

6

Experimente mit den Infrarotbauteilen des EV3-Sets

In diesem Kapitel lernst du die Infrarotfernsteuerung und den Infrarotsensor kennen. Der Sensor kann nicht nur den Abstand von Objekten messen, sondern auch Signale der Fernsteuerung empfangen. Damit kannst du deinem Roboter Befehle senden ähnlich wie mit der Fernbedienung eines Fernsehers. Der IR-Sensor kann auch die Entfernung und die Richtung bestimmen, in der sich die Fernsteuerung befindet. Mit dieser tollen Funktion kannst du viele lustige Dinge mit deinen Robotern machen, du kannst beispielsweise mit ihnen Fangen spielen, sie einer Beute nachjagen lassen oder dafür sorgen, dass sie einen Ausgangspunkt wiederfinden.

Die IR-Fernsteuerung

Die IR-Fernsteuerung wird durch zwei AAA-Batterien mit Strom versorgt. An den Seiten weist sie Löcher auf, sodass du sie in Lego-Technic-Modelle einbauen kannst. Wie du in Abbildung 6-1 siehst, verfügt sie über vier kleine Tasten (Ziffer 1 bis 4), eine große Taste (9) und einen roten Kanalwähler (12), mit dem du zwischen vier Kanälen umschalten kannst. Dadurch ist es möglich, bis zu vier IR-Fernsteuerungen gleichzeitig zu verwenden. Eine auf rotem Kunststoff aufgeprägte Nummer, die du durch das kleine, runde Fenster siehst, gibt den aktuellen Kanal an. (Solange die einzelnen Fernsteuerungen auf unterschiedliche Kanäle eingestellt sind, stören sich ihre Signale gegenseitig nicht.)

Die große Taste (9) dient zum Umschalten in den Signalmodus, in dem das Gerät ein Dauersignal sendet, bis eine der anderen Tasten gedrückt wird oder eine Stunde vergangen ist. Der IR-Sensor kann die Nähe und die Richtung bestimmen, in der sich die Fernsteuerung im Signalmodus von ihm befindet. Damit ist es möglich, den Roboter einer mobilen Fernsteuerung folgen zu lassen oder die Entfernung und Richtung zu einer stationären Fernsteuerung zu ermitteln.

Die vier kleinen Tasten (1 bis 4) senden über zwei an der Vorderseite der Fernsteuerung angebrachte Leuchtdioden (LEDs) Befehle an den IR-Sensor (wie in der folgenden Liste aufgeführt). (Das Kunststoffgehäuse, in dem die LEDs untergebracht sind, ist ein Dunkelblaufilter, der nur Infrarotlicht durchlässt.)

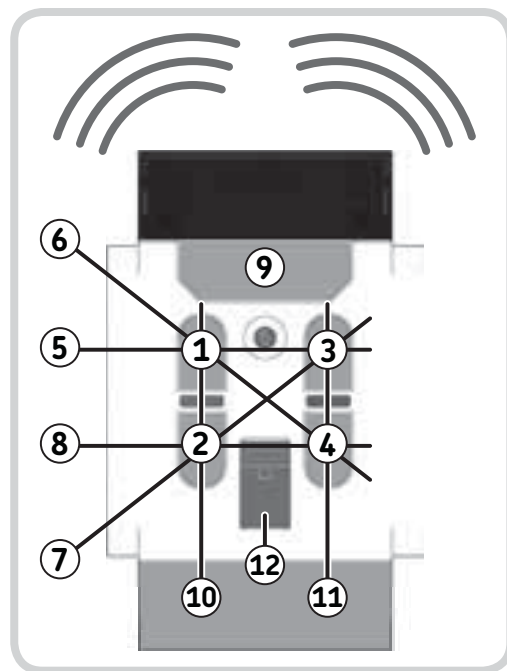


Abbildung 6-1: Die IR-Fernsteuerung

0	Keine Taste ist gedrückt, der Signalmodus ist ausgeschaltet
1	Taste 1
2	Taste 2
3	Taste 3
4	Taste 4
5	Taste 1 und 3
6	Taste 1 und 4
7	Taste 2 und 3
8	Taste 2 und 4
9	Signalmodus ist eingeschaltet
10	Taste 1 und 2
11	Taste 3 und 4

Die IR-Fernsteuerung als Fernbedienung verwenden

Sehen wir uns nun an, wie du die IR-Fernsteuerung als einfache Fernbedienung für deinen Roboter verwendest, und zwar nur mit der Stein-Anwendung **Infrarotsteuerung** (also ohne Programmierung – zumindest vorläufig!). Öffne im Menü auf dem EV3-Stein das Register **Stein-Anwendungen** (das dritte von links) und starte **Infrarotsteuerung**. Dadurch wird der Bildschirm aus Abbildung 6-2(a) angezeigt.

Wie du in Abbildung 6-2 siehst, kannst du zwischen zwei Modi wählen. In dem Modus aus Abbildung 6-2(a) steuerst du die Motoren über die Kanäle 1 und 2, im Modus aus Abbildung 6-2(b) über die Kanäle 3 und 4. Zum Umschalten zwischen den Betriebsarten drückst du die Taste **Eingabe** auf dem EV3-Stein.

Wenn du den Kanalwähler auf 1 gestellt hast (Ziffer 12 in Abbildung 6-1), kannst du im ersten Modus einen an Ausgang B angeschlossenen Motor über die Tasten 1 (vorwärts) und 2 (rückwärts) steuern und einen Motor an Ausgang C über Taste 3 (vorwärts) und 4 (rückwärts). Gleichzeitig kannst du mit einer weiteren Fernsteuerung auf Kanal 2 Motoren an den Ausgängen A und D steuern. Der zweite Modus funktioniert auf die gleiche Weise, empfängt aber Befehle von Fernsteuerungen, die auf Kanal 3 und 4 eingestellt sind.

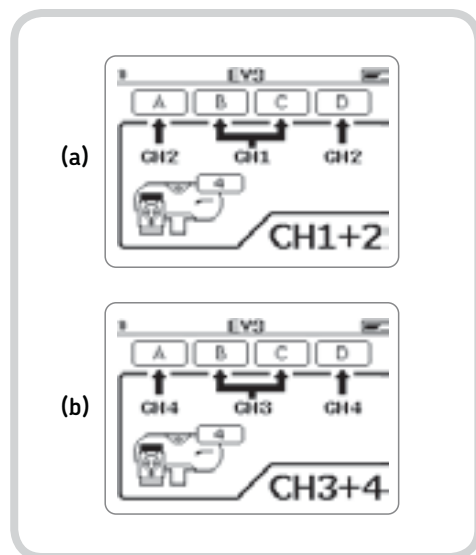


Abbildung 6-2: Die Stein-Anwendung Infrarotsteuerung. Um zwischen den Kanälen (CH für Channel) 1 und 2 (a) und 3 und 4 (b) umzuschalten, drückst du die Eingabe-Taste auf dem EV3-Stein.

Mit der Anwendung **Infrarotsteuerung** ist es einfach, einen Roboter mit Rädern wie den ROV3R fernzusteuern. Sie erweist sich auch als sehr hilfreich, wenn du einen motorgetriebenen Mechanismus baust, da du damit den Mechanismus testen kannst, ohne ein Programm dafür schreiben zu müssen.

Um die Fernsteuerung auszuprobieren, baust du den ROV3R in einer der Varianten, die in Kapitel 2 vorgestellt wurden, und schließt die Motoren an die Ausgänge B und C an. Starte dann die Anwendung **Infrarotsteuerung** auf dem EV3-Stein und wähle auf der IR-Fernsteuerung Kanal 1. Jetzt kannst du den ROV3R mit den kleinen Tasten dirigieren. Tabelle 6-1 zeigt, was die einzelnen Tasten bewirken.

Tabelle 6-1: Tasten zur Fernsteuerung eines Roboters mit Differenzialantrieb (z.B. der ROV3R)

Taste	Bewegung
1 und 3	Fährt vorwärts
2 und 4	Fährt rückwärts
1 und 4	Dreht sich nach rechts
2 und 3	Dreht sich nach links
1	Wendet sich nach rechts, indem er sich um das rechte Rad dreht
2	Wendet sich beim Rückwärtsfahren nach links, indem er sich um das rechte Rad dreht
3	Wendet sich nach links, indem er sich um das linke Rad dreht
4	Wendet sich beim Rückwärtsfahren nach rechts, indem er sich um das linke Rad dreht

HINWEIS Sollte die Fernsteuerung nicht richtig funktionieren, musst du überprüfen, ob der IR-Sensor an Eingang 4 angeschlossen und die Anwendung **Infrarotsteuerung** auf den richtigen Modus eingestellt ist, um Befehle über Kanal 1 zu empfangen (oder Kanal 2, falls die Motoren an Ausgang A und D angeschlossen sind).

Siehst du? Jetzt kannst du die IR-Fernsteuerung als Fernbedienung für den ROV3R verwenden, ohne ein Programm dafür zu schreiben! Kettenfahrzeuge wie Panzer oder Bagger werden auf ähnliche Weise gelenkt: Der Fahrer bedient dort zwei Hebel, die jeweils den Motor für die Kette auf der zugehörigen Seite steuern, ebenso wie du auf der Fernsteuerung Taste 1 und 2 oder 3 und 4 drückst. Mit zusätzlichen Programmen kannst du mit der IR-Fernsteuerung auch Fahrzeuge mit einer anderen Form der Lenkung dirigieren. Das wirst du in Kapitel 12 noch am Beispiel des SUP3RCARS sehen.

Sensorblöcke und Datenleitungen verwenden

Ein Roboter verwendet die von seinen Sensoren gelieferten Daten, um seine Umgebung wahrzunehmen. In den Kapiteln 3, 4 und 5 haben wir Sensormesswerte mit Schwellenwerten verglichen, um Warte- oder Schalterblöcke auszulösen. Für den direkten Zugriff auf Sensormesswerte stehen Sensorblöcke zur Verfügung (zu finden in der Palette mit dem gelben Register).

Jeder Sensorblock weist verschiedene Modi für unterschiedliche Funktionen auf. Im Messmodus stellt ein Sensorblock die Messwerte als Zahlenwerte für andere Blöcke zur Verfügung. Im Vergleichsmodus werden die Messwerte mit Schwellenwerten verglichen, um einen logischen Wert zu erhalten (siehe »Was sind Datentypen?« auf Seite 89). Manche Sensorblöcke (wie der Motorumdrehungs- und der Zeitgeberblock) haben auch einen Modus zum Zurücksetzen, durch den die gemessenen Werte wieder auf 0 gestellt werden.

Der Block für den Infrarotsensor verfügt über einen Nahemodus (in dem der Abstand zum nächsten Objekt gemessen wird), einen Fernsteuerungsmodus (in dem der Sensor Befehle von einer IR-Fernsteuerung empfängt) und einen Signalmodus (in dem der Roboter die Nähe und Richtung zur Fernsteuerung ermittelt).

Wir wollen nun Daten vom Infrarotsensorblock abrufen und verwenden. Baue dazu den ROV3R mit IR-Sensor (siehe S. 31) und stelle das Programm aus Abbildung 6-3 zusammen, indem du die Blöcke nach dem in Kapitel 5 beschriebenen Verfahren hinzufügst. Den IR-Sensorblock stellst du auf den Modus **Messen ▶ Nähe**, den Bewegungslenkungsblock auf **An**. Das Grundprinzip dieses einfachen Programms besteht darin, den vom IR-Sensor gelieferten Nähwert als Eingang für die Lenkung an den Bewegungslenkungsblock zu senden. Der ROV3R bewegt sich also in Abhängigkeit zum Abstand, den der Sensor zu einem Objekt festgestellt hat. Dadurch dreht sich der ROV3R auf der Stelle, bis du deine Hand vor seinen IR-Sensor hältst. Anschließend folgt er deiner Hand, bis du sie wieder wegziehst.

Um die Sensorausgabe an den Lenkungseingang zu senden, musst du eine *Datenleitung* einrichten. (Datenleitungen übertragen

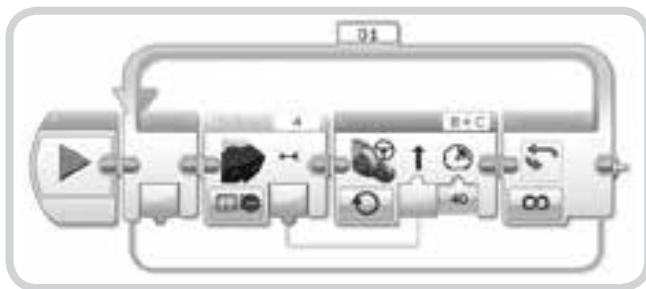


Abbildung 6-3: Dieses Programm sorgt dafür, dass der Roboter in Abhängigkeit von dem Abstand gesteuert wird, die der IR-Sensor misst.

Werte von einem Block zum anderen.) Dazu klickst du mit der Maus auf den *Ausgang des Sensorblocks*, hältst die Maustaste gedrückt und ziehst die Leitung von links nach rechts zum Eingang **Lenkung** des Bewegungsblocks (siehe Abbildung 6-4). Wenn du mit dem Mauszeiger über einen Ausgang fährst, verwandelt er sich in eine Garnspule.

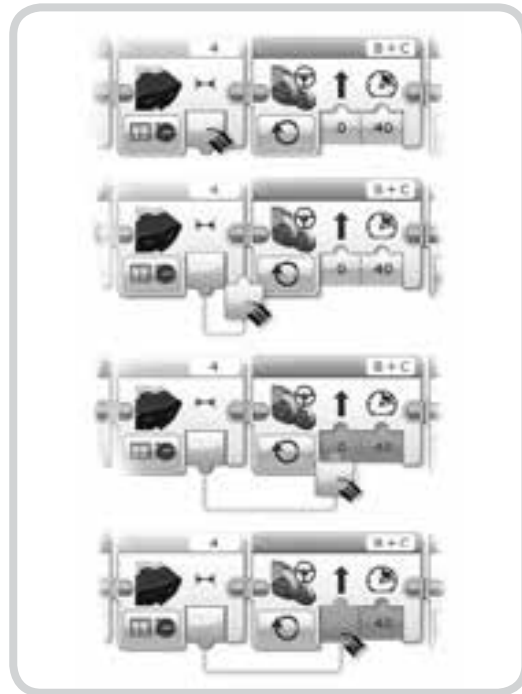


Abbildung 6-4: Um eine Datenleitung hinzuzufügen, musst du auf einen Blockausgang klicken. Am Ende der Leitung ist ein Stecker dargestellt. Wenn du ihn zum Eingang eines anderen Blocks ziehst, wird auf dem Bildschirm automatisch eine Datenleitung aufgespannt.

DATENLEITUNGEN ENTWIRREN

Um eine Datenleitung zu entfernen, musst du auf ihr Ende klicken und sie vom Eingang wegziehen (also praktisch die letzten beiden Schritte in Abbildung 6-4 umkehren). Willst du eine Datenleitung verschieben, klickst du darauf und ziehst sie an den gewünschten Ort. Sollten die Datenleitungen unübersichtlich verschlungen sein, doppelklickst du auf sie, damit die EV3-Software sie entwirrt und neu zeichnet.

Denke daran, dass der Block, der den Ausgangswert liefert, vor dem Block stehen muss, der den Wert in seinem Eingang empfängt. Die Blöcke werden dabei von links nach rechts verarbeitet: Der Ausgangspunkt, an dem die Datenleitung beginnt, muss also links von dem Eingangsblock stehen, an dem sie endet. Allerdings muss eine Datenleitung nicht unmittelbar an den benachbarten Block angeschlossen werden, sondern kann auch mehrere Blöcke überspannen.

Klickst du auf einen Ausgang, so erscheint dort ein Leitungsstecker. Wenn du die Leitung in die Nähe der Eingänge des Zielblocks ziehst, werden alle Eingänge blau hervorgehoben, die diese Art von Eingaben aufnehmen können. Platziere den Stecker auf dem gewünschten Eingang und lass die Maustaste los.

Lade das Programm aus Abbildung 6-3 herunter und führe es aus. Was geschieht? Der ROV3R sollte sich auf der Stelle drehen, bis du deine Hand in die Nähe des Sensors hältst. Dann fährt er auf deine Hand zu und folgt ihr, wenn du sie langsam zurückziehst.

Wie funktioniert das? Wenn der IR-Sensor einen großen Abstand misst (da kein Objekt in der Nähe ist), sendet der Ausgang **Nähe** einen hohen Wert von etwa 80 oder 90 %. Sobald du deine Hand vor den Sensor hältst, geht der Nähewert in Richtung von 0 %. Der Nähewert wird über die Datenleitung in den Eingang **Lenkung** des Bewegungslenkungsblocks gespeist, der Werte zwischen -100 und 100 annimmt (Prozentwerte für die Lenkung). Bei einem hohen Wert rotiert der Roboter um die eigene Achse, bei einem niedrigen fährt er fast geradeaus.

EXPERIMENT 6-1

Was macht das Programm aus Abbildung 6-3, wenn du die Datenleitung an den Eingang **Leistung** anschließt und den Eingang **Lenkung** auf 0 setzt?

Fehlersuche mit der EV3-Software

In Kapitel 5 hast du gesehen, wie du die Sensorwerte auf dem Register **Anschlussansicht** der Hardwareseite auch während der Ausführung des Programms ablesen kannst. Mit der EV3-Software kannst du aber noch mehr tun!

Da Datenleitungen Daten übertragen, kannst du den aktuellen Messwert des Sensors auf der Leitung anzeigen lassen. Wenn du mit dem Mauszeiger über die Leitung fährst, wird ein kleines Fenster mit dem derzeitigen Wert eingeblendet, wie du in Abbildung 6-5 siehst. Die Zahl in diesem Fenster ändert sich ständig, da sich die Blöcke in einer Endlosschleife befinden und daher immer wieder sehr schnell hintereinander ausgeführt werden. Diese Anzeige funktioniert aber nur, wenn du das Programm über die Hardwareseiten-Steuerung in der EV3-Software ausführst (siehe Abbildung 5-3 auf S. 71), nicht, wenn du es vom Menü auf dem EV3-Stein aus startest (auch wenn der Stein mit der EV3-Software verbunden ist).

In Abbildung 6-5 siehst du, dass die Kopfleiste jedes Blocks diagonale Streifen aufweist (in der Software sind sie sogar noch animiert). Dies bedeutet, dass die Blöcke zurzeit ausgeführt werden.

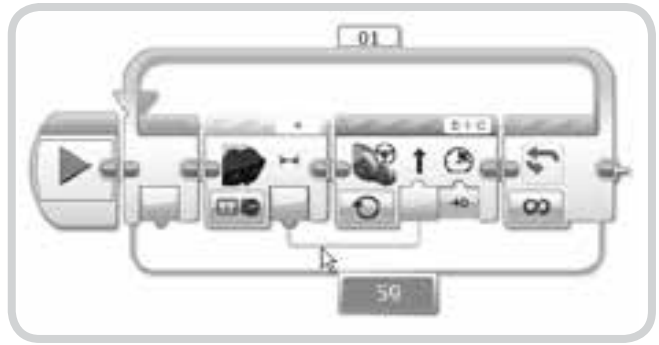


Abbildung 6-5: Wenn du mit dem Mauszeiger über eine Datenleitung fährst, wird ein Fenster mit dem Wert eingeblendet, der gerade über die Leitung übertragen wird. Die zurzeit ausgeführten Blöcke sind durch animierte Schrägstreifen gekennzeichnet.

Diese Darstellung des Ausführungszustands und die Echtzeit-Datenanzeige auf den Datenleitungen helfen dir beim *Debuggen* des Programms. (Das bedeutet, Fehler – oder *Bugs*, wie der Programmierer sagt – zu finden und zu beheben.)

Diese Anzeigen kannst du beispielsweise nutzen, um zu überprüfen, ob eine Datenleitung die erwarteten Werte überträgt oder ob das Programm an einem Warteblock hängen geblieben ist.

HINWEIS Woher die Begriffe *Bug* und *Debugging* stammen, ist unklar und umstritten. Es gibt die Legende, nach der diese Bezeichnungen auf die 1950er-Jahre zurückgehen, in denen Computer noch so groß wie begehbare Wandschränke waren. Damals soll es manchmal Insekten (engl. *bugs*!), wie Motten und Kakerlaken, gelungen sein, in die riesigen relaisgesteuerten Rechner hineinzukriechen, was zu elektrischen und mechanischen Problemen führte. Die Ingenieure hätten also buchstäblich *Bugs* entfernen müssen, um die Computer wieder zum Funktionieren zu bringen!

Datenanzeige mit dem Textblock

Als Nächstes werden wir dem Programm aus Abbildung 6-5 noch weitere Blöcke hinzufügen, um die Werte des IR-Sensors auf dem Bildschirm des EV3-Steins anzuzeigen, während der Roboter einer Hand folgt. Ergänze das Programm nach dem Muster aus Abbildung 6-6 um einen Textblock (aus der roten Palette der **Datenoperationsblöcke**) und einen Anzeigeblock.

Textblöcke haben nur einen einzigen Modus, nämlich **Aneinanderhängen**, der die Textstrings aus den Eingängen **A**, **B** und **C** kombiniert. Ein *String* ist nichts anderes als ein Stück Text, der aus einer beliebigen Kombination von Buchstaben, Ziffern, Leerzeichen und Symbolen wie `!"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~` bestehen kann.

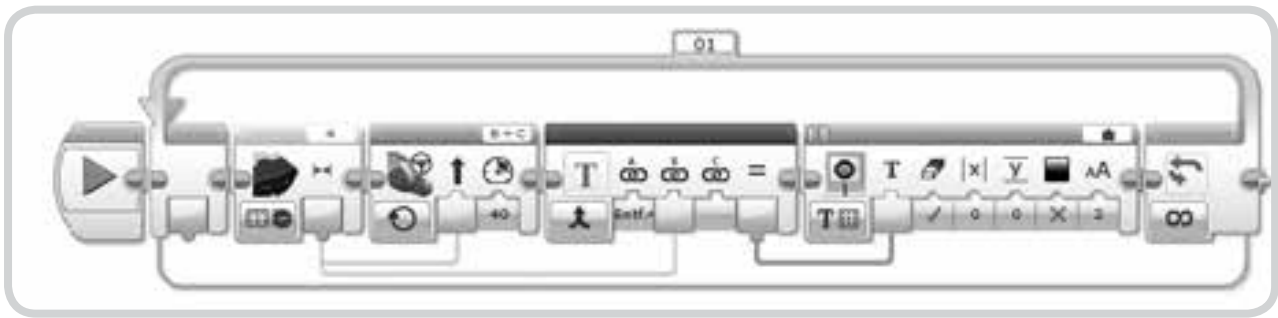


Abbildung 6-6: Mit einem Textblock kannst du auf dem Bildschirm des EV3-Steins aussagekräftige Meldungen anzeigen lassen. Dieses Programm ähnelt dem aus Abbildung 6-3.

Um Zeichen in einen Textblock einzuspeisen, tippt du sie entweder in eines seiner Eingabefelder ein oder schließt eine Datenleitung von einem anderen Block an einen seiner Eingänge an.

Wenn wir den Ausgang des IR-Sensors direkt an den Texteingang des Anzeigeblocks angeschlossen hätten, wären die Zahlenwerte automatisch in Text umgewandelt worden. Allerdings ist es bei der Anzeige von vielen Zahlenwerten nicht unbedingt klar, was sie bedeuten. Mit dem Textblock dagegen kannst du für eine aussagekräftigere Anzeige sorgen, indem du die Zahlenwerte mit Text ergänzt. Beispielsweise kannst du einen String wie **Abstand beträgt 20 %** oder **Entfernung = 40** erstellen, wobei die Werte **20** und **40** Sensormesswerte sind, die über die Datenleitung eingehen.

Probieren wir das gleich aus! Gib **Entf. =** in das Textfeld **A** ein (mit einem Leerzeichen hinter dem Gleichheitszeichen) und stell den Anzeigeblock auf den Modus **Text ▶ Raster** ein. Wähle dann **Per Leitung übertragen** anstelle des statischen Textes aus, indem du auf das Textfeld in der Kopfleiste klickst (siehe Abbildung 6-7). Der Anzeigeblock stellt jetzt auf dem Bildschirm eine neue Texteingabe dar, deren variable Textdaten von dir bereitgestellt werden. Setze den Eingang **Bildschirm löschen** auf **Wahr** (was durch ein Häkchen unter dem Radiogummisymbol angezeigt wird), damit der Block den Bildschirm bei jeder Ausführung löscht.

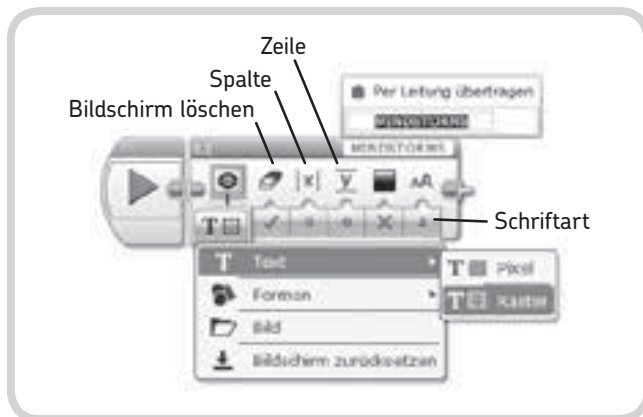


Abbildung 6-7: Richte den Anzeigeblock so ein, dass der Text aus einer Datenleitung übernommen und in einem Raster dargestellt wird.

Im Modus **Text ▶ Raster** wird der Text in einem Raster aus Zeilen (Eingang **Y**) und Spalten (Eingang **X**) angezeigt. Bei der normalen Schrift (Schriftart **0**) und der Fettschrift (Schriftart **1**) nimmt eine Rasterzelle jeweils ein Zeichen auf, bei der großen Schriftart (**2**) belegt jedes Zeichen zwei Zeilen und zwei Spalten.

Ziehe jetzt eine neue Datenleitung vom Ausgang des IR-Sensors zum zweiten Eingang des Textblocks und eine weitere vom Ausgang des Textblocks zum Texteingang des Anzeigeblocks. (Jetzt entspringen dem Ausgang also zwei Datenleitungen.) Lade das Programm herunter und führe es aus. Der Roboter verhält sich genauso wie vorher, zeigt auf seinem Bildschirm aber zusätzlich die Messdaten des IR-Sensors an.

WARNUNG Einem Ausgang können viele Datenleitungen entspringen, es ist aber nicht möglich, einen Eingang mit mehreren Datenleitungen zu verbinden.

Was sind Datentypen?

In den bisherigen Programmen hast du als Daten entweder Zahlen oder Text verwendet. Es gibt aber noch einen dritten Datentyp, nämlich *logische Werte* (also *wahr* oder *falsch*). Sie werden häufig eingesetzt, um das Ergebnis eines Vergleichs auszudrücken. Um besser zwischen den verschiedenen Datentypen unterscheiden zu können, weisen die Eingänge für *numerische Werte*, *Text*, *logische Werte*, *numerische Arrays* und *logische Arrays* unterschiedlich geformte Stecker auf, und auch die zugehörigen Datenleitungen haben jeweils eine andere Farbe (siehe Abbildung 6-8).

Folgende Steckerformen und Farben werden verwendet:

- * Numerische Ein-/Ausgänge sind rund geformt, die Datenleitungen sind gelb.
- * Logische Ein-/Ausgänge sind dreieckig geformt, die Datenleitungen sind grün.
- * Ein-/Ausgänge für Text sind viereckig geformt, die Datenleitungen sind orangefarben.
- * Ein-/Ausgänge für numerische Arrays weisen eine doppelte Rundung auf, die Datenleitungen sind dick und gelb.
- * Ein-/Ausgänge für logische Arrays zeigen zwei Dreiecke, die Datenleitungen sind dick und grün.