. Voor de aansluiting heb je twee digitale ingangen en een 9 V-voedingsspanning (plus en min) nodig.

■ Het belangrijkste onderdeel om een rupsvoertuig in elkaar te zetten, is de ROBO Interface, dat in vrijwel alle verschillen modellen wordt ingebouwd. Hierop sluit je naar behoefte je sensoren en schakelaars aan. De basisbedrading vindt je in de bijgeleverde montagehandleiding.

■ Bij de ROBO Explorer-modellen gaat het om autonome voertuigen, die zich vrij in de ruimte bewegen. Voor de voeding kunt je daarom het beste de accuset van fischertechnik gebruiken.

■ ROBO Pro is een eenvoudig grafische programmeeropervlak waarmee je zelf je eigen programma's kunt schrijven. Het voordeel is, dat je geen programmeertaal hoeft te leren. Eigenlijk kun je gewoon direct beginnen.

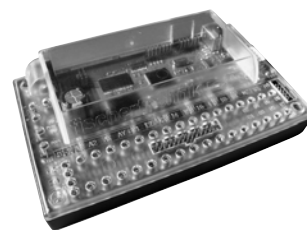
Voor het bouw pakket ROBO-Explorer heb je de ROBO Pro versie 1.2.1.30 nodig. Als je over een oudere softwareversie beschikt, kun je die gratis updaten. Je kunt of via het help-menu in ROBO Pro – Nieuwe versie downloaden of dit onder

www.fischertechnik.de/robopro/update.html doen.

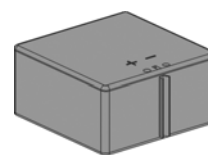
ROBO RF Data Link

De draadloze interface ROBO RF Data Link vervangt de interfacekabel tussen de PC en de interface. Door middel van een radiografische overdracht worden de gegevens dan aan de interface doorgegeven. Het voordeel is, dat je geen lastige kabels meer nodig hebt. Een ander voordeel is, dat je programma's draadloos in de online-modus kunt laten draaien. Zo kunnen fouten gemakkelijker worden opgespoord dan in de Download-modus. En deze methode heeft nog een voordeel – mobiele robots kunnen online via het beeldscherm worden aangestuurd en kunnen meetwaarden draadloos doorgeven aan de computer. Als meerdere robots met een Data Link zijn uitgerust, kunnen ze zelfs onderling gegevens uitwisselen, zoiets als met elkaar „praten”.

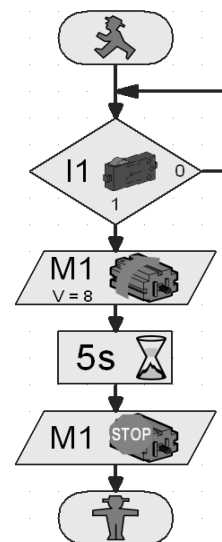
ROBO Interface



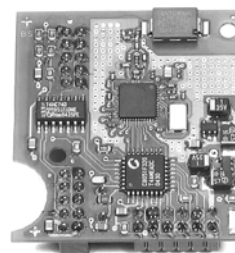
Voeding



Software ROBO Pro



Interface-uitbreidingen



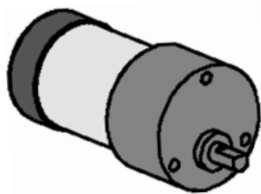
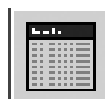
Toelichting

■ Net als bij alle andere fischertechnik-robots wordt je bij de ROBO Explorer stap voor stap door de fascinerende techniek en de programmering geleid. Je begint met een eenvoudig model en gaat daarna verder met steeds ingewikkelder systemen met ongekennde mogelijkheden. Bij alle modellen is echter een zorgvuldige opbouw en zorgvuldige ingebruikname het allerbelangrijkste.

Uitgangspunten met betrekking tot de interface

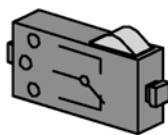
■ Voordat je aan de verschillende modellen begint, moet je je eerst met behulp van een paar experimenten vertrouwd maken met de interface. Leg daarvoor het handboek van de software ROBO Pro klaar, zodat je dit bij problemen kunt raadplegen.

Nadat je de software hebt geïnstalleerd, kun je de interface met de bijgeleverde kabel op je PC aansluiten. Start dan het programma ROBO Pro en open „Interface testen“.



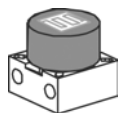
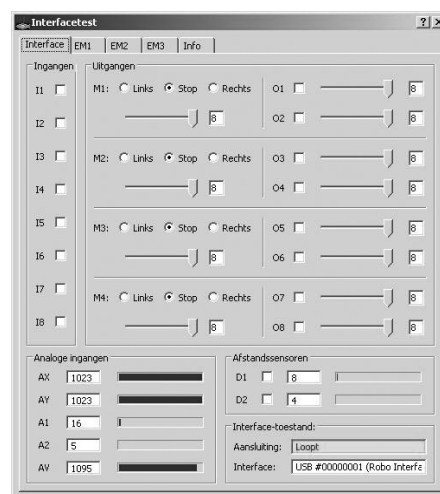
Powermotor

Verbind de aansluitingen van de powermotor met aansluiting M1. Klik met de muis op het mogelijkheid „Links“ of „Rechts“. De motor start met zijn maximale snelheid op. Door het bedienen van de regelaar kun je de omwentelingssnelheid instellen. Met „Stop“ wordt het proces beëindigd.



Voeler

Sluit vervolgens een voeler (sluiter) op de digitale ingang I1 aan en kijk wat er op het testdisplay gebeurt wanneer je de toets indrukt. Een optische aanduiding (haakje) geeft de toestand van de voeler aan. Sluit de voeler als opener aan en bekijk het resultaat. Eerst verschijnt de toestandsaanduiding voor „Schakelaar gesloten“. Druk je op de schakelaar, dan gaat de aanduiding uit.



Fotoweerstand

Sluit de bijgeleverde fotoweerstand aan op de aansluiting AX of AY.

Verander de lichtsterkte van de fotoweerstand door deze langzaam af te dekken met een zwart strookje papier. Wat gebeurt er? Je zult zien dat de blauwe lichtbalk verandert evenals de getalswaarde van de gebruikte ingang.

De test van de interface wordt uitgebreid beschreven in het handboek in hoofdstuk 2. Ook wordt er hulp geboden, wanneer er problemen tussen jouw computer en de interface en de software ontstaat – het is de moeite waard om ook dan in het handboek te kijken!

■ Nu begint het pas echt. Nadat je vertrouwd bent geraakt met de interface en programmering, kun je de eerste taak uitvoeren. Eerst wordt het basismodel aan de hand van de montagehandleiding opgebouwd.



Taak 1 - ROBO Pro Niveau 1:

Het rupsvoertuig moet 6 seconden rechtuit rijden, daarna moet en 3 seconden lang naar links draaien en dan stil blijven staan.

Bij jouw eerste programma zullen wij je nog een beetje helpen. Klik als eerste op de knop „Bestand-Nieuw”. Het programma begint met een groen verkeerslichtpoppetje om het programma te starten.



Vervolgens heb je 2 motorsymbolen nodig. Zet het eerste symbool zo onder de programmastart, dat automatisch verbinding wordt gemaakt. Zet de muis op het motorsymbool en schakel het eigenschappenvenster in (rechter muisknop). Daar zet je de motoruitgang op „M1” en onder Actie de draairichting op „Links”. Dan druk je op OK. Voeg het tweede motorsymbool op dezelfde manier in en herhaal de proceduren voor motoruitgang „M2”.

Daarna moet het programma een bepaalde tijd wachten. Daarvoor gebruik je het wachttijd-symbool. Dit plaats je onder de tweede motor en stelt de tijd in op 6 seconden.

Aansluitend moet het rupsvoertuig 3 seconden draaien. Daarvoor voeg je nogmaals twee motorsymbolen voor M1 en M2 in. M1 moet naar links en M2 naar rechts draaien. Omdat beide motoren 3 seconden moeten werken, voeg je aansluitend het wachttijd-symbool in en zet de tijdwaarde op 3 seconden.

Daarna moet je de beide motoren stoppen. Dit gebeurt net als voor het draaien door het invoegen van de beide motorsymbolen en de parameterinstelling „Stop”.

Tenslotte moet je nog het symbool voor Einde programma, d.w.z. het „Rode verkeerslichtpoppetje” invoegen. Nu is je eerste programma klaar en kun je het opslaan. Test het programma in de online-modus. Klik daarvoor op de knop „Programma in de online-modus starten”.

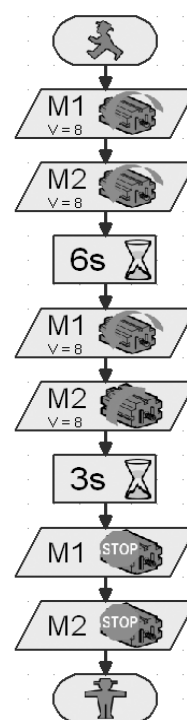
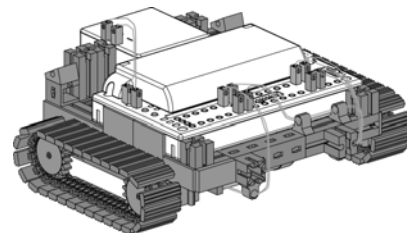
Als je alles goed hebt gedaan, kun je het programma op de interface laden. Klik daarvoor op de knop „Programma voor interface downloaden”. Neem de instelling van het download-venster over. Direct na het downloaden begint het model te rijden. Helaas hangt er nog een USB-kabel aan. Laad het programma nogmaals, activeer echter „Programma via de voeler op de interface starten”. Nadat het programma is overgedragen, kun je de kabel lostrekken. Druk de „Prog-voeler” om het programma op de interface te starten.

Het kant-en-klare programma vindt je onder:

C:\Programmas Files\Robopro\Voorbeeldprogrammas\Robo_Explorer\Basismodel_1.rpp

Dit was natuurlijk een makkie voor jou. Maar zoals gezegd, wordt je stap voor stap voorbereid. Daarom wordt elke taak iets moeilijker.

Het basismodel

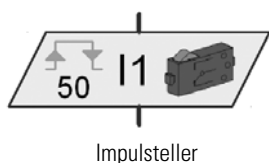
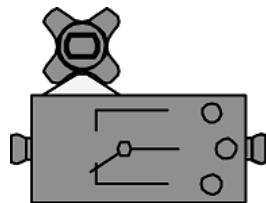


Programma in de online-modus starten



Programma downloaden naar de ROBO-interface

Impulsbesturing



■ Het basismodel heeft al 2 sensoren (voelers) voor de trajectmeting en steeds een impulswiel, die met de motoras is verbonden. Dit impulswiel bedient de schakelaar bij elke omwenteling vier maal. Controleer het model met „Interface testen” op een correcte werking.

Taak 2 - ROBO Pro Niveau 1:

Jouw rupsvoertuig moet 50 impulsen vooruit rijden. Daarna moet het 11 impulsen naar links rijden en vervolgens stoppen. Informatie over de impulsteller vindt je in het handboek in het hoofdstuk 3.6.3.



Oplossing

Schakel eerst beide motoren in. Voeg aansluitend de impulsteller toe. Verander in de eigenschappen het aantal impulsen in 50. Voeg nogmaals 2 motorsymbolen in. M1 draait verder naar „Links”, met M2 schakel je om naar „Rechts”. Dit draaien moet 11 impulsen duren. Tot slot moet je nogmaals 2 motorsymbolen invoegen en hieraan „Stop” toekennen. Draag het programma over en test het bij je eigen voertuig.

Wanneer je in het eigenschappenvenster van de impulsteller het impulstype op „0->1” of „1->0” instelt, krijg je acht in plaats van vier impulsen per omwenteling van het impulswiel. D.w.z. dat het aantal impulsen per afgelegd traject wordt verdubbeld en de nauwkeurigheid van de trajectmeting stijgt.

Het kant-en-klare programma vindt je onder:

C:\Programmas Files\Robopro\Voorbeeldprogrammas\Robo_Explorer\Basismodel_2.rpp

Subprogramma's



Nieuw subprogramma maken

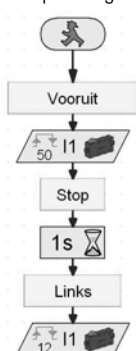


Het actuele subprogramma kopiëren



Het actuele subprogramma wissen

Oplossing



Natuurlijk heb je bij het maken van het programma al gezien dat je voor iedere verandering van richting of om te stoppen steeds de motoren moet instellen. Grote programma's worden dan al heel snel goed overzichtelijk en voor het opsporen van fouten heb je veel geduld nodig.

Hiervoor heeft de ROBO Pro echter als elegante oplossing het werken met „Subprogramma's”. Lees daarvoor in het handboek hoofdstuk 4.1 door. Het is belangrijk dat je in ROBO Pro omschakelt naar **Niveau 2**.

Taak 3 - ROBO Pro Niveau 2:

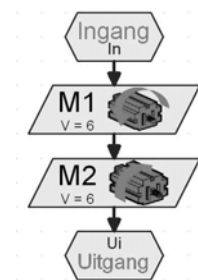
Jouw rupsvoertuig moet een vierkant rijden. Gebruik net als in de 1e taak de parameters 50 en 11. Maak nu voor iedere rijrichting en voor het stopcommando een subprogramma.



Maak eerst het subprogramma „Vooruit” (zie het ROBO Pro handboek hoofdstuk 4). Markeer de programmaonderdelen en kopieer deze naar het klembord. Vervolgens maak je het subprogramma „Links” en „Stop”. Aan beide voeg je van het klembord de programmaonderdelen voor „Vooruit” toe en wijzigt de parameters dienovereenkomstig. Voor het draaien gebruik je een lagere omwentelsnelheid. Om je te helpen, laten we je alsvast een deel van de opdracht zien.

In de tabel hieronder kun je al direct zien, hoe je de motoren voor de rijrichtingen moet programmeren.

Rijrichting	Draairichting motor 1	Draairichting motor 2
Vooruit	Links	Links
Achteruit	Rechts	Rechts
Links	Links	Rechts
Rechts	Rechts	Links
Stop	Stop	Stop



Subprogramma „Links”

Aan de hand van deze tabel zijn alle motoren in de voorbeeldprogramma's geprogrammeerd.

Kant-en-klaar programma:

C:\Programmas Files\Robopro\Voorbeeldprogrammas\Robo_Explorer\Basismodel_3.rpp

Taak 4 - ROBO Pro Niveau 3:

De robot rijdt geen exact vierkant. Controleer de volgende vragen: Wat is daarvan de oorzaak? Hoe kan het resultaat verbeterd worden?



Tip:

Synchroniseer de motoren M1 en M2 met de voelers I1 en I2 zo, dat het model exact rechthoekig rijdt. Daarvoor hebben wij voor jou het subprogramma „vooruit_sync” gemaakt. Hier hoeft je alleen het aantal impulsen als constante bij de oranje subprogramma-ingang in te voeren (zie Basismodel_4.rpp). Precies zo kun je in het subprogramma „draaien_sync” het aantal impulsen voor de draaihoek invoeren.

Kant-en-klaar programma:

C:\Programmas Files\Robopro\Voorbeeldprogrammas\Robo_Explorer\Basismodel_4.rpp

■ Nadat je voldoende met het basismodel hebt geëxperimenteerd, moet je robot nu ook op verschillende signalen van buitenaf reageren.

Om ervoor te zorgen dat jouw rupsvoertuig zijn omgeving herkent en bepaalde taken uitvoert, moet je hem met sensoren uitrusten. In de volgende modelvoorbeelden laten wij je de verschillende varianten van rupsvoertuigen met verschillende sensoren zien. Zo moeten de verschillende wegetrajecten, licht of kleuren, maar ook warmtebronnen of afstanden worden herkend. De diverse programma's vindt je in de directory: C:\Programmas Files\Robopro\Voorbeeldprogrammas\Robo_Explorer\

Natuurlijk heb je op TV wel eens films gezien van fabriekshallen zonder mensen, waarin als door een onzichtbare hand aangestuurde transportvoertuigen rijden. Voor een deel worden dergelijke systemen aangestuurd door dataleidingen in de vloer of in de vloer aangebrachte wegmarkeringen.

Uitgangspunt voor je programmering moet zijn, dat de robot langs een zwarte lijn rijdt.

Voordat je met het programmeren begint, bouw je eerst de sporenzoeker op aan de hand van de montagehandleiding. Een experimenteerparcours met opgedrukte zwarte lijn vindt je in het bouwpakket. De lijn, waarlangs de sporenzoeker moet rijden, moet in eerste instantie een rechte lijn zijn.

Autonoom rupsvoertuig

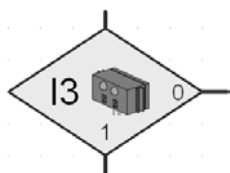
Spoorzoeker



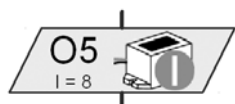


■ Hoe moet het model werken?

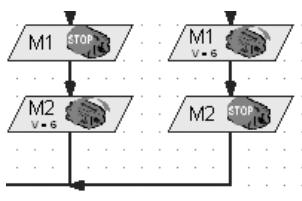
De robot moet een zwarte lijn op een witte ondergrond vinden en deze dan volgen. Om dit te realiseren, heb je in jouw model de IR-sporensensor ingebouwd. De module zendt een infrarood licht uit over de rijbaanondergrond. Dit licht wordt, afhankelijk van de ondergrond, gereflecteerd en door fototransistoren gemeten. Voor jouw programmering betekent dat: Een lichte/witte ondergrond reflecteert het licht en je krijgt de waarde 1. Bij een zwarte ondergrond wordt het licht niet gereflecteerd en je krijgt de waarde 0. Als beide tranistoren de waarde 0 hebben dan heeft jouw robot het rijtraject (zwarte lijn) gevonden en moet deze dan volgen.



Vertakking



Motoruitgang
Zoemer



Taak 1 - ROBO Pro Niveau 2:

Jouw rupsvoertuig moet op een recht, zwart spoor worden gezet en dan langs dit spoor rijden. Wijkt hij af van het spoor of houdt het spoor op, dan moet hij stil blijven staan en gedurende 1 seconden een hoorbaar signaal afgeven.

Breng het complete programma over naar een hoofdprogramma voor de sensorvraag en steeds een subprogramma voor rechttuit rijden, het zoemersignaal en stoppen.

Een paar kleine tips:

Controleer de sporenherkenning van de sensor met de interfacetest. Als de zwart-/wit-herkenning niet goed werkt, kunnen storende lichtbronnen (bijv. de zon) daarvan de oorzaak zijn. Eventueel moet de sensor iets dichterbij het spoor worden geplaatst of met een bouwplaat worden afgeschermd.

Kant-en-klaar programma: **Spoorzoeker_1.rpp**

■ Met deze eerste oplossing zul je vast nog niet tevreden zijn – want jouw robot rijdt nog maar een klein stukje langs een rechte lijn. Maar omdat je hem nog niet kunt afstellen, verlaat hij de markering, blijft stilstaan en geeft dit aan door een hoorbaar signaal.

Taak 2 - ROBO Pro Niveau 2:

Breid het hoofdprogramma zo uit met aftakkingen van de fototransistoren, dat de robot herkent wanneer hij niet meer exact op het spoor rijdt. Hij moet zijn rijrichting dan dienovereenkomstig aanpassen. Een tip vindt je in de linker programmagedeelte.

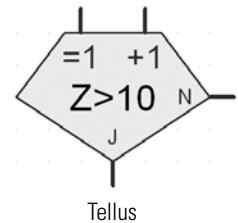


Nu ziet het er al veel beter uit. De robot blijft nu exact op het aangegeven spoor. In een fabriekshal kunnen nu andere robots de getransporteerde lading aan het einde van het spoor overnemen of de robot kan van een nieuwe lading worden voorzien. Deze lading kan dan naar het beginpunt van de rit worden teruggetransporteerd.

Kant-en-klaar programma: **Spoorzoeker_2.rpp**

Taak 3 - ROBO Pro Niveau 2:

In de vorige taken is jouw robot langs een bepaalde zwarte lijn gereden. Bij deze taak moet hij zelf een lijn zoeken. Daarvoor moet hij eenmaal een rondje draaien. Vindt hij daarbij geen spoor, dan moet hij een klein stukje rechtuit rijden en opnieuw zoeken. Nadat jouw robot een spoor heeft gevonden, moet hij dit spoor volgen. Eindigt het spoor of verliest hij deze uit het oog, dan moet hij opnieuw zoeken.

**Tip:**

Denk aan de allereerste taken in het basismodel. Hier moest de robot 90 graden ronddraaien. Dit heb je door middel van een trajectbesturing gerealiseerd. Dezelfde techniek kan ook hier van nut zijn.

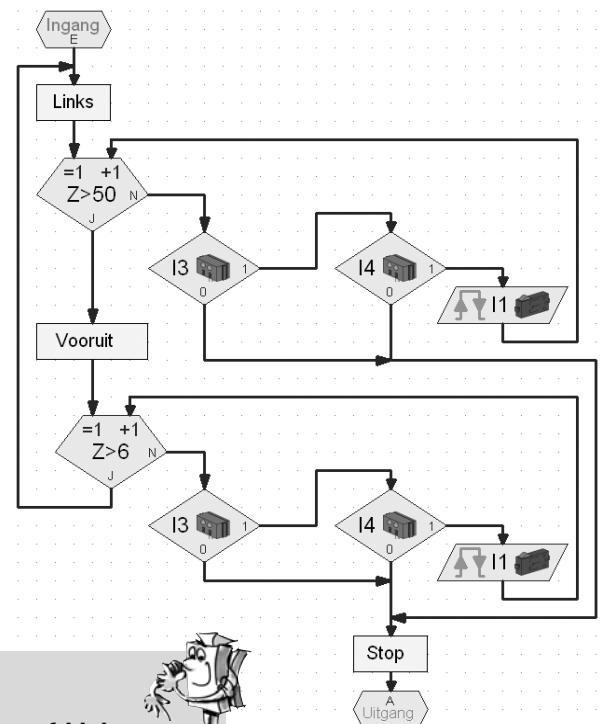
Voor het zoeken van sporen maak je een subprogramma met de naam „Spoorzoeken”. Op de afbeelding zie je, hoe dat in zijn werk gaat.

Schrijf nog een subprogramma met de naam Links. Vooruit en Stop heb je al aangemaakt.

Daarna zijn het hoofdprogramma en alle subprogramma's voor het zoeken van sporen klaar. Vervolgens moet je deze zo in het programma zetten, dat taak 3 kan worden uitgevoerd.

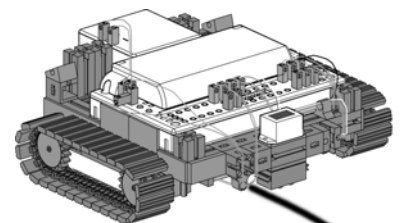
Kant-en-klaar programma: **Spoorzoeker_3.rpp**

De tot nu toe afgelegde trajecten waren altijd rechte lijnen. Soms hebben trajecten echter ook bochten. Dit zie je bijv. in fabriekshallen waarin materiaal of werkstukken van de ene machine naar de andere getransporteerd moeten worden.

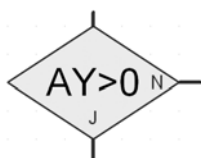
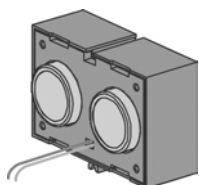
**Taak 4 - ROBO Pro Niveau 2:**

Het experimenteerparcours heeft verschillende bochten die groter of kleiner zijn.

Experimenteer bij jou eigen ronde parcours ook met verschillende snelheden van M1 en M2. Met welke instellingen legt de robot het parcours het snelste af? Noteer de resultaten in een kleine tabel.



Tunnelrobots, afstands- en temperatuursensor



- Met de sensor voor de afstandsmeting en de NTC-weerstand heb je nog meer mogelijkheden om je model te veranderen in een professionele robot.

Afstanden, temperaturen meten en evt. hulpmaatregelen treffen. Waar denk je dat robots met dergelijke vaardigheden kunnen worden ingezet? Je zult vast wel een aantal toepassingen kunnen bedenken. Wij gaan ons hier bezighouden met brandpreventie en brandbestrijding in auto- en treintunnels.



Het is de taak van de robot, zo dicht mogelijk naar de brandhaard te gaan, de temperaturen in de tunnel te meten en de informatie op een centrale plaats te melden. Meestal zijn de robots ook nog van een mobiele blusinstallatie voorzien, die indien nodig gebruikt kan worden.

Bouw ook nu, nauwkeurig volgens de montagehandleiding, het model „Tunnelrobot“ op. De afstandssensor wordt op de aansluitingen D1 aangesloten.

Taak 1: ROBO Pro Niveau 2

Net als de sporenzoeker, die langs een lijn rijdt, moet jouw robot een bepaald traject op een bepaalde afstand (van bijv. 20 cm) langs een muur rijden.



Kant-en-klaar programma: **Tunnel_1.rpp**

- Laat ons voor de volgende taak nog eenmaal naar de brandblusrobot kijken. Omdat hij net als jouw robot langs de wand rijdt, gebruikt hij afstandssensoren. Om echter de brandhaard te kunnen herkennen, gebruikt hij warmtesensoren. Voor jouw model wordt als warmtesensor een NTC-weerstand gebruikt. De natuurkundige eigenschap van dit onderdeel is, dat de weerstandswaarde bij een stijgende temperatuur wordt verlaagd. Deze verandering kun je weer met de interfacetest uitproberen. Sluit de NTC-weerstand aan op de aansluiting AY. Houd een warmtebron bij de NTC en kijk wat de blauwe balk van AY doet.

Taak 2: ROBO Pro Niveau 2

Breidt het programma uit, waarbij de robot langs de wand van de tunnel rijdt. Meet bovendien de actuele warmte via de AY-aansluiting. Stijgt deze tot een bepaalde waarde, dan moet jouw robot stil blijven staan en een warmtesignaal via de zoemer activeren. Gelijktijdig met de zoemer moet ook de rode waarschuwingslamp gaan knipperen.

Na deze gesimuleerde blusprocedure moet jouw robot omdraaien en terugkeren naar het beginpunt.



Kant-en-klaar programma: **Tunnel_2.rpp**

Tip:

Omdat jouw robot slechts één afstandssensor heeft, heeft hij een tweede nodig om langs de muur te rijden en terug naar het uitgangspunt te kunnen keren.

Wanneer je in je fischertechnik-verzameling nog een motor en een propellor hebt, kun je deze ook bij jouw programma betrekken om de brand te blussen.

■ Als volgende sensor zul je de kleursensor leren kennen. Op de afbeelding hiernaast zie je een voorbeeld van een industriële toepassing. Zoals je ziet, worden de verfbussen die in een verkeerde rij staan, uitgesorteerd.

De door het materiaal gereflecteerde licht wordt ontvangen, gedigitaliseerd en met behulp van een computer en bijbehorende software verwerkt. De sensor heeft tot taak, de verschillende kleuren te herkennen en de meetgegevens naar de interface te sturen.

In het model Kleurherkenning is de kleursensor ingebouwd. Daarbij wordt de zwarte draad aangesloten op A1, de rode draad op de + en de groene draad op de -. Voor de eerste testprogramma's gebruik je de op het parcours afgebeelde kleurenvlakken.

Taak 1 - ROBO Pro Niveau 2:

Controleer eerst de waarden die de interface voor de verschillende kleuren bij de interfacetest uitgeeft. Gebruik naast de 3 aangegeven kleuren ook nog zwart en wit.

Maak een kleine tabel en vul de waarden die je gemeten hebt, in. Bekijk ook de veranderingen, wanneer de afstand tot het gekleurde oppervlak of het omgevingslicht veranderen.

**Taak 2 - ROBO Pro Niveau 2:**

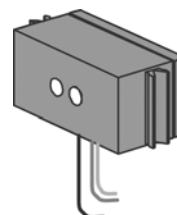
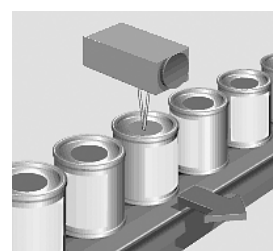
Schrijf een klein programma, waarmee de sensor het groen oppervlak kan herkennen. Licht de gemeten waarde binnen het aangegeven waardebereik, dan wordt de zoemer 1 seconde geactiveerd. Daarna keert het programma terug naar Start.

Kant-en-klaar programma: **Kleurherkenning_2.rpp**

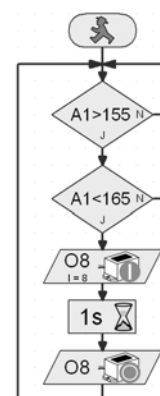
Tip:

Voor de volgende taak heb je drie lampen met lichtkappen in verschillende kleuren nodig, die al in het model zijn ingebouwd.

Kleurherkenning



Kleur	Waarde
wit	
zwart	
blauw	
rood	
groen	





Taak 3 - ROBO Pro Niveau 3:

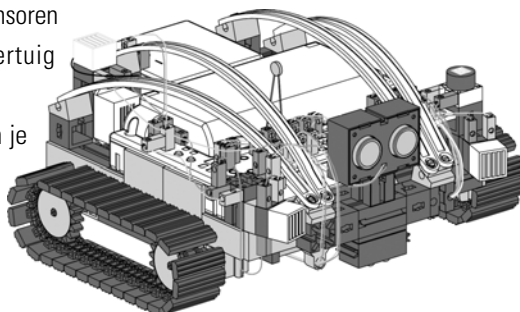
Schrijf een programma, waarin jij je robot een bepaald traject rechtuit laat rijden. Op dit traject bevinden zich drie vlakken in een verschillende kleur. Registreert de sensor een kleur, dan moet de robot 3 seconden stil blijven staan. Gedurende deze tijd worden de lampen met de bijbehorende kleur ingeschakeld en geeft de zoemer een akoestisch signaal. Daarna rijdt de robot naar het volgende vlak en herhaalt zijn taak. Vervolgens rijdt hij naar het laatste vlak, meldt het resultaat en blijft daar staan.

Kant-en-klaar programma: **Kleurherkenning_3.rpp**

Complete model Explorer

■ In het model „Explorer” zijn alle schakelaars en sensoren opgenomen, die voor een autonoom robotvoertuig noodzakelijk zijn.

Vanaf nu zijn je mogelijkheden onbegrensd en kun je alle taken, van eenvoudig tot complex, zelf uitvoeren. In de voorgaande stappen heb je meestal slechts één sensor gebruikt om de toepassingsmogelijkheid te leren kennen.



Taak 1 - ROBO Pro Niveau 2:

Programmeer je robot zo, dat hij tijdens zijn rit tegen een aanwezige hindernis rijdt. Op een afstand van ca. 60 cm moet de robot zijn snelheid met de helft verlagen. Op een afstand van 40 cm moet hij stoppen. Beweegt de hindernis nu verder naar de robot toe, dan moet de robot vanaf een afstand van 20 cm langzaam en vanaf 10 cm snel achteruit rijden.

Kant-en-klaar programma: **Explorer_1.rpp**

Taak 2 - ROBO Pro Niveau 2:

Nu gaat jouw robot op ontdekkingsreis. Maak een programma voor het gebruik van 2 sensoren - de sporensensor en de afstandssensor. Eerst moet de robot op het experimenteerparcours de zwarte lijn volgen. Op dit traject plaats je een hindernis. De robot moet ca. 10 cm voor de hindernis stoppen en een centimeter achteruit rijden. Daarna moet hij omkeren en het spoor in de andere richting volgen.



Kant-en-klaar programma: **Explorer_2.rpp**

Taak 3 - ROBO Pro Niveau 2:

Het programma uit taak 2 moet met 3 sensoren worden uitgebreid – de kleurherkenning, de temperatuursensor en de fotoweerstand voor de helderheidsmeting.

Langs het spoor bevinden zich vlakken in verschillende kleuren. Dit geeft de robot aan door verschillende akoestische signalen. Als de omgevingstemperatuur tijdens de rit te hoog wordt, moet ook de rode waarschuwingslamp knipperen. Zodra het donker in de ruimte wordt, schakelt jouw robot zijn 2 schijnwerpers in. Wordt het weer lichter, dan worden de schijnwerpers weer uitgeschakeld.



Kant-en-klaar programma: **Explorer_3.rpp**

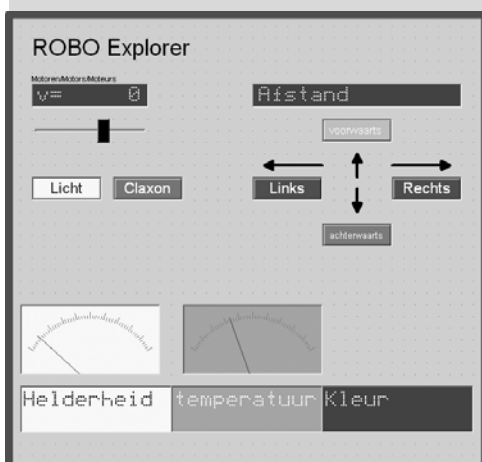
■ In het jaar 2003 was de planeet Mars de aarde dichterbij gekomen als ooit tevoren. Dit gebruikten Europese en Amerikaanse wetenschappers om elk een eigen onderzoeksvoertuig naar Mars te sturen. Zijn taak: aantonen dat er leven op Mars bestaat.

Ook onze Explorer kan als afstandsbestuurde robot zo worden geprogrammeerd, dat hij een onbekende wereld kan onderzoeken. Daarvoor heb je de extra RF Data Link nodig.

Mars - reis naar de vierde planeet

**Taak 4 - ROBO Pro Niveau 3:**

Voor deze taak vormt jouw computer de besturingscentrale voor een expeditie naar Mars. Het is de bedoeling, dat de meetwaarden van een Mars-landschap worden doorgegeven aan een station op aarde. De besturing van jouw robot wordt in de ROBO Pro in het bedieningsveld aangemaakt (zie het handboek hoofdstuk 8).

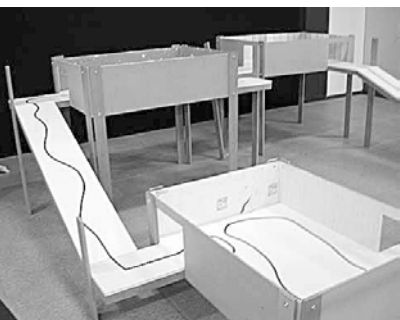


Je robot moet zo geprogrammeerd zijn, dat hij meetwaarden van de kleur van de vloer, de temperatuur, helderheid en hindernissen doorgeeft. De robot wordt handmatig aangestuurd via het bedieningsveld van het hoofdsprogramma in ROBO-Pro.

Kant-en-klaar programma: **Explorer_4.rpp**



RoboCup Junior Rescue Robot



Nu heb je alle taken uitgevoerd, die voor het bouwen van robots en de programmering ervan nodig zijn – nu kun je de vruchten van je inspanningen plukken en met het model „Rescue Robot“ toetreden tot de Rescue-Liga van RoboCupJunior.

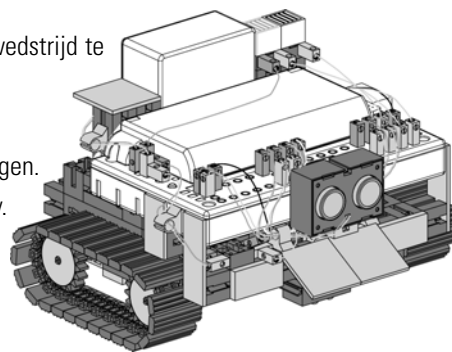
■ RoboCupJunior is een wereldwijd, projectgeoriënteerd opleidingsinitiatief, dat regionale, nationale en internationale robottentoonstellingen voor jonge mensen organiseert. Doelstelling is, om kinderen en jongeren kennis met robots en hun toepassingen te laten maken.

Aan de hand van de taakstelling, die je ook op internet vindt onder

www.robocupjunior.org

zul je het heel eenvoudig vinden, om jouw Rescue Robot voor deze wedstrijd te programmeren.

Op de linker afbeelding zie je het parcours, dat jouw robot moet afleggen. Op dit traject moeten verschillende opdrachten worden uitgevoerd, bijv. langs een rechte lijn rijden, figuren in verschillende kleuren op de vloer opzoeken, door een deur rijden enz. Is dat niet wat voor jou?



Belangrijke tips

■ Het plezier in de robotechniek kan al gauw verdwijnen, wanneer de robots niet werken zoals ze eigenlijk zouden moeten.

Meestal kan een fout met eenvoudige middelen opgespoord en verholpen worden.

Kabels

Hier moet je heel zorgvuldig te werk gaan. Eerst worden de kabels op de voorgeschreven lengte afgesneden en dan worden de uiteinden gestrips en met de stekkers verbonden. Controleer de werking met behulp van een kroonsteentje (38216) met conische insteeklamp (37869) en de accupack.

Voeding

Meestal is een bijna lege accu oorzaak van de problemen bij jouw ROBO-Explorer. Als de spanning tot onder 5 V daalt, wordt dit door een rode LED op de interface aangeduid. Gelijktijdig wordt de interface automatisch uitgeschakeld. Een onjuiste werking kan ook optreden als de accu nog niet zo leeg is. Ook dan moet de accu worden opgeladen.

Programmering

Wanneer alle mechanische problemen opgelost zijn en de robot doet nog steeds niet wat hij moet doen, dan ligt het meestal aan een onjuiste programmering. Daarvoor is de online-modus voor jouw ROBO Pro beschikbaar, waarin je het programmeerloop op het beeldscherm kunt volgen. Hier vindt je de meest kleine programmafouten die in het programma zijn geslopen.



Contenido



Vehículos con tracción de orugas	P.66
La dirección	P.66
Modelo Explorer de fischertechnik	P.67
Actores	P.67
Sensores	P.67
ROBO Interface	P.69
Suministro de corriente	P.69
Software ROBO Pro	P.69
Extensiones de interfase	P.69
Reflexiones previas	P.70
Fundamentos sobre la interfase	P.70
El modelo básico	P.71
Programa básico	P.71
Mando por impulso	P.72
Subprogramas	P.72
Vehículos autónomos de orugas	P.73
Buscador de pistas	P.73
Robot de túnel	P.76
Detector de colores	P.77
Explorer	P.78
RoboCupJunior – Rescue Robot	P.80
Sugerencias importantes	P.80

Vehículos con tracción de orugas

■ ¿Para que se necesitan vehículos autónomos con tracción de orugas? La invención de la tracción de orugas para vehículos se hizo necesaria para poder superar también terrenos intransitables. Donde una tracción de neumáticos ya no funciona, p.ej. en el desierto se empleó la tracción de orugas. De este modo en la 1ª guerra mundial se construyeron y emplearon los primeros tanques y camiones con tracción de orugas.



Según el terreno, se podían reequipar los vehículos de tracción por neumáticos a orugas.

También en el empleo civil se utilizaron vehículos de orugas. Como puedes reconocer a través de las ilustraciones, en realidad los vehículos de tracción por ruedas siempre fueron la base para los vehículos de orugas.



Pero pronto se comprobó un punto débil: Las ruedas delanteras direccionables. Por esta razón, se pasó a ampliar la tracción de orugas a todos los ejes.

W

La dirección

■ ¿Pero como funciona entonces la dirección? Muy sencillo, mediante ralentización o aceleración de una de ambas orugas. Si se quería circular una curva hacia la derecha, se ralentizaba a través del puño de mando (uno por oruga) la oruga derecha. De este modo esta se movía más lentamente y con ello el vehículo se desplazaba hacia la derecha.



También hoy, naturalmente de acuerdo al estado de la técnica, encontrará muchos vehículos con tracción de orugas. Desde excavadoras pequeñas hasta colosos gigantes en la explotación a cielo abierto de lignito.



En la pirámide de Keops en Egipto con ayuda de un minirobot se quería seguir la pista de más secretos.

Los investigadores enviaron el robot del tamaño de una locomotora de juguete a través de un pozo estrecho y oscuro. Este conduce desde una cámara al centro de la antigua pirámide de 4500 años de edad y finaliza delante de una misteriosa puerta de piedra.



■ Explorar espacios desconocidos, medir distancias, seguir pistas, indicar direcciones de marcha mediante señales intermitentes, reconocer colores, medir temperaturas, eludir obstáculos inmóviles, reconocer día y noche, conectar y desconectar automáticamente los faros, activar alarmas, etc. Todo esto – y mucho más – lo posibilitan los sensores del ROBO Explorer. En detalle estos son: el resistor NTC, el fotoresistor, el nuevo sensor de distancia por ultrasonido, el sensor de color infrarrojo así como el sensor infrarrojo de pistas especialmente desarrollado. Gracias a dos Power-Motores y el accionamiento de oruga, también se puede circular y explorar terreno intransitable. Con el modelo incluido Rescue Robot el kit de montaje es la base ideal para participar en el RoboCup-Junior.

Antes que comiences, deberías dedicarle algo de tiempo a los principales componentes. Estos se describen a continuación:

Powermotor

Para traccionar y conducir las orugas y con ello tus modelos, necesitas 2 Powermotores. Estos se conectan a la interfase en las salidas M1 a M4.

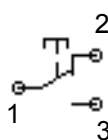
Lámpara globo

Se trata de una lámpara incandescente para una tensión de 9 V. Esta puede ser empleada como señal de aviso para la dirección de marcha o simplemente como iluminación. Ella se conecta como el Powermotor a las salidas de interfaces M1 a M4.

Zumbador

El zumbador sirve p.ej. para informar acústicamente de obstáculos o colores. Se conecta asimismo a las salidas de interfaces M1 a M4.

Pulsador



Los pulsadores se cuentan entre los sensores de contacto. Si accionas el botón rojo, en la carcasa se conmuta mecánicamente un contacto y fluye una corriente entre los contactos 1 y 3. Simultáneamente se interrumpe el tramo del interruptor entre los puntos de conexión 1 y 2.

Los pulsadores e interruptores se emplean de dos modos diferentes:

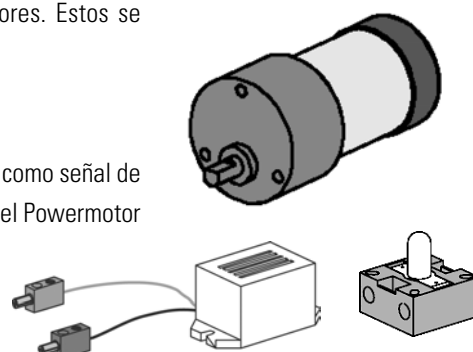
pulsador como „**contacto de cierre**”

Los dos esquemas de circuitos te muestran el montaje de ensayo. Una fuente de tensión (9V) se conecta al contacto 1 del pulsador, una lámpara en el contacto 3 del pulsador y el polo negativo de la fuente de tensión.

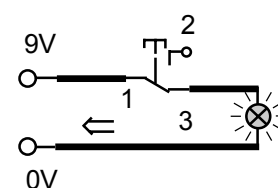
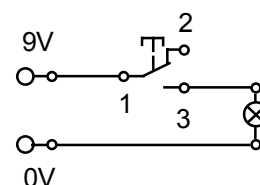
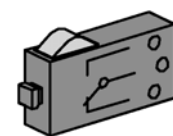
Con el pulsador no accionado la lámpara está apagada. Si se presiona el pulsador el circuito de corriente se cierra a través del contacto 1 y contacto 3 y la lámpara se enciende.

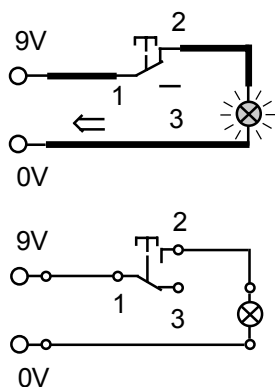
Modelo Explorer de fischertechnik

Actores



Sensores





Pulsador como „ruptor“

También aquí ambos esquemas de circuitos muestran el modo de funcionamiento. El contacto 1 se conecta con la fuente de tensión. El contacto 2 con la lámpara y la lámpara nuevamente con el polo negativo de la fuente de tensión.

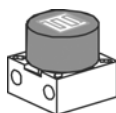
La lámpara está encendida. Si ahora accionas el interruptor, la corriente se interrumpe y la lámpara se apaga.

En tus modelos aplicas el pulsador junto con la rueda de impulsos como pulsador de conteo.



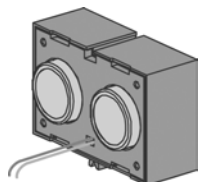
Resistor NTC (resistor de coeficiente negativo de temperatura)

En este caso se trata de un componente, con el que puedes medir diferentes temperaturas. Se habla también de un sensor de calor. Con aprox. 20 grados el resistor NTC tiene un valor de 1,5 kOhm. Si se incrementa la temperatura, desciende el valor de resistencia. Esta información puede ser leída a la interfase en las entradas analógicas AX y AY y está disponible en el ROBO Pro como valor numérico de 0 - 1023.



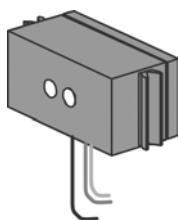
Resistor fotoeléctrico

El LDR 03, un sensor analógico de luminosidad para las entradas AX y AY (medición de resistencia), reacciona a la luz diurna y modifica en este caso su valor de resistencia. Esta es un indicador para la luminosidad de la luz.



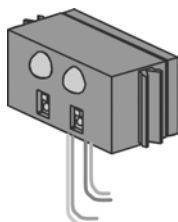
Sensor de distancia por ultrasonido

Un sensor de distancia en un componente técnico, que está en condiciones de medir la distancia entre sí mismo y un objeto. Los sensores de distancia trabajan con luz, rayos infrarrojos, ondas radioeléctricas, o **ultrasonido** y emplean diferentes métodos de medición. El sonido se propaga como onda. Un eco se refleja a la fuente de ultrasonido, la cual se atrapa como señal y de evalúa. La diferencia de tiempo entre la emisión y la recepción de la señal brinda un detalle sobre la distancia entre el obstáculo y el sensor. Como conexiones están previstas D1/D2 en la interfase. El alcance del sensor es de hasta 4 m. El valor numérico emitido corresponde a la distancia en centímetros.



Sensor óptico de colores

Los sensores de colores se emplean mayormente en la técnica de automatización. En este caso p.ej. se debe controlar el color o una impresión a color, para estar seguro que se monta el componente correcto. El sensor de colores fischertechnik emite luz roja, que se refleja con intensidad diferenciada de las diversas superficies de color. La cantidad de luz reflejada se mide a través de un fototransistor y se emite como un valor de tensión entre 0 V y 10 V. El valor de medición está en relación con la luminosidad del entorno así como de la distancia del sensor a la superficie de color. Puedes leer este valor a través de las entradas A1 y A2 y procesarlo como valor numérico de 0 - 1000 en tu programa.



Sensor de pista

El sensor de pista IR 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 de dos entradas digitales y el suministro de tensión de 9 V (positivo y negativo).

■ El componente más importante para construir un vehículo de orugas, es la ROBO Interface, la cual se monta fija en los diferentes modelos. A ella le conectas según necesidad, tus sensores y actores. El cableado básico extráelo de las instrucciones de montaje adjuntas.

■ En el caso de los modelos ROBO Explorer se trata de vehículos autónomos, que se mueven libremente en el ambiente. Por esta razón, como suministro de corriente empleas el kit acumulador fischertechnik.

■ ROBO Pro es una sencilla superficie gráfica de programación con la que tu puedes escribir tus programas. La ventaja es, que no necesitas aprender ningún idioma de programación. En realidad puedes comenzar inmediatamente.

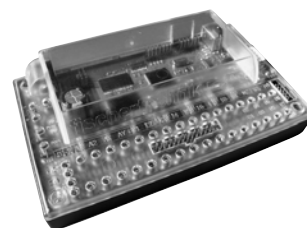
Para el módulo ROBO-Explorer necesitas la ROBO Pro Versión 1.2.1.30. En caso que dispongas de una versión más antigua, puedes actualizarla sin cargo. Ya sea descargando a través del menú Ayuda en ROBO Pro – Nueva versión o bajo

www.fischertechnik.de/robopro/update.html

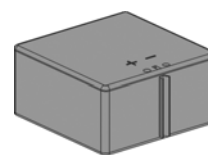
ROBO RF Data Link

La interfase inalámbrica ROBO RF Data Link sustituye el cable de interfase entre el PC y la interfase. A través de una transmisión inalámbrica los datos se transmiten a la interfase. La ventaja es, que se omite el frecuentemente molesto cable. Otra ventaja es, que puedes operar programas de modo inalámbrico en modo online. De este modo se permiten encontrar más fácilmente errores que en el servicio de descarga. Este método aún otra ventaja - los robots móviles permiten ser controlados online a través de la pantalla y pueden transmitir de forma inalámbrica los valores de medición al ordenador. Cuando hay varios robots equipados con Data Link, pueden incluso intercambiar datos entre sí, se puede decir que „dialogan“ entre sí.

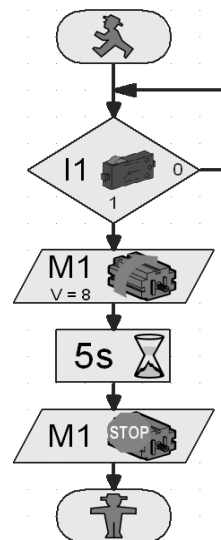
ROBO Interface



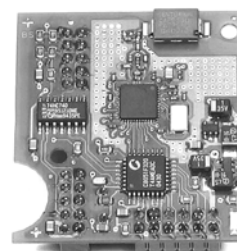
Suministro de corriente



Software ROBO Pro



Extensiones de interfase



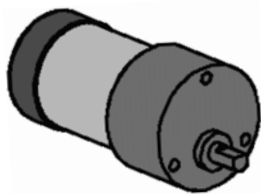
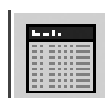
Reflexiones previas

■ Como en todos los otros robots fischertechnik, con el ROBO Explorer te introducirás paso a paso en la fascinación de la técnica y la programación. Tu comienzas con un modelo sencillo y avanzas hasta sistemas cada vez más extensos con posibilidades fascinantes. En primer plano se encuentra en todos los modelos un cuidadoso montaje y una cuidada puesta en marcha.

Fundamentos sobre la interfase

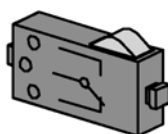
■ Antes de atreverte con los modelos individuales, deberías familiarizarte con la interfase, con ayuda de experimentos propios. Ten a mano el manual del software ROBO Pro para poder consultar en caso de problemas.

Después que hayas instalado el software, puedes conectar la interfase a tu PC a través del cable suministrado. Inicia ahora el programa ROBO Pro y abre „Comprobar la interfase“.



Powermotor

Conecta las conexiones del Powermotor con la conexión M1. Haz clic con el ratón sobre la selección „Izquierda“ o „Derecha“. El motor se pone en marcha con su velocidad máxima. Accionando el regulador puedes ajustar la velocidad de rotación. A través de „Detener“ el procedimiento finaliza.



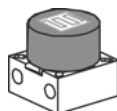
Pulsador

Conecta primeramente un pulsador (contacto de cierre) a la entrada digital I1 y observa lo que se produce en la indicación de prueba, cuando accionas el pulsador. Una indicación óptica (ganchillo) indica el estado del pulsador. Conecta el pulsador como ruptor y observa el resultado. Aparece primero la indicación de estado para „Interruptor cerrado“. Si accionas el interruptor, se apaga la indicación.

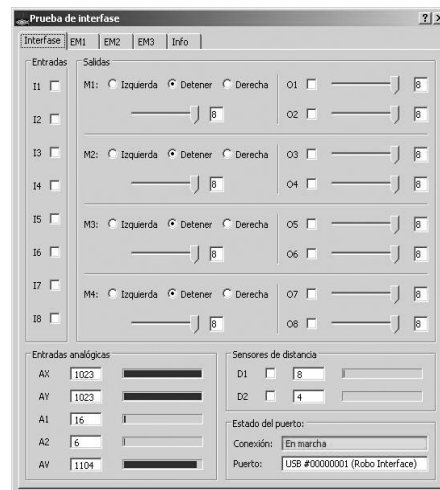
Resistor fotoeléctrico

Conecta el resistor fotoeléctrico suministrado a la conexión AX o AY.

Modifica la intensidad de luz del resistor fotoeléctrico cubriéndolo lentamente con una tira de papel negro. ¿Que pasa? Verás, se modifican la barra luminosa azul y el valor numérico de la entrada utilizada.



La prueba de la interfase se comenta muy bien en el capítulo 2 del manual. Asimismo encontrarás ayuda, en caso que haya problemas entre tu ordenador y la interfase y el software - ¡vale la pena echar una mirada!



■ Ahora nos ponemos en marcha. Después que te hayas familiarizado con la interfase y la programación, puedes ahora solucionar la primera tarea. Primero se ensambla el modelo básico de acuerdo a las instrucciones de construcción.



Tarea 1 - ROBO Pro Nivel 1:

Tu vehículo de orugas debe circular recto durante 6 segundos, a continuación debe girar 3 segundos hacia la izquierda y luego detenerse.

Con tu primer programa aún queremos brindarte un poco de soporte. Haz clic como primera medida sobre el botón „Archivo nuevo“. Tu programa se inicia con un hombrecillo en verde para el inicio del programa.



A continuación necesitas 2 símbolos de motor. Coloca el primer símbolo debajo del inicio del programa de tal manera, que la conexión se establezca automáticamente. Mueve el ratón sobre el símbolo de motor y conecta la ventana de propiedades (tecla derecha del ratón). Allí ajustas la salida del motor „M1“ y en acción el sentido de rotación „Izquierda“. A continuación confirmas con OK (Aceptar). Inserta del mismo modo el segundo símbolo de motor y repite el procedimiento para la salida de motor „M2“.

Como siguiente paso el programa debe aguardar un cierto tiempo. Para ello empleas el símbolo de tiempo de espera. Este lo ubicas debajo del segundo motor y ajustas el tiempo a 6 segundos.

A continuación el vehículo de orugas debe girar durante 3 segundos. Para ello insertas una vez más dos símbolos de motores para M1 y M2. M1 debe girar hacia la izquierda y M2 hacia la derecha. Debido a que ambos motores deben trabajar 3 segundos, inserta a continuación el símbolo de tiempo de espera y coloca el valor de tiempo en 3 segundos.

A continuación debes detener ambos motores. Esto se produce como antes de la rotación con el inserto de ambos símbolos de motores y la configuración del parámetro „Detener“.

Finalmente deben insertar aún el símbolo de final de programa „hombrecillo rojo“. Ahora tu primer programa está terminado y lo puedes guardar. Pruébalo entonces en modo online. Para ello haz clic sobre el botón „Iniciar programa en el modo online“.

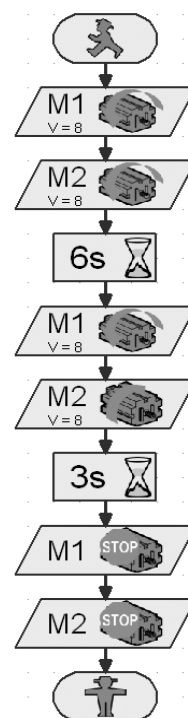
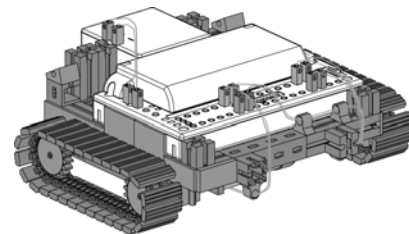
Si has hecho todo correctamente, puedes descargar el programa a la interfase. Para ello haz clic sobre el botón „Descargar programa a la ROBO interface“. Acepta la configuración de la ventana de descargas. Inmediatamente tras la descarga el modelo se pone en marcha. Lamentablemente aún está sujeto al cable USB. Carga nuevamente el programa, pero activa „Iniciar el programa a través del pulsador en la interfase“. Cuando el programa se haya transferido, puedes desenchufar el cable. Para iniciar el programa oprime en la interfase la „Tecla-Prog“.

El programa terminado lo encontrarás bajo:

C:\Programas\Robopro\Programas de ejemplo\Robo_Explorer\modelo_básico_1.rpp

Esto en realidad ha sido muy simple para ti. Pero como ya hemos comentado, deben avanzar despacio. Para ello se amplía un poco la tarea.

El modelo básico

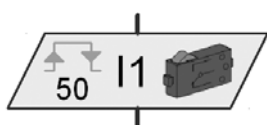
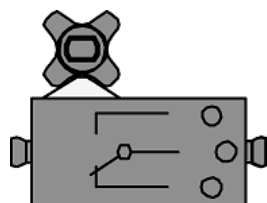


Iniciar programa en el modo online



Descargar programa a la ROBO Interface

Mando por impulso



Contador de impulso

■ El modelo básico ya contiene 2 sensores (pulsadores) para medición de recorrido así como una rueda de impulso en cada caso que está conectada con el eje motor. Esta rueda de impulso acciona cuatro veces el interruptor en cada rotación. Comprueba la capacidad de funcionamiento del modelo con „Probar la interfase“.

Tarea 2 - ROBO Pro Nivel 1:

Tu vehículo de orugas debe avanzar 50 impulsos. A continuación debe desplazarse 11 impulsos hacia la izquierda y luego detenerse. Informaciones sobre el contador de impulso las encontrarás en el capítulo 3.6.3. del manual.



Solución

Conecta primero ambos motores. Adiciona a continuación el contador de impulso. Modifica en las propiedades la cantidad de impulsos a 50. Inserta nuevamente 2 símbolos de motores M1 continua girando hacia la „Izquierda“, M2 lo conmutas a „Derecha“. Este giro debe durar 11 impulsos. Finalmente debes insertar una vez más 2 símbolos de motores y ocuparlos con „Detener“ Transfiere el programa y compruébalo en tu vehículo.

Cuando configuras en la ventana de propiedades del contador de impulsos, el tipo de impulso „0->1“ o „1->0“, obtienes ocho en lugar de cuatro impulsos por rotación de la rueda de impulso, Esto es, la cantidad de impulsos por tramo recorrido se duplica y se incrementa la exactitud de la medición de este último.

El programa terminado lo encontrarás bajo:

C:\Programas\Robopro\Programas de ejemplo\Robo_Explorer\modelo_básico_2.rpp

Subprogramas



Crear un nuevo subprograma



Copiar el subprograma actual



Eliminar subprograma actual

Seguramente has comprobado al configurar el programa, que para cada cambio de dirección o detener en cada caso se deben poner los motores. En este caso, los programas grandes rápidamente se tornan complejos y la búsqueda de errores se torna frecuentemente un juego de paciencia.

Aquí ROBO Pro ofrece como solución elegante, trabajar con „Subprogramas“. lee para ello el capítulo 4.1 en el manual. Es importante que conmutes en Robo Pro a **Nivel 2**.

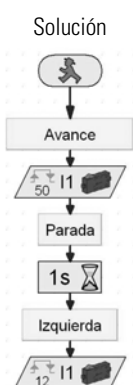


Tarea 3 - ROBO Pro Nivel 2:

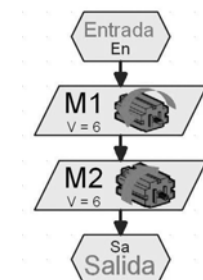
Tu vehículo de orugas debe desplazarse en un cuadrado. Emplea como en la 1ª tarea los parámetros 50 y 11. Crea para cada dirección de desplazamiento y para la orden de detener, un subprograma.

Crea primero el subprograma „Avanzar“ (véase manual ROBO Pro capítulo 4). Marca las partes del programa y cópialos en la memoria intermedia. A continuación crea el subprograma „Izquierda“ y „Detener“. En ambos insertas de la memoria intermedia las partes de programa para „Avance“ y modificas adecuadamente los parámetros. Para el giro empleas un velocidad de rotación menor. Como pequeña ayuda te indicamos una sección parcial de las tareas.

La siguiente tabla debe darte en una visión, como debes programar los motores para las direcciones de desplazamiento.



Dirección de desplazamiento	Sentido de rotación Motor 1	Sentido de rotación Motor 2
Avance	Izquierda	Izquierda
Retroceso	Derecha	Derecha
Izquierda	Izquierda	Derecha
Derecha	Derecha	Izquierda
Detener	Detener	Detener



Subprograma „Izquierda“

En función de esta tabla todos los motores están programados en los programas de ejemplo.

Programa terminado:

C:\Programas\Robopro\Programas de ejemplo\Robo_Explorer\modelo_básico_3.rpp

Tarea 4 - ROBO Pro Nivel 3:

Tu robot no se desplaza en un cuadrado exacto. Comprueba las siguientes preguntas:

¿Cual es la causa? ¿Como puede mejorarse el resultado?



Sugerencia:

Sincroniza los motores M1 y M2 a través de los pulsadores I1 e I2, de manera tal que el modelo se desplace exactamente en línea recta. Para ello hemos suministrado un subprograma terminado „avance_sync“. Aquí se debe introducir aún el número de impulsos como constante en la entrada naranja del subprograma (véase modelo_básico_4.rpp). Del mismo modo puedes introducir en el subprograma „girar_sync“ la cantidad de impulsos para el ángulo de rotación.

Programa terminado:

C:\Programas\Robopro\Programas de ejemplo\Robo_Explorer\modelo_básico_4.rpp

■ Después que hayas experimentado suficiente con el modelo básico, tu robot ahora debe reaccionar a varias señales externas.

Para que tu vehículo de orugas reconozca su entorno y realice determinadas tareas, lo debes equipar con sensores. Las siguientes propuestas de modelos te muestran diferentes variantes de vehículos de oruga con diferentes sensores. De este modo se deben reconocer diferentes tramos de recorrido, luz y colores pero también fuentes de calor o distancias. Los programas individuales los encontrarás en el directorio: C:\Programas\Robopro\Programas de ejemplo\Robo_Explorer\

Seguramente conocerás de las películas de televisión las naves de fábricas vacías de personas, en las que como controlados por una mano invisible se desplazan vehículos de transporte. En parte este tipo de sistemas se controlan con conductores de datos incorporados en el suelo o marcaciones de recorrido delineadas sobre él.

El fundamento de tu programación debe ser, que el robot se desplace a lo largo de una línea negra.

Antes que inicies la programación, ensambla primero el buscador de pistas de acuerdo a las instrucciones de construcción. Una pista de recorrido para experimentar con una línea negra impresa la encontraras en el kit de construcción. La línea, a lo largo de la cual debe desplazarse el buscador de pistas, primero debería ser una recta.

Vehículos autónomos de orugas

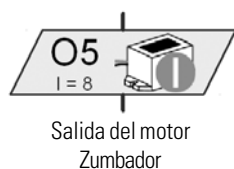
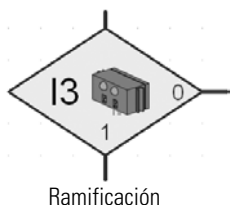
Buscador de pistas





■ ¿Como debe funcionar ahora el modelo?

El robot debe encontrar una línea negra sobre un sustrato blanco y seguir esta. Para posibilitar esto, hasta montado en tu modelo el sensor de pista IR. El componente emite una luz en la gama infrarroja sobre el sustrato de la pista de circulación. De acuerdo al sustrato, esta son reflejadas y medidas por fototransistores. Para una programación esto significa: El sustrato claro/oscuro refleja la luz y tu obtienes el valor 1. Con un sustrato negro la luz no se refleja y obtienes el valor 0. Cuando ambos transistores tiene el valor 0, el robot ha encontrado el recorrido (línea negra) y debe proceder a seguirla.



Tarea 1 - ROBO Pro Nivel 2:

Tu vehículo de orugas debe ser colocado sobre una pista recta negra y a continuación desplazarse a lo largo de ella. Si pierde la pista o esta ha finalizado debe detenerse y emitir una señal acústica durante un segundo.

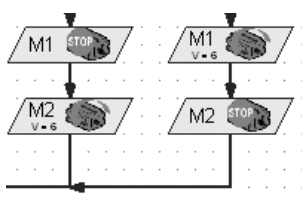
Estructura el programa completo en un programa principal para la consulta de sensores y en cada caso un subprograma para el desplazamiento recto, la señal de zumbador y para detener.

Un par de pequeñas sugerencias:

Comprueba en la prueba de la interfase la detección de pista por parte del sensor. En caso que la detección blanco y negro no funcione, la causa puede estar en la interferencia de fuentes de luz (p.ej. el sol). En caso dado el sensor debe ser posicionado algo más cerca de la pista o ser blindado con una placa de construcción.

Programa terminado: **Buscador de pistas_1.rpp**

■ Seguramente no estarás conforme con tu solución – porque tu robot se desplaza sólo un pequeño tramo a lo largo de una línea. No obstante, debido a que aún no puede reajustar, abandona la marcación, se detiene y te señaliza esto.



Tarea 2 - ROBO Pro Nivel 2:

Expande tu programa principal en las ramas de consulta de los fototransistores de tal manera, que el robot reconozca, cuando ya no circula más exactamente sobre la pista. En este caso debe corregir adecuadamente su dirección de desplazamiento. Encontrarás una sugerencia en la sección izquierda del programa.

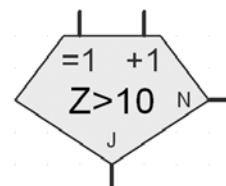


Ahora ya se ve todo mejor. Tu robot se mantiene exactamente en la pista predeterminada. En la nave industrial ahora otros robots extraerían la carga transportada al finalizar la pista o proveer el robot con una carga nueva. Esta podría entonces ser transportada de retorno al punto de partida del viaje.

Programa terminado: **Buscador de pistas_2.rpp**

Tarea 3 - ROBO Pro Nivel 2:

En las tareas anteriores, tu robot se ha desplazado a lo largo de una determinada línea negra. En esta tarea este debe buscar la línea. Para ello debe girar una vez sobre si mismo. Cuando en este caso no encuentra una pista, debe desplazarse un pequeño tramo en dirección recta y luego buscar nuevamente. Una vez que el robot ha encontrado una pista, este debe seguirla. Si finaliza o la pierde, debe comenzar nuevamente con la búsqueda.



Bucle contador

Sugerencia:

Recuerda las primeras tareas en el modelo básico. Aquí el robot debe girar en 90 grados. Esto lo has realizado con un control de recorrido. También aquí esta técnica te puede ser de utilidad.

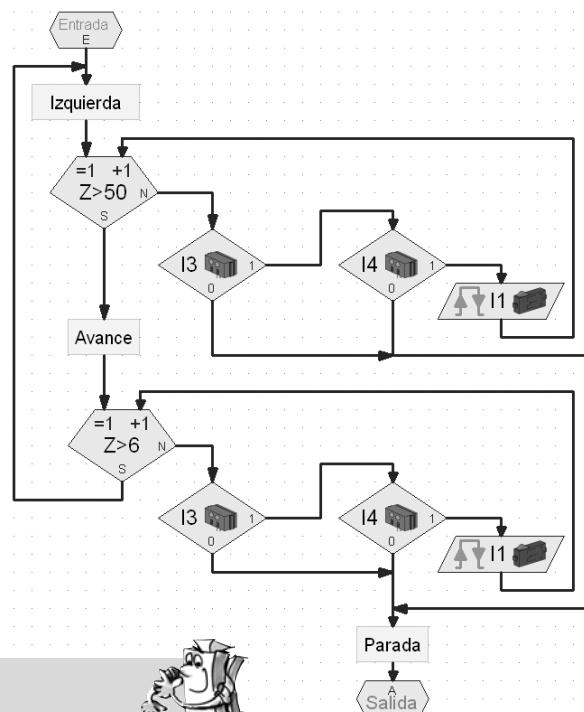
Para la búsqueda de una pista creas un subprograma propio con el nombre de „Seguim. rastros“. En la ilustración puedes ver, como lo propondríamos.

Escribe además un subprograma con el nombre Izquierda. Avance y Detener ya los has creado.

De este modo el programa principal y todos los subprogramas están listos para el buscador de pistas. La tarea es ahora, incorporar estos en el programa de tal manera, que se pueda realizar la tarea 3.

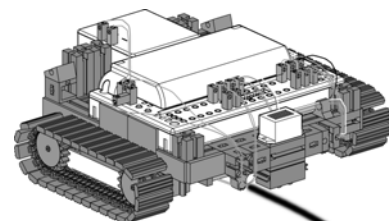
Programa terminado: **Buscador de pistas_3.rpp**

Los tramos de desplazamiento hasta ahora siempre han sido una recta. Sin embargo también sucede, que los tramos de desplazamiento estén equipados con curvas. Estas se encuentran p.ej. en instalaciones industriales en las que material o piezas deben ser transportados de una máquina a otra.

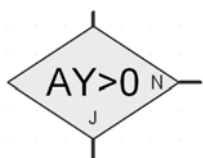
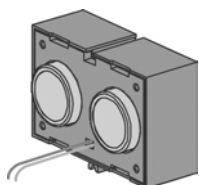
**Tarea 4 - ROBO Pro Nivel 2:**

La pista de recorrido de experimentación contiene diferentes curvas con diversos radios.

Experimenta en tu curso circular también con diferentes velocidades de M1 y M2. ¿Con que configuración logra el robot recorrer la pista con la mayor rapidez? Registra los resultados en una pequeña tabla.



Robot de túnel, sensor de distancia y de temperatura



- Con el sensor para la medición de distancia y el resistor NTC se te ofrecen otras posibilidades de expandir tu modelo a un verdadero robot profesional.

Medir distancias, temperatura y eventualmente encaminar medidas auxiliares. ¿Que es lo que opinas, donde podría ser empleados robots con estas capacidades? Seguramente se te ocurrirán para ello diferentes áreas de aplicación. Nosotros queremos ocuparnos del área de protección contra incendios y el combate de los mismos en túneles de automóviles y ferrocarriles.



La tarea de un robot de este tipo es acercarse al foco del incendio, medir temperaturas en el túnel y comunicar los datos a una central de guía. En la mayoría de los casos los robots están equipados con una instalación móvil de extinción, que puede ser empleada de acuerdo a las condiciones.

Ensambla también aquí a conciencia de acuerdo a las instrucciones de construcción el modelo „Robot de túnel“. El sensor de distancia se conecta a las conexiones D1.

Tarea 1: ROBO Pro Nivel 2

De forma similar al buscador de pistas, que se desplaza a lo largo de una línea, tu robot debe desplazarse un determinado tramo a lo largo de una pared a una determinada distancia (aprox. 20 cm).



Programa terminado: **túnel_1.rpp**

- Permítenos para la siguiente tarea abordar una vez más el tema del robot extintor de incendios. Para que como tu robot, se desplace a lo largo de la pared, este utiliza sensores de distancia. Sin embargo para detectar el foco del incendio, emplea sensores de calor. Este sensor de calor es para tu modelo el resistor NTC. Las propiedades físicas de este componente es, que ante el incremento de la temperatura se reduce la resistencia. Esta modificación la puedes probar nuevamente con la prueba de la interfase. Conecta el resistor NTC a la conexión AY. Acerca una fuente de calor al NTC y observa la barra azul de AY.

Tarea 2: ROBO Pro Nivel 2

Extiende el programa, en el cual el robot se desplaza a lo largo de la pared del túnel. Mide adicionalmente a través de la conexión AY el calor actual. Cuando este aumenta a un valor de calor determinado, el robot debe detenerse y emitir una señal de advertencia a través del zumbador. Simultáneamente con el zumbador la luz de advertencia roja debe emitir una señal intermitente.

Tras este procedimiento de extinción simulado el robot debe virar y retornar la punto de partida.



Programa terminado: **túnel_2.rpp**

Sugerencia:

Debido a que tu robot sólo posee un sensor de distancia, este necesita para retornar al punto de partida una segunda pared para desplazarse a lo largo de ella.

En caso que en tu colección fischertechnik dispongas aún de un motor y una hélice, puede incluir aún la extinción del fuego en tu programa.

■ Como siguiente sensor conocerás ahora el sensor de colores. La ilustración a un lado pretende darte un ejemplo del empleo industrial. Como podrás reconocer, se separan latas de pintura que han sido ubicadas erróneamente.

Se recepciona, digitaliza y procesa mediante un ordenador y un software, la luz reflejada por el objeto palpado. La tarea del sensor es, reconocer los diferentes colores y enviar datos de medición a la interfase.

En el modelo de detección de colores el sensor de colores está incorporado. Este se conecta con el alambre negro a A1, el alambre rojo a + y el alambre verde a -. Para los primeros programas de prueba empleas las superficies de color impresas sobre la pista de recorrido.

Tarea 1 - ROBO Pro Nivel 2:

Comprueba primero en la prueba de la interfase, los valores que la interfase emite para los diferentes colores. Emplea junto con los 3 colores preestablecidos también blanco y negro.

Confecciona una tabla y registra allí los valores que tu has medido. Observa también modificaciones, cuando se altera la distancia a la superficie de color o la luz del entorno.

**Tarea 2 - ROBO Pro Nivel 2:**

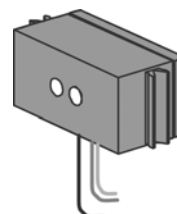
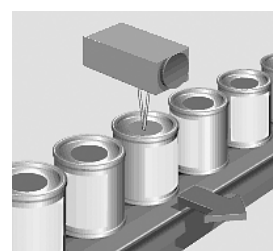
Escribe un pequeño programa, con el que el sensor detecta la superficie de color verde. Cuando el valor medido se encuentra en el rango de valor especificado, se activa el zumbador durante 1 segundo. A continuación el programa salta al inicio.

Programa terminado: **Detector de colores_2.rpp**

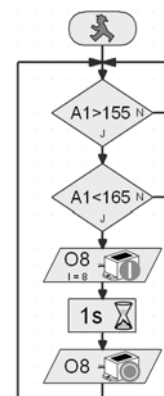
Sugerencia:

Para la siguiente tarea necesitas las tres lámparas con capuchones de color diferentes, que ya se encuentran montadas en tu modelo.

Detector de colores



Color	Valor
Blanco	
Negro	
Azul	
Rojos	
Verde	





Tarea 3 - ROBO Pro Nivel 3:

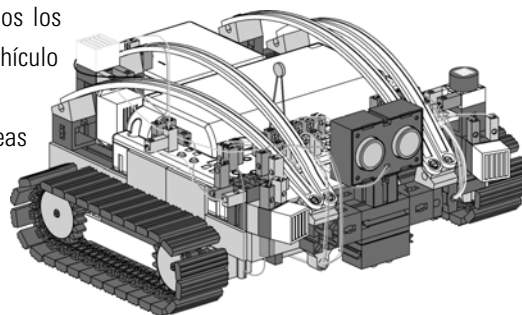
Escribe un programa, el cual permita a tu robot desplazarse en línea recta un tramo determinado. En este tramo se encuentran tres superficies de color. Cuando el sensor detecta un color, el robot debe detenerse durante 3 segundos. En este tiempo conecta las lámparas con el color correspondiente y emite una señal acústica a través del zumbador. A continuación se desplaza hasta la siguiente superficie y repite su trabajo. A continuación se desplaza hasta la última superficie, informa el resultado y allí se detiene.

Programa terminado: **Detector de colores_3.rpp**

Modelo completo Explorer

■ En el modelo „Explorer“ están contenidos todos los actores y sensores que son necesarios para un vehículo autónomo robotizado.

Ahora ya no tienes límites para solucionar tanto tareas fáciles como difíciles. En los módulos de construcción antes tratados, en la mayoría de los casos sólo has empleado un sensor para aprender las posibilidades de aplicación.



Tarea 1 - ROBO Pro Nivel 2:

Programa tu robot de tal manera, que durante el desplazamiento se encamine hacia un obstáculo. A una distancia de aprox. 60 cm debe reducir su velocidad a la mitad. A una distancia de 40 cm se detiene. Si el obstáculo continúa su movimiento hacia el robot, este debe desplazarse a partir de una distancia de 20 cm lentamente y a partir de 10 cm rápidamente hacia atrás.

Programa terminado: **Explorer_1.rpp**

Tarea 2 - ROBO Pro Nivel 2:

Ahora el robot parte hacia su viaje de exploración. Confecciona un programa para el empleo de 2 sensores - sensor de pista y sensor de distancia. Primero el robot debe seguir sobre la línea negra de la pista de recorrido de experimentación. Sobre el tramo ubicas un obstáculo. Este debe detenerse aprox. 10 cm delante del obstáculo y retroceder un centímetro. A continuación debe virar y seguir la pista en la otra dirección.



Programa terminado: **Explorer_2.rpp**

Tarea 3 - ROBO Pro Nivel 2:

El programa de la tarea 2 debe ser ampliado en tres sensores - el detector de colores, el sensor de temperatura y el fotoresistor para medición de brillo.

A lo largo de la pista se encuentran diferentes superficies de color. Estas las informa el robot a través de diversas señales acústicas. Si durante el viaje la temperatura del entorno asciende demasiado, debe parpadear la luz roja de advertencia. En el momento que se oscurece el ambiente, el robot conecta sus 2 faros. Cuando se aclara nuevamente, los faros se vuelven a desconectar.



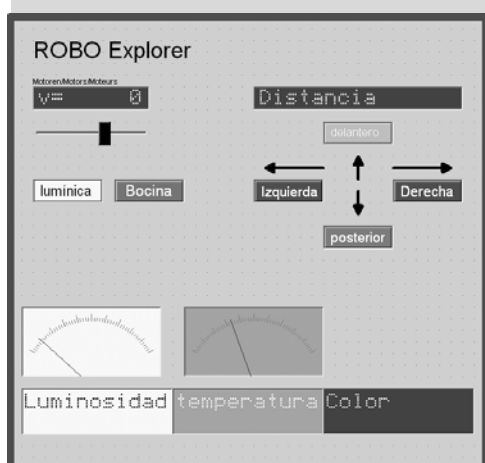
Programa terminado: **Explorer_3.rpp**

■ En el año 2003 Marte ha estado tan cerca de la tierra como sólo ocasionalmente. Esto lo aprovecharon los europeos y americanos, para cada uno enviar vehículos de exploración propios. Su tarea: Comprobar la existencia de vida.

También nuestro Explorer permite ser programado como robot controlado a distancia para investigar mundos desconocidos. Para ello se necesita adicionalmente el RF Data Link.

**Tarea 4 - ROBO Pro Nivel 3:**

Para esta tarea tu ordenador es la central de control para una expedición a Marte. La tarea es, que valores de medición de un paisaje marciano sean transmitidos a la estación de tierra. El control para tu robot se genera en el panel de control de ROBO Pro (véase capítulo 8 del manual).



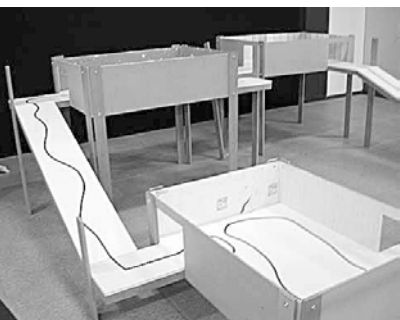
Tu robot debe ser programado de tal manera, que transmita valores de medición de color del suelo, temperatura, brillo y obstáculos. El robot se controla manualmente a través del panel de control del programa principal en ROBO-Pro.

Programa terminado: **Explorer_4.rpp**

Marte - Partida hacia el cuarto planeta



RoboCup Junior Rescue Robot



Ahora has realizado todas las tareas, te has introducido en la construcción de robots y la técnica de programación - ahora pueden recoger los frutos de tu trabajo y con el modelo „Rescue Robot“ presentarte a la liga de rescate del RoboCup Junior.

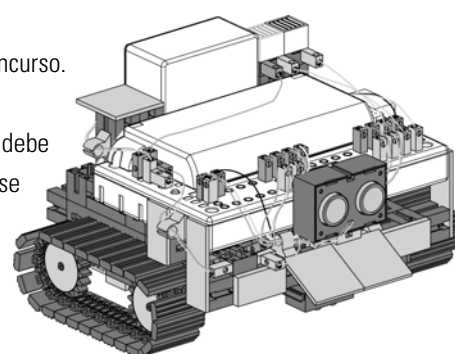
■ RoboCupJunior es una iniciativa mundial de formación orientada en proyectos, que promueve eventos internacionales de robótica para gente joven. El objetivo es, presentar robots y sus aplicaciones a niños y jóvenes.

En función de la definición del problema, que puedes obtener en Internet bajo

www.robocupjunior.org

seguramente te será fácil, programar tu Rescue Robot para este concurso.

La imagen izquierda te muestra la pista de recorrido que el robot debe atravesar. En este tramo debe realizar diferentes tareas, p.ej. desplazarse a lo largo de una línea, buscar figuras de diferentes colores en el suelo o atravesar una puerta etc. ¿No sería esto algo para ti?



Sugerencias importantes

■ La diversión en el tema de robótica puede perderse muy fácilmente cuando el robot no funciona como uno quisiera.

Frecuentemente con medios sencillo se permiten detectar errores y subsanarlos.

Cable

Aquí debes proceder exactamente. Primero se cortan los cables a la longitud predeterminada y a continuación se desaislan los extremos y se conectan firmemente con los conectores. Comprueba con ayuda de un bloque de iluminación (38216) con lámpara esférica de enchufe (37869) y con el kit acumulador, la capacidad de funcionamiento.

Suministro de corriente

Frecuentemente un acumulador casi descargado es la causa de comportamientos erróneos de tu ROBO-Explorer. Cuando la tensión desciende por debajo de 5 V, esto se indica con un LED rojo en la interfase. Simultáneamente la interfase desconecta automáticamente. Un comportamiento erróneo también se puede presentar, cuando el acumulador aún no está tan descargado. También en ese caso se debe cargar el acumulador.

Programación

Una vez solucionados todos los problemas mecánicos y el robot aún no quiere funcionar correctamente, frecuentemente esto se debe a una programación errónea. Aquí te ofrece ROBO Pro el modo online, en el que puedes hacer un seguimiento del flujo de programa sobre la pantalla. Aquí encontrarás mayormente pequeños errores de programación, que se han colado.



Índice



Veículo com acionamento de lagarta	S.82
A direção	S.82
Modelo Explorer da fischertechnik	S.83
Atores	S.83
Sensores	S.83
ROBO Interface	S.85
Alimentação elétrica	S.85
Software ROBO Pro	S.85
Extensões da interface	S.85
Considerações prévias	S.86
Princípios da interface	S.86
O modelo básico	S.87
Programa básico	S.87
Controle de pulsos	S.88
Subprogramas	S.88
Veículo de lagarta autônomo	S.89
Dispositivo de rastreamento	S.89
Robo de túnel	S.92
Identificador de cores	S.93
Explorer	S.94
RoboCupJunior – Robô de salvamento	S.96
Dicas importantes	S.96

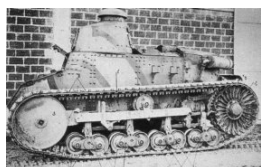
Veículo com acionamento de lagarta

■ Para que são necessários veículos autônomos com acionamento de lagarta? A descoberta do acionamento de lagarta para veículos foi necessário para que eles pudessem também passar por terrenos agrestes. Onde um pneu não mais resolve, p.ex., no deserto, o acionamento de lagarta tem a sua aplicação. Assim, na 1ª guerra mundial, foram construídos e empregados os primeiros caminhões e tanques acionados por lagarta.



Dependendo do terreno, foi possível equipar os veículos com acionamentos de pneus e de lagarta.

Os veículos de lagarta também foram empregados para o uso civil. Como podes identificar, baseado nas ilustrações, os veículos com rodas pneumáticas foram sempre a base para os veículos movidos a lagarta.



Mas, logo em seguida, foi detectado um ponto fraco: As rodas dianteiras manobráveis. Por isso, o desenvolvimento foi ampliado com o acionamento de lagarta para todos os eixos.

A direção



■ Então, a pergunta. Como funciona a direção? Muito simplesmente, pela desaceleração ou aceleração de uma das lagartas. Se o desejado é movimentar-se numa curva à direita, a lagarta direita deve ser desacelerada, através de uma barra de controle (uma por lagarta), a lagarta direita. Assim, ela gira mais lentamente e, com isso, o veículo se movimenta para à direita.

Mesmo hoje em dia, naturalmente, no mais moderno estado da técnica, irás encontrar muitos veículos acionados por lagarta. Desde pequenas escavadeiras até gigantes enormes na mineração de carvão mineral.



Na pirâmide de Queops, no Egito, foi desejado o rastreamento de outros segredos com um minirobô.

Os pesquisadores enviaram o robô do tamanho de uma locomotiva de brinquedo através de um poço escuro e estreito. Ele levou a uma câmara no centro da antiga pirâmide de 4.500 anos e parou na frente de uma porta de pedra cheia de segredos.



■ Descobrir locais desconhecidos, medir distâncias, seguir faixas, indicar direções de movimentação por sinais pisca-pisca, identificar cores, medir temperaturas, desviar obstáculos sem os tocar, detectar noite e dia, ligar e desligar faróis automaticamente, disparar alarmes, etc. Isto tudo e ainda muito mais, é possível com os sensores do ROBO Explorer. Detalhadamente: a resistência CNT, a fotoresistência, o sensor de distância de ultrassom, o sensor óptico de cores, bem como o sensor de rastreamento infravermelho especialmente desenvolvido. Devido a dois motores potentes e do acionamento de lagarta, poderão também ser explorados e rodados terrenos agrestes. Com o modelo Robô de Salvamento recebido, o módulo é uma base ideal para a participação na RoboCup-Junior.

Antes, porém, de iniciar, debes te familiarizar um pouco mais com os componentes principais. Estes serão descritos a seguir:

Motor de acionamento

Para movimentar e manobrar as lagartas e, com isso, teus modelos, necessitarás 2 (dois) motores de acionamento. Eles serão conectados nas saídas M1 até M4.

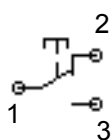
Lâmpada esférica

Aqui, trata-se de uma lâmpada incandescente para uma tensão de 9 V. Esta poderá ser empregada como sinal avisador para a direção de movimentação ou, simplesmente, como iluminação. Ela é conectada, como o motor de acionamento, nas saídas da interface M1 até M4.

Cigarra

A cigarra serve, p.ex., para notificar obstáculos ou cores identificadas acusticamente. Ela é conectada, da mesma maneira, nas saídas da interface M1 até M4.

Detetor apalpador



Os detetores apalpadores estão incluídos entre os sensores de contato. Acionando o botão vermelho, será cambiado na carcaça, mecanicamente, um contato e passa corrente entre os contatos 1 e 3. Simultaneamente, é interrompido o percurso do interruptor entre os pontos de conexão 1 e 2.

Os detetores apalpadores ou interruptores são utilizados de dois modos diversos:

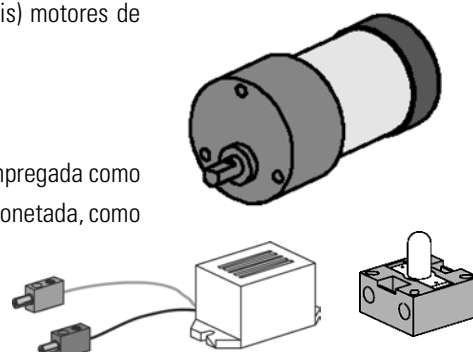
Apalpador como „Dispositivo de fechamento“

As duas ilustrações de circuitos, indicam a montagem de ensaio. Uma fonte de tensão (9V) será conectada no contato 1 do apalpador, uma lâmpada no contato 3 do apalpador e no polo negativo da fonte de tensão.

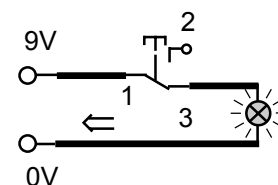
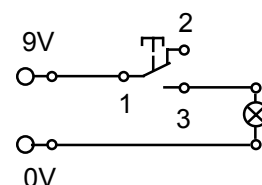
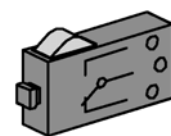
Quando o apalpador não for acionado, a lâmpada apaga. Se o apalpador for carregado, o circuito elétrico será fechado através do contato 1 e do contato 3, a lâmpada acende.

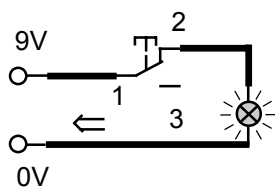
Modelo Explorer da fischertechnik

Atores



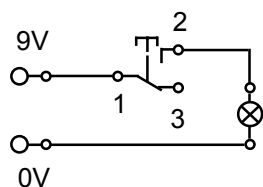
Sensores





Apalpador como „Dispositivo de abertura”

Aqui também, as duas ilustrações de circuitos te indicam o modo de funcionamento. O contato 1 será ligado com a fonte de tensão. O contato 2 com a lâmpada e a lâmpada novamente com o polo negativo da fonte de tensão.



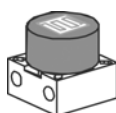
A lâmpada acende. Se acionares, agora, o interruptor, a corrente será interrompida, a lâmpada apaga.

Empregues nos teus modelos, o apalpador juntamente com a roda de pulsos como apalpador de contagem.



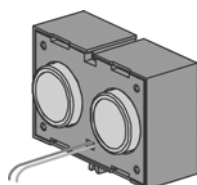
Resistência CNT (Coeficiente Negativo de Temperatura)

Aqui, trata-se de um componente, com o qual poderás medir diferentes temperaturas. Ele se denomina também um sensor térmico. A aprox. 20 graus, a resistência CNT possui um valor de 1,5 kOhm. Se a temperatura aumenta, o valor da resistência diminui. Esta informação pode ser lida na interface como entradas analógicas AX e AY e encontra-se à disposição no ROBO Pro como valor numérico de 0 - 1.023.



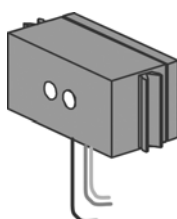
Fotoresistência

O LDR 03, um sensor de luminosidade analógico para as entradas AX e AY (medição de resistência), reage à luz diurna e modifica, com isso, o seu valor de resistência. Este é um indicador para a luminosidade.



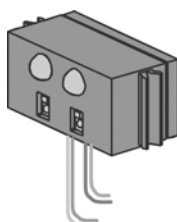
Sensor de distância de ultrassom

Um sensor de distância é um componente técnico que está em condições de mensurar a distância entre ele mesmo e um objecto. Os sensores de distância trabalham com luz, radiação infravermelha, ondas de rádio ou **ultrassom** e utilizam diversos métodos de medida. O som se espalha como uma onda. Um eco é refletido para a fonte de ultrassom, o que é novamente recebido e avaliado como um sinal. A diferença temporal entre emissão e recepção do sinal fornece informação sobre a distância entre o obstáculo e o sensor. Como conexões, são previstas D1/D2 na interface. O alcance do sensor é de até 4 m. O valor numérico fornecido corresponde à distância em centímetros.



Sensor óptico de cores

Os sensores de cor são empregados, na sua maioria, na técnica da automação. Neste caso, deve ser controlada, p.ex., a cor ou uma impressão de cores, para assegurar-se que o componente correcto foi montado. O sensor de cores da fischertechnik envia luz vermelha, que é refletida por diferentes superfícies coloridas de maneira extremamente diversa. A quantidade da luz refletida é mensurada através de um fototransistor e fornecida como valor de tensão entre 0 V e 10 V. O valor de medição depende da luminosidade do ambiente, bem como da distância do sensor à superfície colorida. Este valor pode ser lido através das entradas A1 e A2 e processado como um valor numérico de 0 - 1.000 no teu programa.



Sensor de rastreamento

O sensor de rastreamento de IV é um sensor digital de infravermelho para a identificação de uma faixa preta sobre um substrato branco em distâncias de 5 - 30 mm. Ele é constituído de dois elementos de emissão e dois de recepção. Como conexão, irás necessitar de duas entradas digitais e da alimentação de tensão de 9 V (mais ou menos).

■ O componente mais importante para montar um veículo de lagarta é a ROBO Interface, que está montada fixamente em diversos modelos. Nela irás, conforme a necessidade, conectar os teus sensores e atores. Podes verificar a fiação básica na instrução de montagem que acompanha.

■ No caso dos modelos ROBO Explorer, tratam-se de veículos autônomos, que se movimentam num espaço livre. Como alimentação elétrica irás utilizar, por isso, o conjunto de acumuladores da fischertechnik (Accu Set).

■ A ROBO Pro é uma superfície de programação gráfica simples, com a qual podes escrever os teus programas. A vantagem é que não necessitas aprender nenhuma linguagem de programação. Realmente, podes iniciar imediatamente.

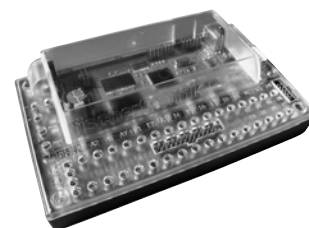
Para o módulo ROBO-Explorer, necessitas o ROBO Pro, versão 1.2.1.30. Caso tenhas uma versão mais antiga do software, poderás atualizá-la gratuitamente. Faça um download através do Menu de Ajuda, em ROBO Pro – Nova Versão ou em

www.fischertechnik.de/robopro/update.html

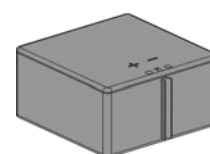
ROBO RF Data Link

A interface de rádio ROBO RF Data Link substitui o cabo da interface entre o PC e a Interface. Com uma transferência de rádio, os dados serão transferidos para a Interface. A vantagem é a não existência do cabo, frequentemente incômodo. Uma outra vantagem é a que podes operar o programa por rádio no modo online. Assim, podem ser encontrados erros mais facilmente do que na operação de download. E este método possui ainda uma vantagem a mais, os robôs móveis se deixam controlar online através do monitor e podem transmitir valores de medição sem cabo para o computador. Se vários robôs estiverem equipados com um Data Link, poderão inclusive trocar dados entre si, por assim dizer „conversar“ uns com os outros.

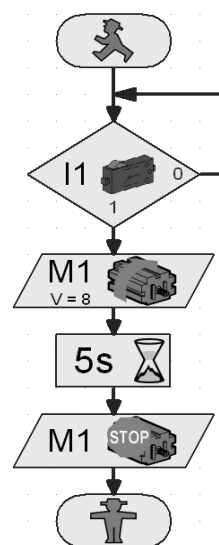
ROBO Interface



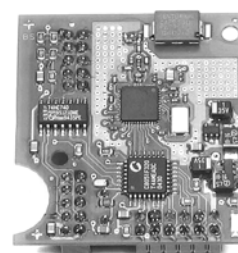
Alimentação elétrica



Software ROBO Pro



Extensões da interface



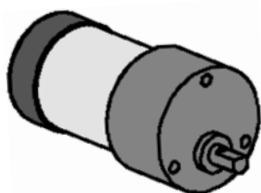
Considerações prévias

■ Como no caso de todos os outros robôs da fischertechnik, irás te familiarizar com o ROBO Explorer passo a passo na fascinação da técnica e programação. Irás iniciar com um modelo fácil e progredir até sistemas sempre mais completos, com possibilidades fascinantes. Em primeiro plano encontra-se, no caso de todos os modelos, uma construção cuidadosa e uma colocação em funcionamento cuidadosa.

Princípios em relação à interface

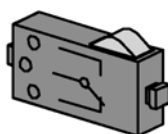
■ Antes que se aventure nos modelos individuais, deverás familiarizar-te com a interface, auxiliado por experimentos próprios. Coloque, para isso, o manual do software ROBO Pro em prontidão para poder consultar em caso de problemas.

Depois de teres instalado o software, poderás conectar a Interface, através do cabo juntamente fornecido, num PC. Inicia, a seguir, o programa ROBO Pro e abre „Teste de interface“.



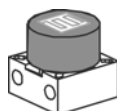
Motor de acionamento

Coneta as conexões do motor de acionamento com a conexão M1. Clica com o mouse sobre a selecção „à esquerda“ ou „à direita“. O motor se movimenta com uma velocidade máxima. Através do acionamento do regulador, poderás ajustar a velocidade de rotação. Através de „Stop“, o procedimento é encerrado.



Detetor apalpador

Fecha, a seguir, um apalpador (dispositivo de fecho) na entrada digital I1 e observa o que acontece na indicação de teste, quando acionas o apalpador. Uma indicação óptica (gancho) te indica o estado do apalpador. Fecha o apalpador como dispositivo de abertura e verifica o resultado. Primeiramente, aparece a indicação de estado para „Interruptor fechado“. Acionando o interruptor, apaga a indicação.

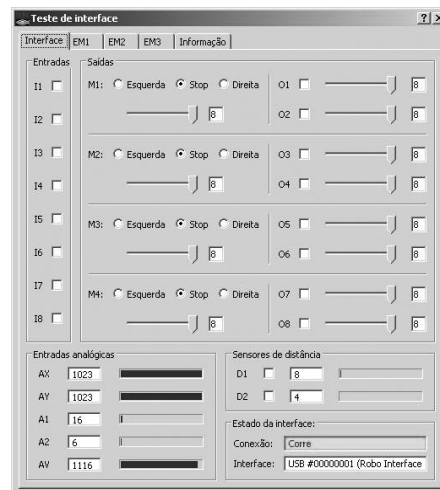


Fotoresistência

Coneta a fotoresistência juntamente fornecida na conexão AX ou AY.

Modifica a intensidade luminosa da resistência pela cobertura lenta com uma faixa de papel preto. O que ocorre? Irás ver que a barra luminosa azul se modifica e o valor numérico da entrada utilizada.

O teste da interface está muito bem esclarecido no manual, no capítulo 2. Da mesma forma, encontrarás ajuda, caso existam problemas entre o teu computador, a interface e o software, uma olhada vale a pena!



■ Agora, tudo inicia. Depois de estares familiarizado com a interface e a programação, podes resolver a primeira tarefa. Primeiramente, será montado o modelo básico, baseado no manual de construção.



Tarefa 1 - ROBO Pro, nível 1:

O veículo de lagarta deve movimentar-se em linha reta por 6 segundos, a seguir, ele deverá girar para a esquerda por 3 segundos e, a seguir, parar.

Quando do teu primeiro programa, gostaríamos ainda de te ajudar um pouco. Clica, em primeiro lugar, no botão „Arquivo-Novo“. O teu programa inicia com um sinalizador verde para o início do programa.



A seguir, irás necessitar de 2 símbolos de motor. Coloca o primeiro símbolo debaixo do início do programa, de maneira que a ligação seja puxada automaticamente. Movimenta o mouse sobre o símbolo do motor e liga a janela de características (botão direito do mouse). Ali, irás ajustar a saída do motor „M1“ e em ação, a direção de rotação „à esquerda“. A seguir, confirma com OK. Insere, da mesma maneira, o símbolo do segundo motor e repete o procedimento para a saída do motor „M2“.

A seguir, o programa deve esperar um determinado tempo. Para além disso, irás utilizar o símbolo tempo de espera. Coloca-o debaixo do segundo motor e ajusta o tempo para 6 segundos.

A seguir, o veículo de lagarta deve girar 3 segundos. Para além disso, insere novamente dois símbolos de motor para M1 e M2. M1 deve girar para a esquerda e M2 para a direita. Como ambos os motores devem trabalhar por 3 segundos, insere, a seguir, o símbolo de tempo de espera e ajusta o valor temporal para 3 segundos.

A seguir, debes parar ambos os motores. Isto acontece, como anteriormente ao giro, com a introdução dos dois símbolos de motor e o ajuste paramétrico „Stop“.

Para encerrar, debes ainda introduzir o símbolo para o encerramento do programa „Sinalizador vermelho“. Agora, o teu programa está pronto e poderás salvá-lo. Teste-o, então, no modo online. Clica, para isso, sobre o botão „Iniciar o programa no modo online“.

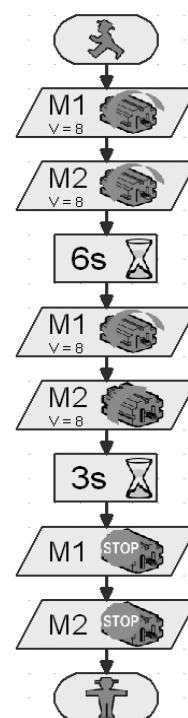
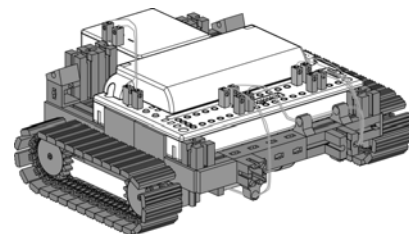
Se tiveres feito tudo corretamente, podes carregar o programa na Interface. Para isso, clica sobre o botão „Programa para transferência para a interface“. Assume o ajuste da janela Download. Logo após o download, o modelo se movimenta. Infelizmente, ele ainda está agarrado ao cabo USB. Carrega novamente o programa, ativando, entretanto „Iniciar o programa através de interruptor na interface“. Quando o programa estiver sido transferido, poderás retirar o cabo. Carregar, para iniciar o programa, na interface, a „Tecla Prog“.

Irás encontrar o programa pronto em:

C:\Programas\Robopro\Programas de exemplo\Robo_Explorer\modelo_básico_1.rpp

Isto foi simplesmente muito fácil para ti. Mas como já mencionado, debes avançar lentamente. A seguir, a tarefa será ampliada.

0 modelo básico

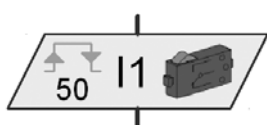
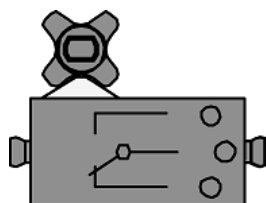


Iniciar o programa no modo online



Fazer o download do programa para a ROBO Interface

Controle de pulsos



Contador de pulsos

■ O modelo básico já possui 2 sensores (apalpadores) para a medição de percurso, assim como respectivamente, uma roda de pulsos, que está ligada com o eixo do motor. Esta roda de pulsos aciona o interruptor quatro vezes quando de cada rotação. Testa o modelo com „Testar a Interface“ quanto à capacidade de funcionamento.

Tarefa 2 - ROBO Pro, nível 1:

O teu veículo de lagarta deve movimentar-se 50 pulsos para a frente. A seguir, deve movimentar-se 11 pulsos para a esquerda e, então, parar. As informações sobre o contador de pulsos podem ser encontradas no manual, no capítulo 3.6.3.



Solução

Liga, primeiramente, os dois motores. Adiciona, a seguir, o contador de pulsos. Modifica, na propriedades, a quantidade de pulsos para 50. Insere, novamente 2 símbolos de motores. Continua a girar M1 para „à esquerda“, comuta M2 para „à direita“. Este giro deve durar 11 pulsos. Para encerrar, deverás, novamente, inserir dois símbolos de motores e comprovar com „Stop“. Transfere o programa e testa-o no teu veículo.

Se ajustares na janela de propriedades do contador de pulsos, o tipo de pulso „0->1“ ou „1->0“, irás receber oito pulsos ao invés de quatro por rotação da roda de pulsos. Isto significa que a quantidade dos pulsos por percurso percorrido duplica e a exatidão da medição de percurso aumenta.

Irás encontrar o programa pronto em:

C:\Programas\Robopro\Programas de exemplo\Robo_Explorer\modelo_básico_2.rpp

Subprogramas



Gerar novo subprograma



Copiar o subprograma corrente



Apagar o subprograma corrente

■ Certamente, determinaste que, quando da elaboração do programa, deve-se empregar os motores a cada vez para cada modificação de direção ou Stop (parada). Os grandes programas ficam, então, rapidamente pouco claros e a busca de erros se torna frequentemente um jogo de paciência.

Aqui, o ROBO Pro oferece, como solução elegante, o trabalho com „Subprogramas“. Lê, adicionalmente, no manual, o capítulo 4.1. O importante é que comutes no ROBO Pro para o **Nível 2**.



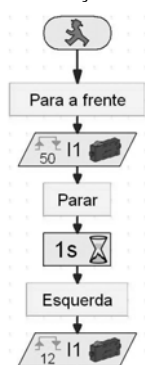
Tarefa 3 - ROBO Pro, nível 2:

O teu veículo de lagarta deve movimentar-se num trajeto quadrado. Utiliza, como na 1a. tarefa, os parâmetros 50 e 11. Cria um subprograma para cada direção de movimentação e para o comando Stop.

Cria primeiramente o subprograma „Para a frente“ (vide o manual do ROBO Pro, capítulo 4). Marca a parte do programa e copia-o para a memória intermediária. A seguir, cria o subprograma „à esquerda“ e „Stop“. Em ambos insere, a partir da memória intermediária, a parte de programa para „Para a frente“ e modifica os parâmetros de maneira correspondente. Para a rotação, utiliza uma velocidade de rotação baixa. Como auxílio, destacamos uma secção parcial da tarefa.

A tabela seguinte deve te mostrar numa visão geral, como deves programar os motores para as direções de movimentação.

Solução



Direção de mov.	Direção de rot. motor 1	Direção de rot. motor 2
Para a frente	À esquerda	À esquerda
Para trás	À direita	À direita
À esquerda	À esquerda	À direita
À direita	À direita	À esquerda
Stop	Stop	Stop

Baseando-se nesta tabela, devem ser programados todos os motores nos programas exemplo.

Programa pronto:

C:\Programas\Robopro\Programas de exemplo\Robo_Explorer\modelo_básico_3.rpp

Tarefa 4 - ROBO Pro, nível 3:

O robô não se movimenta exatamente num percurso quadrado. Testa as seguintes questões: Qual a causa? Como pode ser melhorado o resultado?

Dica:

Sincroniza os motores M1 e M2 através dos apalpadores I1 e I2 de maneira que o modelo se movimente exatamente em linha reta. Adicionalmente, fornecemos incluído para ti um subprograma pronto „para a frente_sync“. Aqui, somente deve ser fornecida ainda a quantidade dos pulsos como constante na entrada cor-de-laranja do subprograma (vide modelo_básico_4.rpp). Da mesma forma, poderás entrar, no subprograma „rodar_sync“, a quantidade de pulsos para o ângulo de rotação.



Programa pronto:

C:\Programas\Robopro\Programas de exemplo\Robo_Explorer\modelo_básico_4.rpp

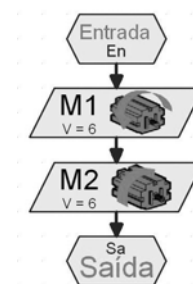
■ Depois de teres experimentado suficientemente com o modelo básico, o teu robô deve reagir a diversos sinais externos.

Para que o teu veículo de lagarta identifique o seu ambiente e cumpra determinadas tarefas, deverás equipá-lo com sensores. As seguintes propostas de modelo te mostram diversas variantes de veículos de lagarta com diferentes sensores. Assim, devem ser identificados diversos percursos, luz ou cores, mas também fontes térmicas ou distâncias. Podes encontrar os programas individuais no diretório: C:\Programas\Robopro\Programas de exemplo\Robo_Explorer\

Seguramente, é de teu conhecimento, nos filmes da televisão, pavilhões de fábricas completamente sem pessoas, nos quais veículos de transporte, comandados por mãos de fantasmas, se movimentam. Tais sistemas são controlados parcialmente com condutores de dados deixados no solo ou marcações de percurso registradas no solo.

A base da tua programação deve ser que o robô se movimenta ao longo de uma linha preta.

Antes de iniciar a programação, controlo, primeiramente, baseado no manual de construção, o dispositivo rastreador. Irás encontrar um percurso experimental com linha preta impressa no módulo. A linha, ao longo da qual o dispositivo de rastreamento deve movimentar-se, deve ser, primeiramente, uma linha reta.



Subprograma „À esquerda“

Veículo de lagarta autônomo

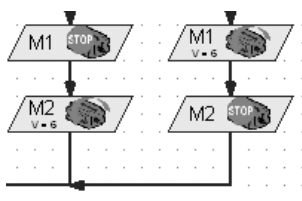
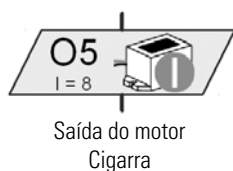
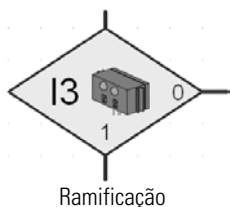
Dispositivo de rastreamento





■ Como deve, então, funcionar o modelo?

O robô deve encontrar, sobre um substrato branco, uma linha preta e segui-la. Para que isso seja possível, montaste no teu modelo um sensor de rastreamento de infravermelho. O componente envia um luz na região do infravermelho sobre a base da faixa de rodagem. Esta será refletida, dependendo do substrato, e mensurada pelos fototransístores. Para a tua programação, isto significa: Substrato claro/branco reflete a luz e obténs o valor 1. No caso de um substrato preto, a luz não será refletida e obténs o valor 0. Se os dois transístores possuem o valor 0, o teu robô encontrou o percurso de movimentação (linha preta) e deve segui-la.



Tarefa 1 - ROBO Pro, nível 2:

O teu veículo de lagarta deve ser colocado sobre uma faixa preta e movimentar-se ao longo dela. Se ele abandona a faixa ou esta está no fim, ele deve parar e emitir um sinal acústico por um segundo.

Estrutura o programa total num programa principal para a consulta do sensor e respectivamente um subprograma para a movimentação em linha reta, o sinal da cigarra e para parar.

Algumas dicas simples:

Verificar, com o teste da interface, a deteção da faixa do sensor. Se a deteção preto-branco não funciona corretamente, as causas podem ser fontes de luz (p.ex. sol) perturbadoras. Caso necessário, o sensor deve ser posicionado um pouco mais próximo na faixa ou blindado com uma placa.

Programa pronto: **Dispositivo de rastreamento_1.rpp**

■ Seguramente, não ficarás satisfeito com a tua primeira solução, pois o robô movimenta-se apenas ao longo de um pequeno percurso. Pois como ele ainda não pode reajustar, ele abandona a marcação, permanece parado e te sinaliza isso.

Tarefa 2 - ROBO Pro, nível 2:

Amplia o teu programa principal com uma ramificação de consulta dos fototransístores, de maneira que o robô perceba quando ele não mais se movimenta exatamente sobre a faixa. A seguir, ele deve corrigir a sua direção de movimentação de maneira correspondente. Irás encontrar uma dica no trecho esquerdo do programa.

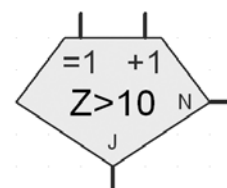


Agora, já parece melhor. O robô permanece exatamente sobre a faixa prevista. Num pavilhão industrial, outros robôs recolhem, agora, a carga transportada no final da faixa ou carregam o robô com uma nova carga. Este poderia, então, ser retornado para o ponto de partida da viagem.

Programa pronto: **Dispositivo de rastreamento_2.rpp**

Tarefa 3 - ROBO Pro, nível 2:

O teu robô movimentou-se ao longo de uma determinada linha preta nas tarefas anteriores. Nesta tarefa, ele deve procurar uma linha. Para isso, ele deve girar uma vez em círculo. Se ele não encontrar nenhuma faixa, ele deve movimentar-se um pequeno trajeto em linha reta e, a seguir, procurar novamente. Quando o teu robô encontrar uma faixa, ele deve segui-la. Se esta termina ou ele a perde, ele deve reiniciar com a busca.



Ciclo de contagem

Dica:

Lembras da primeira tarefa no modelo básico. Naquela tarefa, o robô tinha que girar de 90 graus. Isto foi realizado com um controle de percurso. Aqui, também, esta técnica pode te auxiliar.

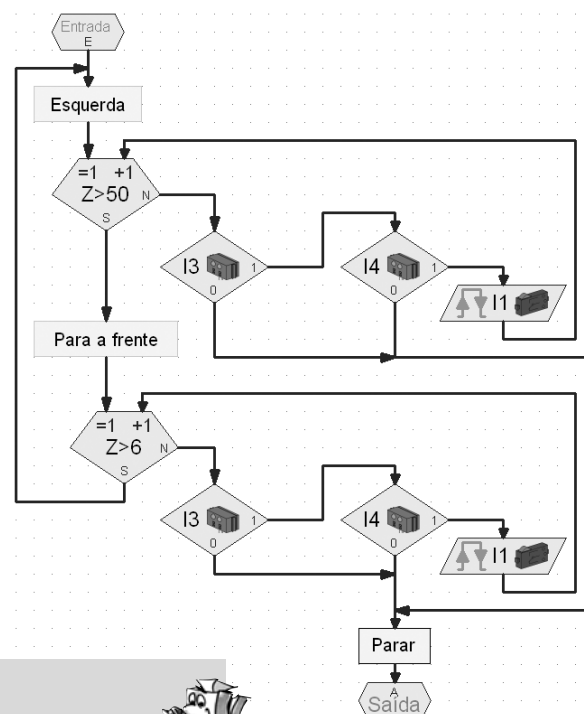
Compila, para a busca da faixa, um subprograma próprio com o nome „Busca trilha“. Na ilustração estás vendo como iríamos propor este subprograma.

Escreve ainda um subprograma com o nome À esquerda. Para a frente e Stop tu já fizeste.

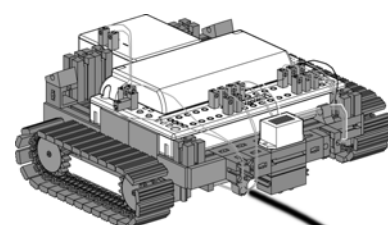
Com isso, estão prontos o programa principal e todos os subprogramas para o dispositivo de rastreamento. A tua tarefa é, agora, integrá-los no programa, de maneira que a tarefa 3 possa ser executada.

Programa pronto: **Dispositivo de rastreamento_3.rpp**

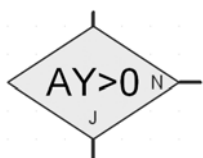
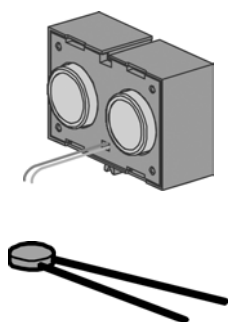
Os trajetos de movimentação até o momento foram sempre uma linha reta. Mas, pode ocorrer, que os trajetos de movimentação sejam dotados de curvas. Isto pode ser encontrado, p.ex., em instalações industriais nas quais o material ou as peças tenham de ser transportadas de uma máquina para a próxima.

**Tarefa 4 - ROBO Pro, nível 2:**

O percurso experimental recebe diversas curvas com raios diferentes. Experimenta, no teu circuito também com diferentes velocidades de M1 e M2. Com qual configuração, o robô vence o percurso mais rapidamente? Coloca os resultados numa pequena tabela.



Robô de túnel, sensor de distância e de temperatura



■ Com o sensor para a medição de distância e a resistência CNT são oferecidas outras possibilidades para ampliar a montagem do teu modelo para um robô profissional.

Medir distâncias, temperaturas e, eventualmente, conduzir medidas de ajuda. O que achas? Onde poderiam ser empregados robôs com estas capacidades? Seguramente, vem a tua mente, aqui, diversas áreas de aplicação. Gostaríamos de nos ocupar com o sector da proteção contra incêndio e o combate ao incêndio em túneis de rodovias e ferrovias.



A tarefa de um tal robô é sondar perto do foco do incêndio, medir temperaturas no túnel e informar os dados para uma central de comando. Na sua maioria, os robôs estão equipados com um dispositivo móvel de extinção, que podem ser empregados conforme as condições.

Monta aqui, também, conscientemente, o modelo „Robô de túnel“ conforme a instrução de montagem. O sensor de distância deve ser conectado nas conexões D1.

Tarefa 1: ROBO Pro, nível 2

De maneira similar ao dispositivo de rastreamento, que se movimenta ao longo de uma linha, o teu robô deve movimentar-se numa parede, a uma determinada distância (aprox. 20 cm), num determinado percurso.



Programa pronto: **túnel_1.rpp**

■ Deixe-nos concentrar, para a próxima tarefa, ainda uma vez, com um robô extintor de incêndio. Para que ele se movimente, como o teu robô, ao longo da parede, ele utiliza sensores de distância. Para identificar o foco do incêndio, entretanto, ele aplica sensores térmicos. Este sensor térmico, para o teu modelo, é a resistência CNT. A característica física deste componente é a de que o valor da resistência diminui com o aumento da temperatura. Poderás novamente testar esta modificação com o teste da Interface. Coneta a resistência CNT na conexão AY. Mantenha uma fonte de calor na CNT e observa a barra azul no AY.

Tarefa 2: ROBO Pro, nível 2

Amplia o programa que movimenta o robô ao longo da parede do túnel. Meça o calor atual, adicionalmente, através da conexão AY. Se este aumenta acima de um determinado valor de temperatura, o teu robô deve parar e emitir um sinal de alerta através da cigarra. Simultaneamente ao som da cigarra, o luminoso de alerta vermelho deve emitir um sinal intermitente (piscar).

Após este procedimento simulado de extinção, o teu computador deve virar e retornar para o ponto de partida.



Programa pronto: **túnel_2.rpp**

Dica:

Como o teu robô somente possui um sensor de distância, ele necessita, para o retorno ao ponto de partida, uma segunda parede, ao longo da qual ele possa se movimentar.

Se ainda tiveres, na tua coleção fischertechnik um motor e uma hélice, poderás incluir a extinção do incêndio no teu programa.

■ Como sensor adicional irás conhecer o sensor de cores. A ilustração ao lado deve te dar um exemplo de uma aplicação industrial. Como podes identificar, latas coloridas, que foram enfileiradas incorretamente, serão triadas.

A luz refletida pelo material de detecção é recebida, digitalizada e preparada através de um computador e um software. A tarefa do sensor é identificar as diferentes cores e enviar dados de medição para a Interface.

O sensor de cores está montado no modelo identificador de cores. Ele é conectado com o fio preto na A1, o fio vermelho no + e o fio verde no -. Para o primeiro programa de teste, irás utilizar as superfícies coloridas impressas sobre o percurso.

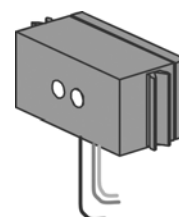
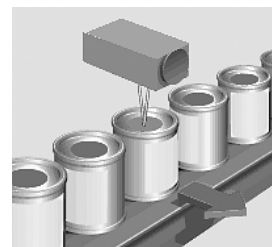
Tarefa 1 - ROBO Pro, nível 2:

Verifica primeiramente os valores que a interface fornece para as diversas cores no teste da interface. Utiliza, além das 3 cores pré-fornecidas, ainda o preto e o branco.

Cria uma pequena tabela e registra os valores que mediste. Observa, também, modificações, quando a distância à superfície colorida ou a luz ambiental for alterada.

Tarefa 2 - ROBO Pro, nível 2:

Escreve um programa curto, com o qual o sensor identifica a superfície colorida verde. Se o valor mensurado se encontrar no intervalo de medição pré-fornecido, a cigarra será activada por um segundo. A seguir, o programa salta para o início (Start).

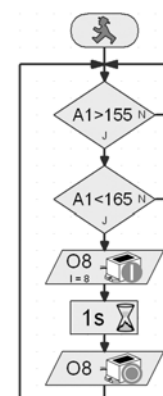


Cor	Valor
Branco	
Preto	
Azul	
Vermelho	
Verde	

Programa pronto: **Identificador de cores_2.rpp**

Dica:

Para a próxima tarefa, irás necessitar das três lâmpadas com tampas luminosas de cores diferentes, que já estão montadas no modelo.



**Tarefa 3 - ROBO Pro, nível 3:**

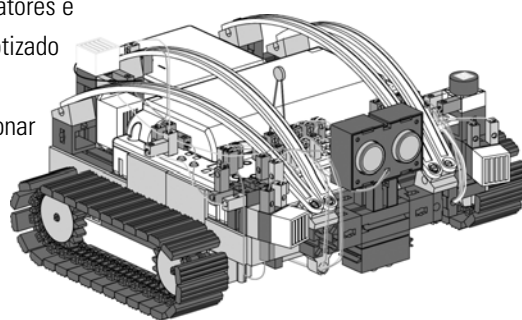
Escreve um programa que permita o teu robô se movimentar num determinado percurso em linha reta. Sobre o percurso se encontram três superfícies coloridas. Se o sensor detetar uma cor, o robô deverá parar por 3 segundos. Neste tempo, ele liga a lâmpada com a cor correspondente e fornece um sinal acústico através da cigarra. A seguir, ele se movimenta para a próxima superfície e repete o seu trabalho. Para finalizar, ele se movimenta para a última superfície, notifica o resultado e permanece parado.

Programa pronto: **Identificador de cores_3.rpp**

Modelo Explorer completo

■ No modelo „Explorer“, estão incluídos todos os atores e sensores, que são necessários para um veículo robotizado autônomo.

Agora, não tens mais nenhuma limitação para solucionar tarefas tanto simples quanto difíceis. Nos níveis de construção anteriores utilizastes, na maioria das vezes, somente um sensor para conhecer a possibilidade de emprego.

**Tarefa 1 - ROBO Pro, nível 2:**

Programa o teu robô, de maneira que ele se dirija para um obstáculo existente na sua viagem. A uma distância de aprox. 60 cm, ele deve reduzir a sua velocidade à metade. A uma distância de 40 cm, ele para. Se o obstáculo continuar a se movimentar na direção do teu robô, este deve se movimentar mais lentamente a partir de uma distância de 20 cm e rapidamente em marcha a ré a partir de 10 cm.

Programa pronto: **Explorer_1.rpp**

Tarefa 2 - ROBO Pro, nível 2:

Agora o teu robô vai a uma viagem de reconhecimento. Cria um programa para o emprego de 2 sensores: o sensor de rastreamento e o sensor de distância. Primeiramente, o robô deve seguir a linha preta sobre o percurso experimental. Sobre o percurso, irás colocar um obstáculo. Ele deve parar a aprox. 10 cm antes do obstáculo e retornar de um centímetro. A seguir, ele deve girar e seguir a faixa na outra direção.



Programa pronto: **Explorer_2.rpp**

Tarefa 3 - ROBO Pro, nível 2:

O programa da tarefa 2 deverá ser ampliado de três sensores: o identificador de cores, o sensor de temperatura e a fotoresistência para medição de luminosidade.

Ao longo da faixa encontram-se diversas superfícies coloridas. Isto é notificado pelo robô através de diferentes símbolos acústicos. Se a temperatura ambiente ficar muito alta durante a viagem, a lâmpada avisadora vermelha deve piscar. Logo que escureça no ambiente, o robô acende seus dois faróis. Se ficar novamente claro, os faróis desligam novamente.



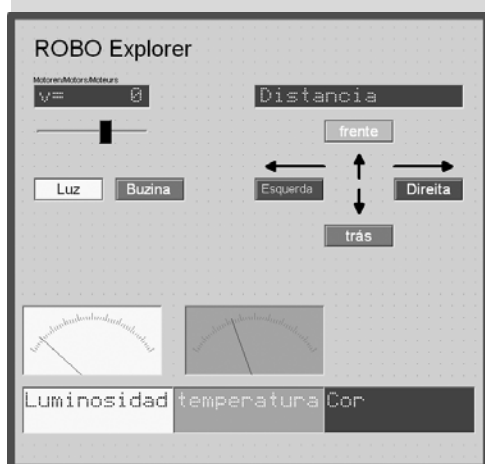
Programa pronto: **Explorer_3.rpp**

■ No ano de 2003, Marte esteve tão próximo da Terra como nunca. Isto foi utilizado pelos europeus e americanos para enviarem seus próprios veículos de reconhecimento. A sua missão: Comprovar a existência de vida.

Também o nosso Explorer, se deixa programar como robô controlado à distância para explorar mundos desconhecidos. Para isso, será necessária, adicionalmente, a RF Data Link.

**Tarefa 4 - ROBO Pro, nível 3:**

Para esta tarefa, o teu robô é a central de controle para uma expedição a Marte. A tarefa é transmitir os valores de medição de uma paisagem marciana para a estação em terra. O controlo para o teu robô é gerado no ROBO Pro, no Painel de comando (vide o manual, capítulo 8).



O teu robô deverá ser programado de maneira que transmita os valores mensurados da cor do solo, temperatura, luminosidade e obstáculos. O robô será controlado manualmente através do campo de comando do programa principal no ROBO-Pro.

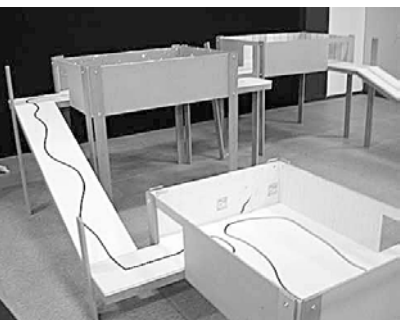
Programa pronto: **Explorer_4.rpp**

Marte - Partida para o quarto planeta



RoboCup Junior

Robô de salvamento



Agora, cumpristes todas as tarefas, estás acostumado à construção do robô e à técnica de programação, poderás, então, colher os frutos do teu trabalho e entrar com o modelo „Robô de salvamento” na Divisão Rescue da RoboCupJunior.

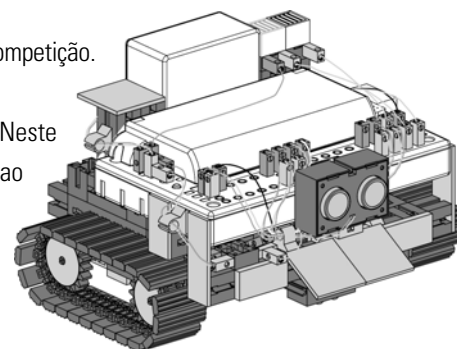
■ A RoboCupJunior é uma iniciativa de formação mundial, orientada a projetos, que promove eventos de robôs para pessoas jovens a nível regional, nacional e internacional. O objetivo é apresentar, às crianças e jovens, robôs e as suas aplicações.

Baseando-se na proposição de tarefa, que podes verificar a partir da Internet em

www.robocupjunior.org

será seguramente fácil, programar o teu Robô de Salvamento para esta competição.

A figura esquerda indica um percurso, que o teu robô deve passar. Neste percurso, ele deve executar diversas tarefas, p.ex., movimentar-se ao longo de uma linha, buscar figuras de cores diferentes no solo ou passar por uma porta, etc. Não seria algo para ti?



■ A diversão sobre o assunto robótica pode perder-se muito rapidamente quando o robô não funcionar como a gente deseja.

Frequentemente, podem ser identificados e eliminados erros com meios fáceis.

Cabo

Aqui, deves proceder da maneira exata. Primeiramente, os cabos serão cortados no comprimento prescrito e, então, isoladas as extremidades e ligadas com as fichas firmemente. Verifica, auxiliado por um luminoso (38216) com uma lâmpada de inserção esférica (37869) e o pacote de bateria recarregável, a capacidade de funcionamento.

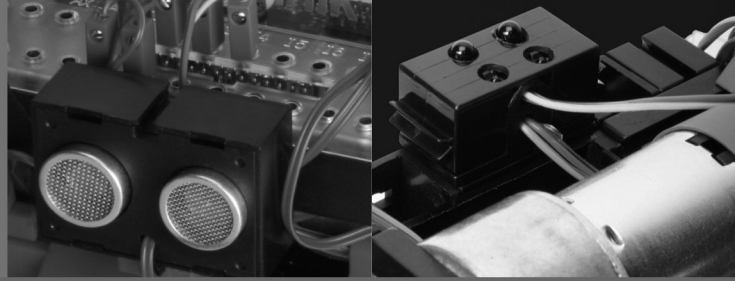
Alimentação elétrica

Frequentemente, uma bateria quase descarregada é a causa do comportamento incorreto do teu ROBO-Explorer. Se a tensão fica abaixo de 5 V, isto é indicado com um LED vermelho na interface. Simultaneamente, a interface desliga automaticamente. Um comportamento falho pode também ocorrer quando a bateria ainda não está tão descarregada. Aqui também, a bateria deverá ser carregada.

Programação

Se todos os problemas mecânicos estiverem solucionados e o robô ainda não funcionar corretamente, isto é devido, frequentemente, a uma programação incorreta. Aqui, o ROBO Pro te oferece o modo online, no qual poderás seguir o decurso do programa no monitor. Ali, irás encontrar, na maioria das vezes, os erros pequenos, que se integraram no programa.



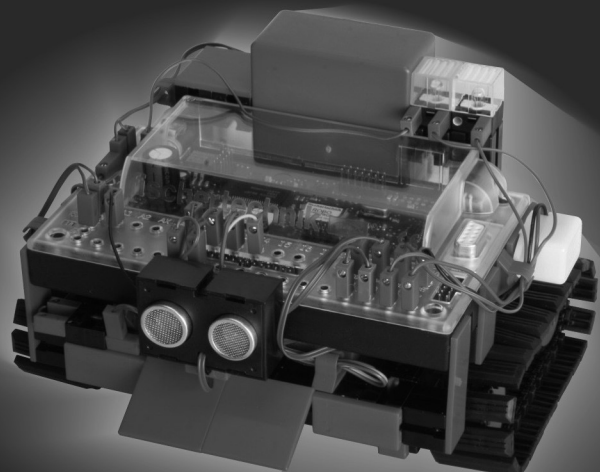
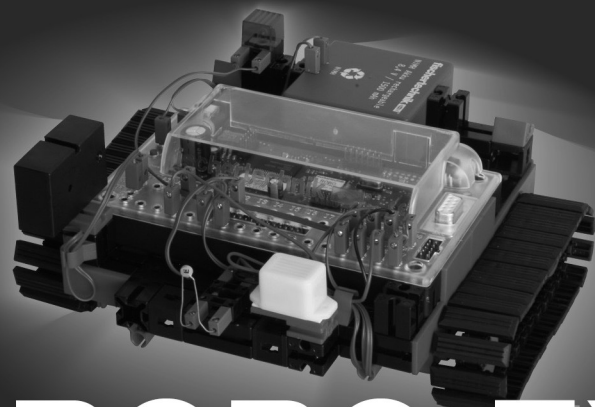
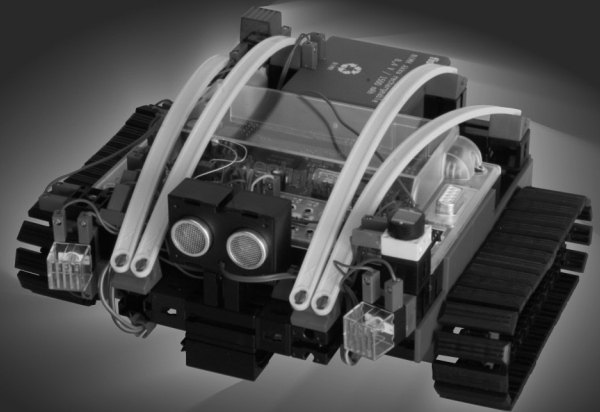
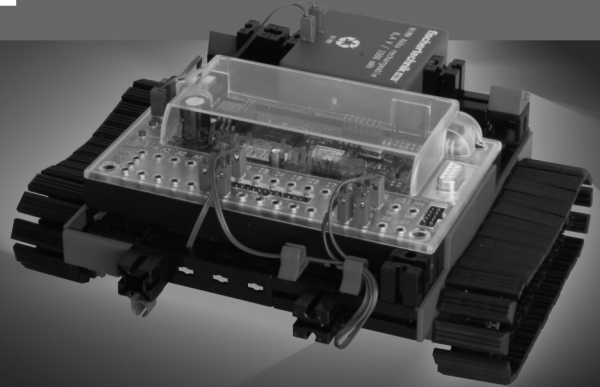


fischertechnik 

fischertechnik GmbH
Weinhalde 14-18
72178 Waldachtal
Germany

Phone +49 74 43/12-43 69
Fax +49 74 43/12-45 91

info@fischertechnik.de
www.fischertechnik.de



ROBO EXPLORER

6 MODELS