LMZ LANDESMEDIENZENTRUM BADEN-WÜRTTEMBERG

BUILD & CODE AUFGABENKARTEN

Mit dem Coding Max-Set vertiefen die Lernenden ihr Wissen über Computational Thinking und erweitern ihre Fähigkeiten im Umgang mit dem Set beim Programmieren verschiedener Konstruktionen, wie beispielsweise einer Ampelanlage.

Die Aufgabenkarten sind in verschiedene Schwierigkeitsstufen unterteilt und bieten zusätzlich Differenzierungsmöglichkeiten, indem die Modelle weiterentwickelt werden können. Dadurch lassen sich die Aufgabenkarten flexibel im Unterricht einsetzen.

Neben den technischen Fähigkeiten werden durch die kreativen Aufgaben auch wichtige kognitive Kompetenzen geschult, wie logisches Denken, Problemlösestrategien und algorithmisches Verständnis. Gleichzeitig stärkt die gemeinsame Arbeit im Team unter anderem die Kommunikations- und Kooperationsfähigkeit.

Gebaut wird im Zweierteam.

Als Material erhält jedes Team ein Coding Max-Set.

Jedes Team benötigt ein digitales Gerät (Tablet oder Computer) mit der App „STEM Suite".

Voraussetzung sind die Einstiegsmodelle der App „STEM Suite".

www.lmz-bw.de

ÜBERSICHT: AUFGABENKARTEN

1. Blinklicht Windkraftanlage

2. SOS-Lichtzeichen

3. Ampelanlage

4. Schranke

5. Quizshow

6. Hin- und Herbewegung

7. Ferngesteuerte Roxy

8. Autonome Roxy

9. Zufallsgenerator

10. Reaktionsspiel

IMPRESSUM

Herausgeber und Bezugsadresse

Landesmedienzentrum
Baden-Württemberg
Vertreten durch Direktor Michael Zieher
Rotenbergstraße 111
70190 Stuttgart

Telefon: 0711 4909-6399
E-Mail: comthink@lmz-bw.de

Dieses Materialpaket ist Teil des Projekts Computational Thinking im Sekundarbereich I

Projektleitung:
Juanjuan Jia und Silke Schick

Autoren:
Johannes Kempf und Timo Münzing

Fotos:
Marlene Feller

Redaktion:
ComThink-Team LMZ

Gestaltung:
Bianca Dreja Grafikdesign

Stuttgart, März 2025

Urheberrecht

Die Inhalte (Layout, Grafiken, Bilder etc.) sind urheberrechtlich geschützt. Sofern nicht anders vermerkt, stehen die Inhalte unter einer CC BY-NC-SA 4.0 Lizenz. Sämtliche Rechte an dieser Publikation liegen beim Landesmedienzentrum Baden-Württemberg (LMZ). Nichtkommerzielle Vervielfältigung und Verbreitung sind erlaubt unter Angabe des Herausgebers LMZ Baden-Württemberg und der Webseite www.lmz-bw.de. Urheberrechte Dritter sind zu beachten. Sie sind als solche kenntlich gemacht.

Bilder und Grafiken

Sämtliche in diesem Dokument verwendeten Fotos stammen vom Landesmedienzentrum Baden-Württemberg. Die Grafiken zu den Schwierigkeitsgraden wurden von Fischertechnik zur Verfügung gestellt.

Internetseiten dritter Anbieter / Links

Soweit Inhalte dieser Materialien auf externe Internetseiten verweisen, hat das LMZ auf den Inhalt dieser Seiten keinen Einfluss. Diese Internetseiten unterliegen der Haftung der jeweiligen Betreiber. Das LMZ hat bei der erstmaligen Verknüpfung der externen Links die fremden Inhalte daraufhin überprüft, ob etwaige Rechtsverstöße bestehen. Zu diesem Zeitpunkt waren keine Rechtsverstöße ersichtlich. Eine ständige inhaltliche Überprüfung der externen Links ist ohne konkrete Anhaltspunkte einer Rechtsverletzung nicht möglich. Bei Kenntnis von Rechtsverstößen werden derartige externe Links unverzüglich gelöscht.

BLINKLICHT EINER WINDKRAFTANLAGE

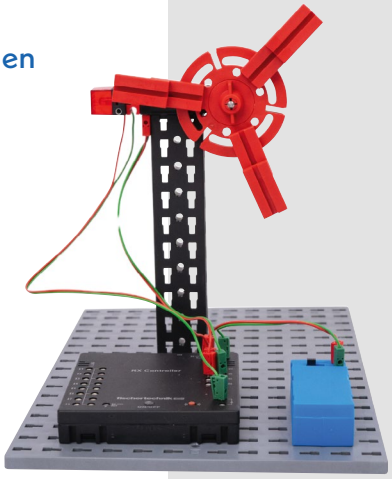
Schwierigkeitsgrad

Hohe Windkraftanlagen benötigen, um Flugzeuge zu warnen, ein rotes Blinklicht an ihrer Spitze. Das Licht blinkt in einem vorgegebenen **Ablauf: 1 Sekunde an, 0,5 Sekunden aus, 1 Sekunde an und dann 1,5 Sekunden aus.**

AUFGABE:

build: Baut einen Turm für eine Windkraftanlage. Bringt eine LED oben am Turm an.

code: Programmiert ein Blinklicht mit dem vorgegebenen Ablauf.



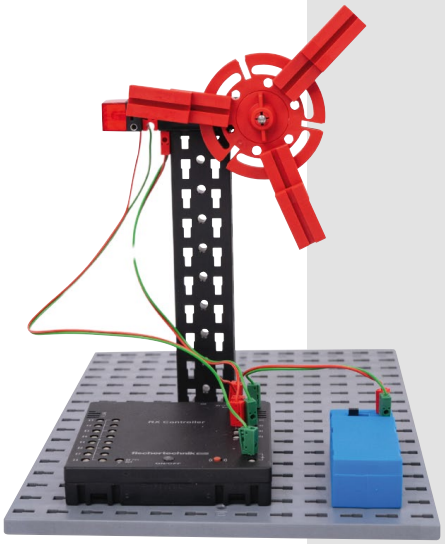
VORAUSSETZUNGEN:

Ihr könnt bereits…

- LEDs zum Leuchten bringen.

WIE KANN ES WEITERGEHEN?

Erweiterung Rotor:
Baut einen Rotor auf den Turm des Windrads und lasst ihn mit einem Motor drehen.



LÖSUNGSVORSCHLAG
„BLINKLICHT WINDKRAFTANLAGE“

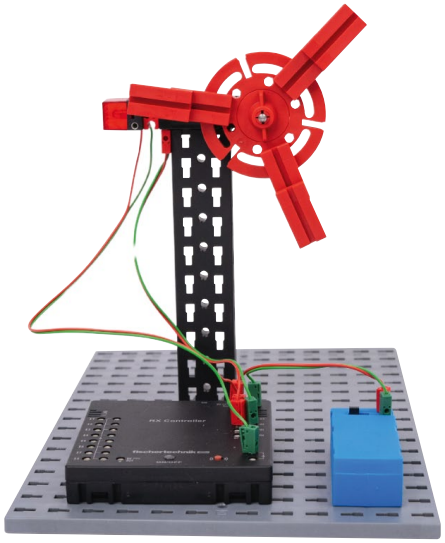
Hohe Windkraftanlagen benötigen, um Flugzeuge zu warnen, ein rotes Blinklicht an ihrer Spitze. Das Licht blinkt in einem vorgegebenen **Ablauf: 1 Sekunde an, 0,5 Sekunden aus, 1 Sekunde an und dann 1,5 Sekunden aus.**

AUFGABE:

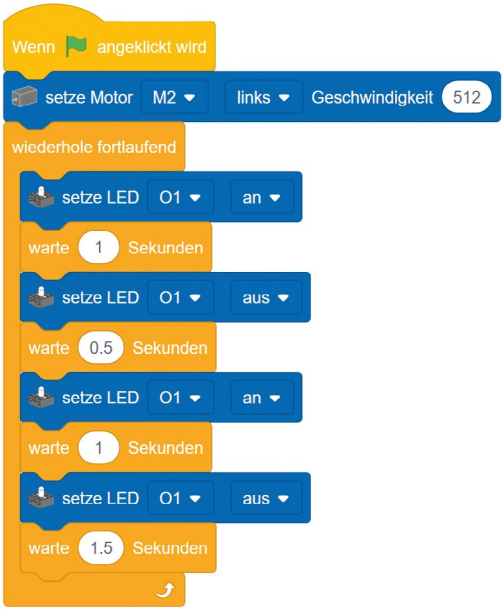
build: Baut einen Turm für eine Windkraftanlage. Bringt eine LED oben am Turm an.
code: Programmiert ein Blinklicht mit dem vorgegebenen Ablauf.

SCHALTUNG:

Ausgänge:
O1: LED 1



PROGRAMMCODE:



SOS-BLINKLICHT

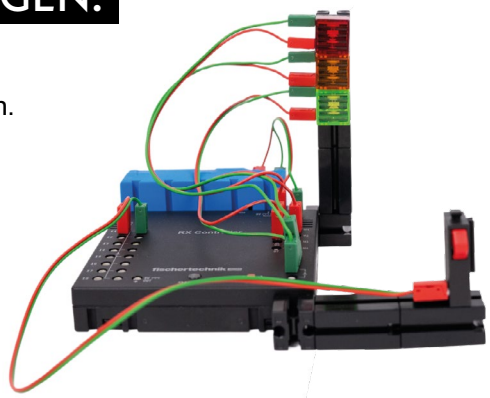
Morsen ist eine Methode zur Kommunikation, bei der Buchstaben durch kurze und lange Impulse (Punkte und Striche) dargestellt werden. **Das Notrufzeichen „SOS“ wird häufig als Lichtsignal verwendet, wobei das „S“ aus drei kurzen und das „O“ aus drei langen Blink-Impulsen besteht.**

AUFGABE:

build: Schließe eine LED an den RX Controller an.
code: Schreibt ein Programm, das die LED in der Zeichenfolge „SOS“ als Lichtsignal leuchten lässt.

VORAUSSETZUNGEN:

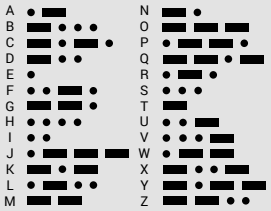
Ihr könnt bereits…
- LEDs zum Leuchten bringen.



Schwierigkeitsgrad

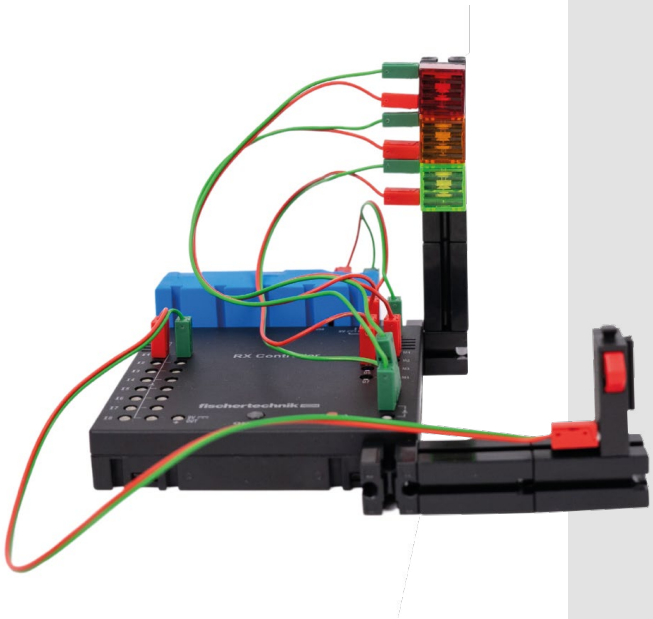
Hier kannst du „SOS“ als akustisches Signal hören

Morsecode



WIE KANN ES WEITERGEHEN?

Rätsel entwickeln:
Stellt weitere Wörter mit der LED im Morsecode dar und lasst die Wörter von anderen Personen erraten.



LÖSUNGSVORSCHLAG „SOS-BLINKLICHT“

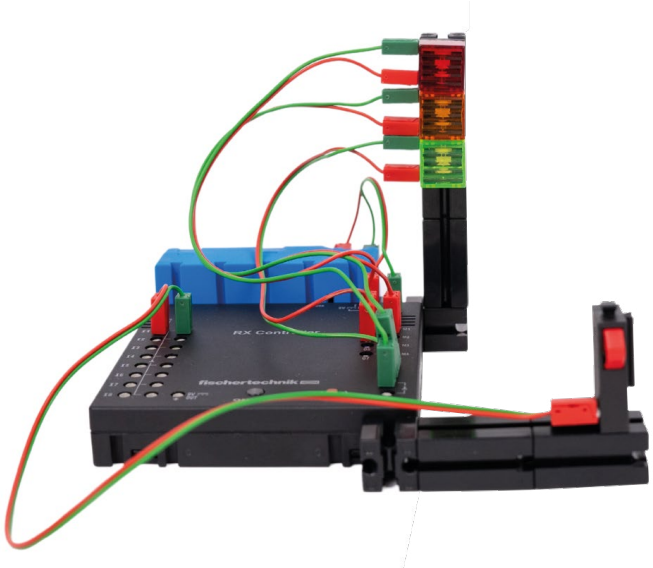
Morsen ist eine Methode zur Kommunikation, bei der Buchstaben durch kurze und lange Impulse (Punkte und Striche) dargestellt werden. **Das Notrufzeichen „SOS“ wird häufig als Lichtsignal verwendet, wobei das „S“ aus drei kurzen und das „O“ aus drei langen Blink-Impulsen besteht.**

AUFGABE:

build: Schließe eine LED an den RX Controller an.
code: Schreibt ein Programm, das die LED in der Zeichenfolge „SOS“ als Lichtsignal leuchten lässt.

SCHALTUNG:

Ausgänge:
O1: LED (rot)



PROGRAMMCODE:

AMPELANLAGE

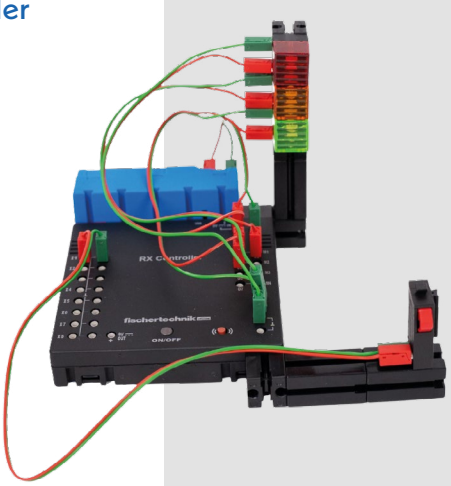
Ampeln steuern den Verkehr für Fahrzeuge und Fußgänger.

Schwierigkeitsgrad

AUFGABE:

build: Schließt für die Ampelanlage drei LEDs in den Farben rot, gelb und grün sowie einen Taster an den RX Controller an.

code: Durch Betätigen des Tasters soll die Ampel von rot für eine Sekunde auf gelb und anschließend für fünf Sekunden auf grün schalten. Danach soll sie wieder eine Sekunde gelb leuchten, bevor sie erneut rot anzeigt.



VORAUSSETZUNGEN:

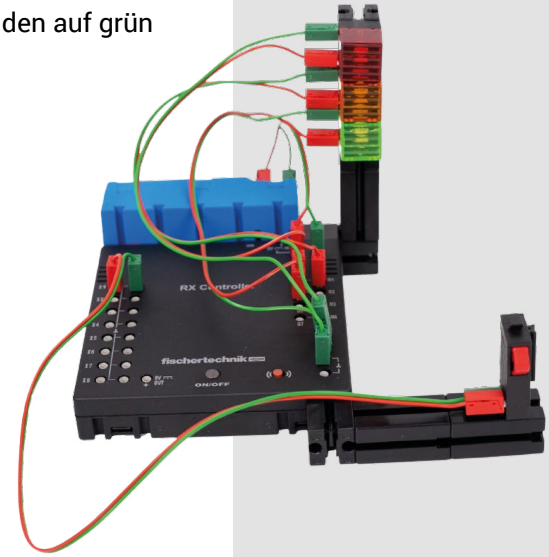
Ihr könnt bereits…

- eine LED an- und ausschalten.
- ein Programm auf einen Taster reagieren lassen.

WIE KANN ES WEITERGEHEN?

Erweiterung Schranke:
Ergänzt eine motorbetriebene Schranke. Sie soll bei rot geschlossen und bei grün geöffnet sein.

Erweiterung Gestensensor:
Durch Betätigen des Tasters soll die Ampel von rot für eine Sekunde auf gelb und anschließend für fünf Sekunden auf grün schalten.



LÖSUNGSVORSCHLAG „AMPELANLAGE“

Ampeln steuern den Verkehr für Fahrzeuge und Fußgänger.

AUFGABE:

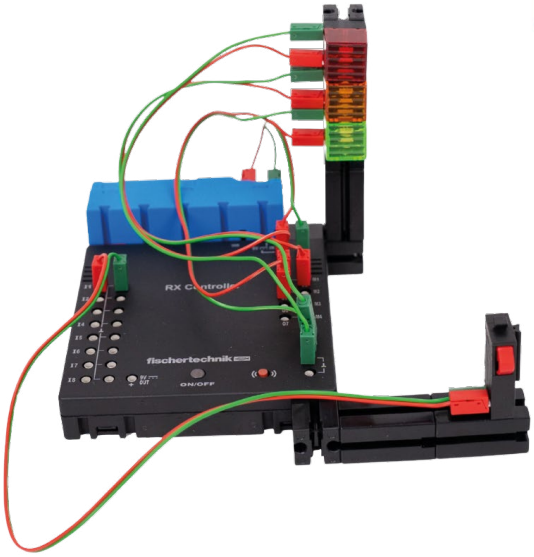
build: Schließt für die Ampelanlage drei LEDs in den Farben rot, gelb und grün sowie einen Taster an den RX Controller an.

code: Durch Betätigen des Tasters soll die Ampel von rot für eine Sekunde auf gelb und anschließend für fünf Sekunden auf grün schalten. Danach soll sie wieder eine Sekunde gelb leuchten, bevor sie erneut rot anzeigt.

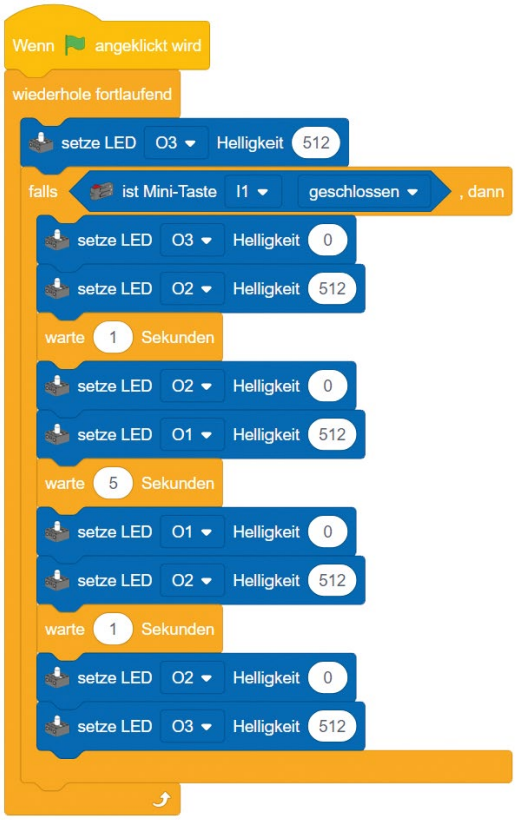
SCHALTUNG:

Eingänge:
I1: Taster

Ausgänge:
O1: LED (grün)
O2: LED (gelb)
O3: LED (rot)



PROGRAMMCODE:



SCHRANKE

Eine Schranke soll aufgehen, sobald sich ihr eine Person nähert. Danach soll die Schranke wieder geschlossen werden.

Schwierigkeitsgrad

AUFGABE:

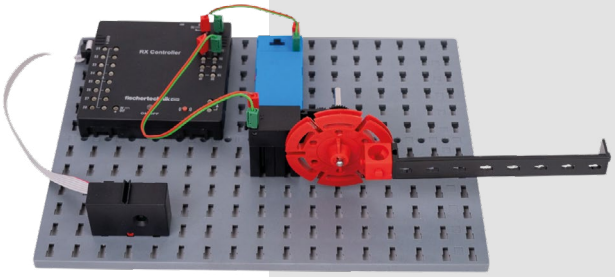
build: Baut eine Schranke aus einem Motor mit Getriebe. Schließt den Motor und den RGB-Gestensensor an den RX Controller an.

code: Programmiert die automatische Schranke.
Hinweis: Zur zuverlässigeren Entfernungsmessung sollte der RGB-Gestensensor an der Tischkante platziert werden.

VORAUSSETZUNGEN:

Ihr könnt bereits…

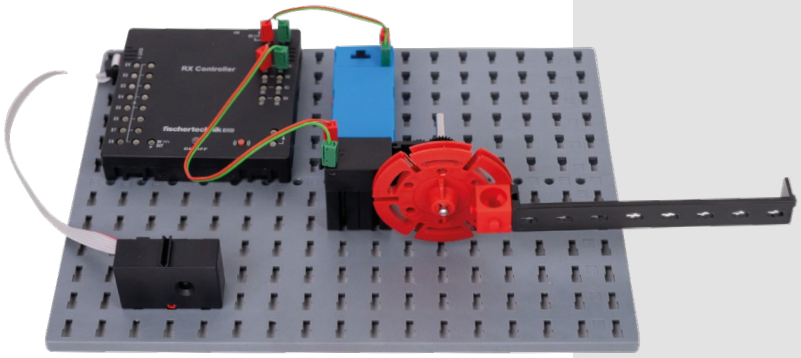
- den Gestensensor einsetzen.
- einen Motor drehen lassen.



WIE KANN ES WEITERGEHEN?

Einsatz von Gesten:
Verwendet zur Steuerung der Schranke Gesten.

Einsatz von Variablen:
Verwendet eine Variable, um den Status der Schranke zu speichern und den Motor entsprechend zu steuern, sodass die Schranke automatisch auf Personen reagiert.



LÖSUNGSVORSCHLAG „SCHRANKE“

Eine Schranke soll aufgehen, sobald sich ihr eine Person nähert. Danach soll die Schranke wieder geschlossen werden.

AUFGABE:

build: Baut eine Schranke aus einem Motor mit Getriebe. Schließt den Motor und den RGB-Gestensensor an den RX Controller an.

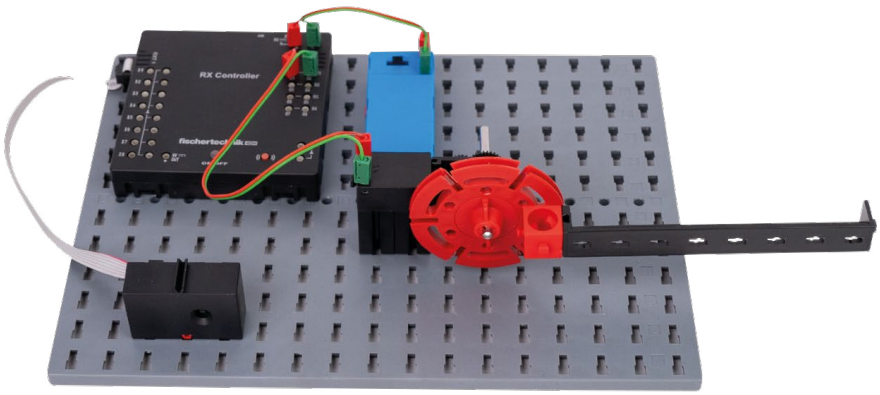
code: Programmiert die automatische Schranke.
Hinweis: Zur zuverlässigeren Entfernungsmessung sollte der RGB-Gestensensor an der Tischkante platziert werden.

PROGRAMMCODE:

SCHALTUNG:

Eingänge:
EXT 1: Gestensensor

Ausgänge:
M1: Motor



QUIZSHOW

In Quizshows entscheidet oft die Schnelligkeit, wer eine Frage beantworten darf. Hierzu bekommt jeder Kandidat einen Buzzer und ein Licht leuchtet bei derjenigen Person, die diesen als erstes drückt.

Schwierigkeitsgrad

AUFGABE:

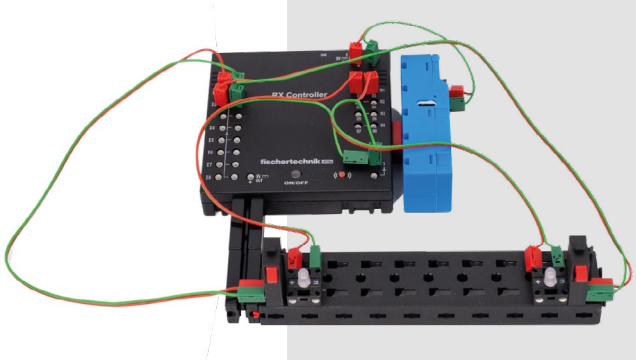
build: Schließt zwei Taster und zwei LEDs an den RX Controller an.

code: Erstellt ein Spiel für zwei Teilnehmende, bei dem der erste Tasterdruck registriert wird. Sobald eine Person den Taster drückt, leuchtet die LED dieser Person auf und der Taster der anderen Person wird ignoriert.

VORAUSSETZUNGEN:

Ihr könnt bereits…

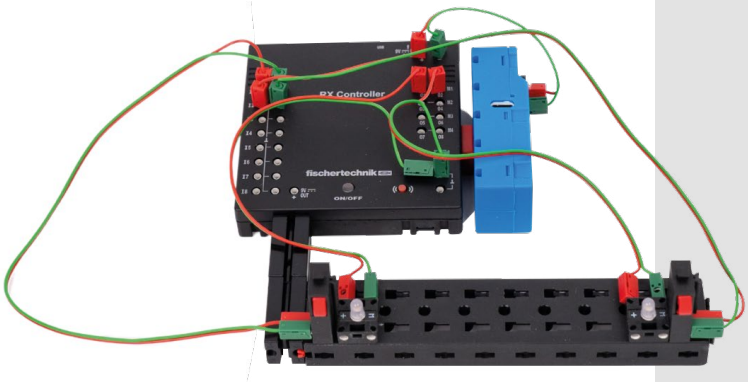
- LEDs zum Leuchten bringen.
- ein Programm auf einen Taster reagieren lassen.



WIE KANN ES WEITERGEHEN?

Quizfragen:
Überlegt euch Quizfragen zum bisher Erlernten und stellt sie einem anderen Team. Wer gewinnt das Quiz?

Erweiterung Schaltung:
Erweitert die Schaltung, indem ihr einen weiteren Taster und eine zusätzliche LED hinzufügt, sodass drei Teilnehmende gegeneinander spielen können.



LÖSUNGSVORSCHLAG „QUIZSHOW“

In Quizshows entscheidet oft die Schnelligkeit, wer eine Frage beantworten darf. Hierzu bekommt jeder Kandidat einen Buzzer und ein Licht leuchtet bei derjenigen Person, die diesen als erstes drückt.

AUFGABE:

build: Schließt zwei Taster und zwei LEDs an den RX Controller an.

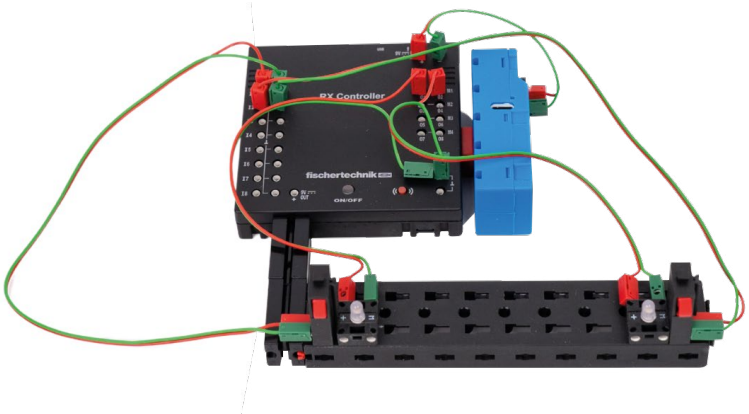
code: Erstellt ein Spiel für zwei Teilnehmende, bei dem der erste Tasterdruck registriert wird. Sobald eine Person den Taster drückt, leuchtet die LED dieser Person auf und der Taster der anderen Person wird ignoriert.

PROGRAMMCODE:

SCHALTUNG:

Eingänge:
I1: Taster 1
I2: Taster 2

Ausgänge:
O1: LED 1
O3: LED 2



HIN- UND HERBEWEGUNG

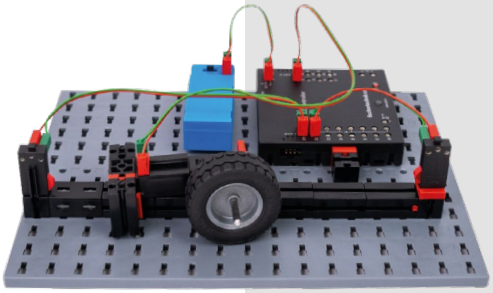
In vielen Unternehmen erledigen Maschinen Aufgaben automatisch, wie das Ein- und Ausräumen von Regalen oder der Transport von Waren.

Schwierigkeitsgrad

AUFGABE:

build: Baut eine Konstruktion, bei der etwas von links nach rechts fahren kann. Ergänzt jeweils am Ende der Fahrbahn einen Taster, der von einem Fahrzeug aktiviert werden kann. Das Bild kann euch eine erste Idee liefern.

code: Programmiert das Fahrzeug so, dass es von allein gegen die beiden Taster fahren kann und anschließend die Richtung wechselt.

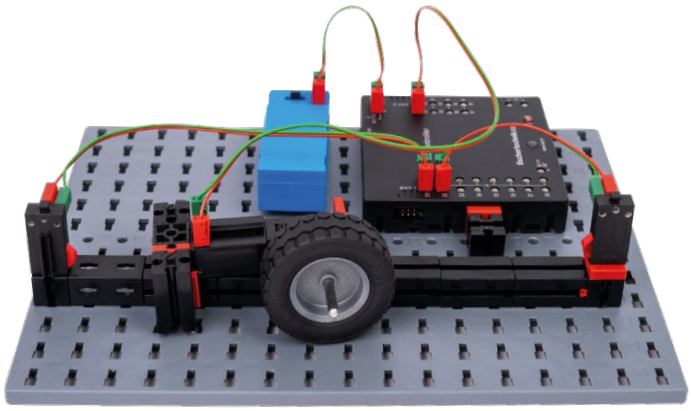


VORAUSSETZUNGEN:

Ihr könnt bereits…

- ein Programm auf einen Taster reagieren lassen.
- einen Motor drehen lassen.

COMTHINK



LMZ LANDESMEDIENZENTRUM BADEN-WÜRTTEMBERG

LÖSUNGSVORSCHLAG „HIN- UND HERBEWEGUNG“

In vielen Unternehmen erledigen Maschinen Aufgaben automatisch, wie das Ein- und Ausräumen von Regalen oder der Transport von Waren.

AUFGABE:

build: Baut eine Konstruktion, bei der etwas von links nach rechts fahren kann. Ergänzt jeweils am Ende der Fahrbahn einen Taster, der von einem Fahrzeug aktiviert werden kann. Das Bild kann euch eine erste Idee liefern.

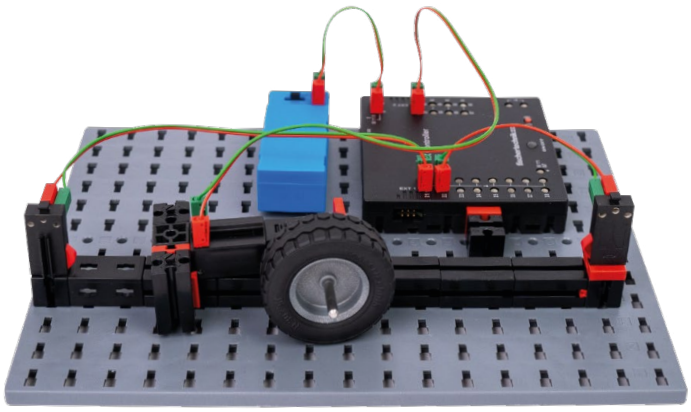
code: Programmiert das Fahrzeug so, dass es von allein gegen die beiden Taster fahren kann und anschließend die Richtung wechselt.

PROGRAMMCODE:

SCHALTUNG:

Eingänge:
I1: Taster 1
I2: Taster 2

Ausgänge:
M1: Motor 1



FERNGESTEUERTE ROXY

Ein ferngesteuerter Roboter kann in vielen Situationen nützlich sein. Ob in Katastrophengebieten zur Rettung von Menschen, im Restaurant, um das Essen an den Tisch zu fahren, oder in unzähligen anderen Situationen.

Schwierigkeitsgrad

AUFGABE:

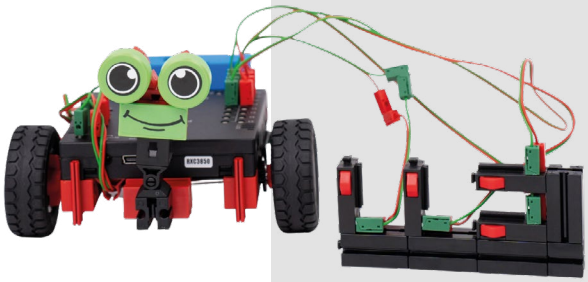
build: Baut Roxy nach Anleitung auf und ergänzt eine Fernsteuerung mit 4 Tastern (vor, zurück, rechts, links).

code: Programmiert die Taster so, dass ihr Roxy damit lenken kannst.

VORAUSSETZUNGEN:

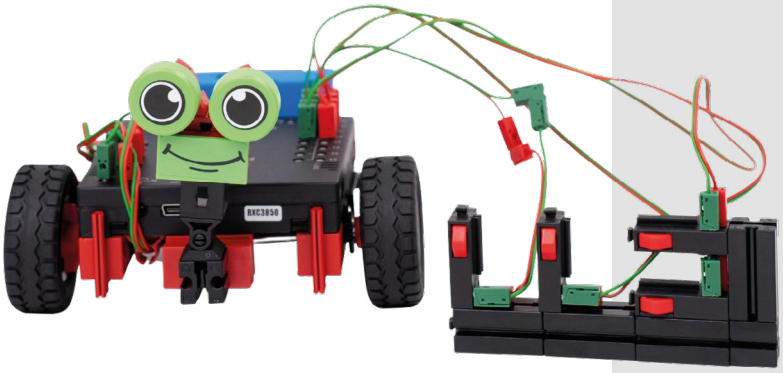
Ihr könnt bereits…

- ein Programm auf einen Taster reagieren lassen.
- einen Motor drehen lassen.



WIE KANN ES WEITERGEHEN?

Parcours:
Baut einen Hindernisparcours auf, durch den ihr Roxy lenken könnt. Wer schafft es am schnellsten durch den Parcours? Könnt ihr im Programmcode noch Anpassungen vornehmen, damit ihr Roxy besser lenken könnt?



LÖSUNGSVORSCHLAG „FERNGESTEUERTE ROXY“

Ein ferngesteuerter Roboter kann in vielen Situationen nützlich sein. Ob in Katastrophengebieten zur Rettung von Menschen, im Restaurant, um das Essen an den Tisch zu fahren, oder in unzähligen anderen Situationen.

AUFGABE:

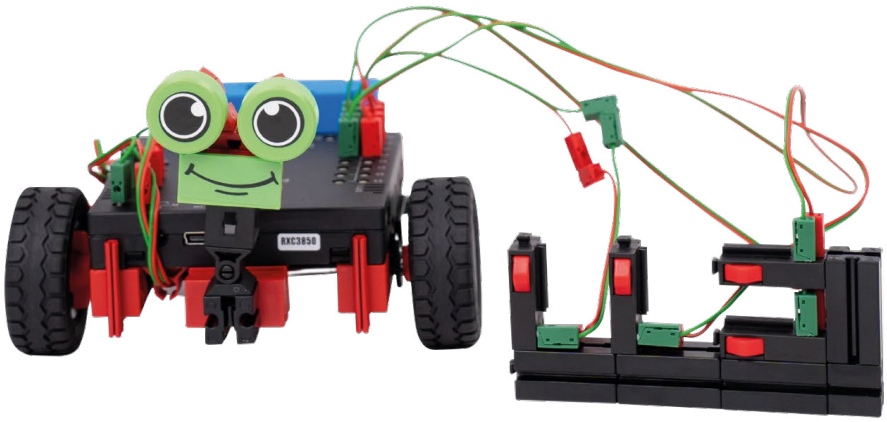
build: Baut Roxy nach Anleitung auf und ergänzt eine Fernsteuerung mit 4 Tastern (vor, zurück, rechts, links).
code: Programmiert die Taster so, dass ihr Roxy damit lenken könnt.

SCHALTUNG:

Eingänge:
I1: Taster 1
I2: Taster 2

Ausgänge:
M1: Motor 1
M2: Motor 2

PROGRAMMCODE:



AUTONOME ROXY

Autonome Fahrzeuge erfassen ihre Umgebung mit Sensoren und reagieren selbstständig auf Hindernisse. Solche Systeme finden sich in Staubsaugerrobotern, Rasenmähroboter und selbstfahrenden Autos.

Schwierigkeitsgrad

AUFGABE:

build: Baut Roxy nach Anleitung zusammen und befestigt, wie auf dem Foto zu sehen, an der Vorderseite den RGB-Gestensensor. Schließt den RGB-Gestensensor an den RX Controller an.

code: Programmiert den RGB-Gestensensor so, dass er als Entfernungsmesser Hindernisse erkennt. Bei einem auftretenden Hindernis soll Roxy dieses umfahren.

VORAUSSETZUNGEN:

Ihr könnt bereits…

- einen Motor drehen lassen.
- den RGB-Gestensensor einsetzen.

LMZ LANDESMEDIENZENTRUM BADEN-WÜRTTEMBERG

WIE KANN ES WEITERGEHEN?

Erweiterung Taster:
Ein Taster startet das Programm von Roxy. Beispielsweise beginnt die Fahrt erst nach einem Tasterdruck.

Erweiterung LEDs:
Verschiedene LEDs zeigen, was Roxy gerade macht. Beispielsweise leuchtet eine rote LED, wenn ein Hindernis erkannt wird.

Erweiterung Fahrfunktionen:
Verwendet den RGB-Gestensensor für weitere Fahrfunktionen.

LÖSUNGSVORSCHLAG „AUTONOME ROXY“

Autonome Fahrzeuge erfassen ihre Umgebung mit Sensoren und reagieren selbstständig auf Hindernisse. Solche Systeme finden sich in Staubsaugerrobotern, Rasenmährobotern und selbstfahrenden Autos.

AUFGABE:

build: Baut Roxy nach Anleitung zusammen und befestigt, wie auf dem Foto zu sehen, an der Vorderseite den RGB-Gestensensor. Schließt den RGB-Gestensensor an den RX Controller an.

code: Programmiert den RGB-Gestensensor so, dass er als Entfernungsmesser Hindernisse erkennt. Bei einem auftretenden Hindernis soll Roxy dieses umfahren.

SCHALTUNG:

Eingänge:
EXT 1: RGB-Gestensensor

Ausgänge:
M1: Motor 1
M2: Motor 2

PROGRAMMCODE:

ZUFALLSGENERATOR

Das Binärsystem ist ein Zahlensystem, bei dem mit den Ziffern 0 und 1 Dezimalzahlen dargestellt werden können. Es wird in der Informatik verwendet, um Informationen zu speichern und zu übertragen.

AUFGABE:

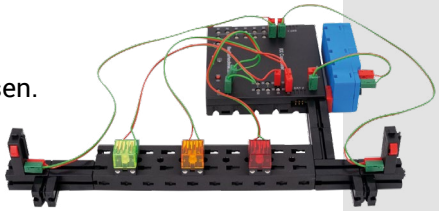
build: Schließt drei LEDs und einen Taster an den RX Controller an.

code: Mit den drei angeschlossenen LEDs können Zahlen im Binärsystem angezeigt werden. Eine leuchtende LED steht für die Zahl 1 und eine ausgeschaltete LED für die Zahl 0. Programmiert einen Zufallsgenerator, der beim Drücken des Tasters mithilfe der LEDs eine Zufallszahl zwischen 0 und 7 anzeigt.

VORAUSSETZUNGEN:

Ihr könnt bereits…

- eine LED an- und ausschalten.
- ein Programm auf einen Taster reagieren lassen.
- mit Variablen programmieren.



Schwierigkeitsgrad

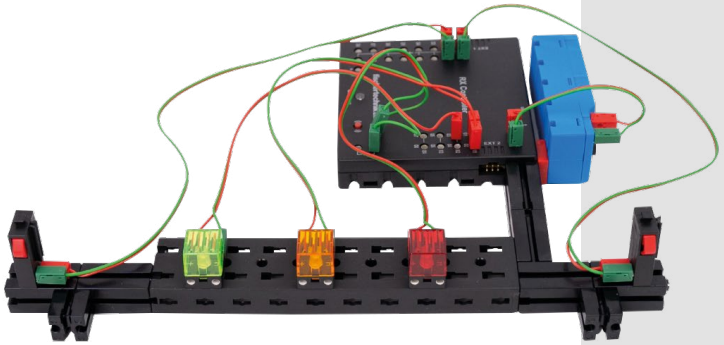
Tabelle Dezimal- und Binärzahlen

Dezimalzahl	Binärzahl
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

WIE KANN ES WEITERGEHEN?

Anwendung Ternärsystem:
Recherchiert, was das Ternärsystem ist.

Zufallsgenerator erstellen:
Die LEDs können auch nur schwach leuchten. Dadurch können die Ziffern 0 (LED ist aus), 1 (LED leuchtet leicht), 2 (LED leuchtet stark) dargestellt werden. Im Ternärsystem können mit den 3 LEDs dann Zahlen von 0 bis 17 dargestellt werden. Programmiert einen entsprechenden Zufallsgenerator.



LÖSUNGSVORSCHLAG „ZUFALLSGENERATOR“

Das Binärsystem ist ein Zahlensystem, bei dem mit den Ziffern 0 und 1 Dezimalzahlen dargestellt werden können. Es wird in der Informatik verwendet, um Informationen zu speichern und zu übertragen.

AUFGABE:

build: Schließt drei LEDs und einen Taster an den RX Controller an.

code: Mit den drei angeschlossenen LEDs können Zahlen im Binärsystem angezeigt werden. Eine leuchtende LED steht für die Zahl 1 und eine ausgeschaltete LED für die Zahl 0. Programmiert einen Zufallsgenerator, der beim Drücken des Tasters mithilfe der LEDs eine Zufallszahl zwischen 0 und 7 anzeigt.

SCHALTUNG:

Eingänge:
I1: Taster

Ausgänge:
O1: LED
O2: LED
O3: LED

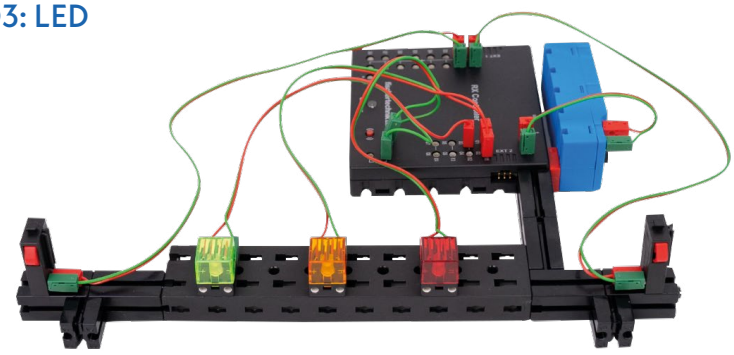


Tabelle Dezimal- und Binärzahlen

Dezimalzahl	Binärzahl
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

PROGRAMMCODE:

REAKTIONSSPIEL

Die Reaktionsfähigkeit ist wichtig, um beispielsweise im Sport schnell reagieren zu können. Baut ein Spiel, dass eure Reaktionsfähigkeit testet.

Schwierigkeitsgrad

AUFGABE:

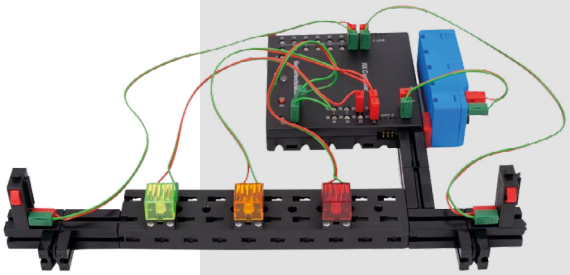
build: Schließt zwei Taster und drei LEDs an den RX Controller an.

code: Lasst die mittlere LED nach einer zufälligen Zeitspanne leuchten. Die Person, die als erste ihren Taster betätigt, bekommt einen Punkt und ihre LED geht kurz an. Anschließend beginnt der nächste Reaktionstest.

VORAUSSETZUNGEN:

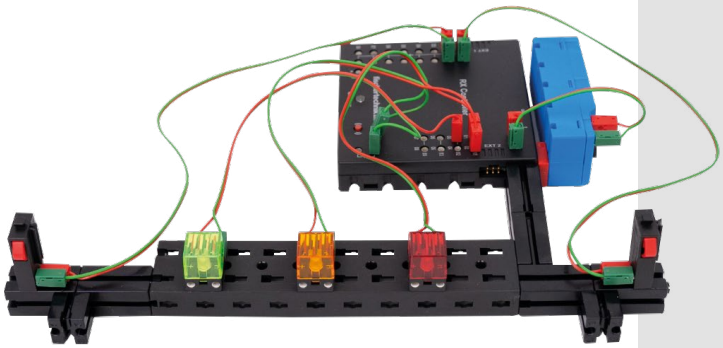
Ihr könnt bereits…

- eine LED an- und ausschalten.
- ein Programm auf einen Taster reagieren lassen..
- mit Variablen programmieren.
- sehr gut mit Bedingungen und Abfragen umgehen.



WIE KANN ES WEITERGEHEN?

Gewinneranzeige:
Sobald die erste Person fünf Mal gewonnen hat, soll die LED der Gewinnerin oder des Gewinners mehrmals blinken.



LÖSUNGSVORSCHLAG „REAKTIONSSPIEL“

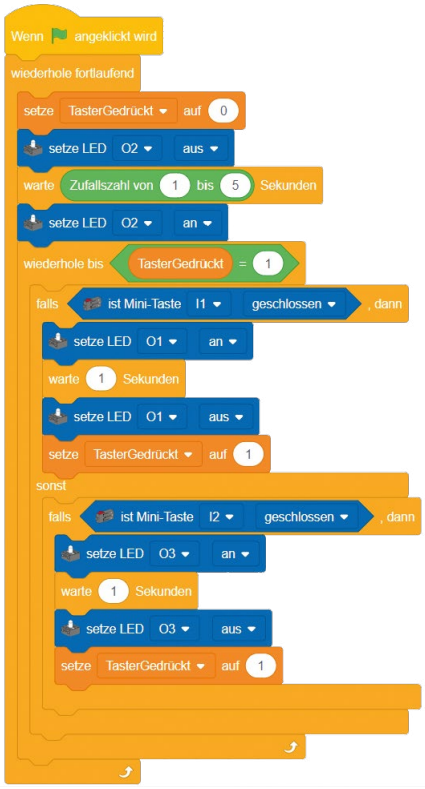
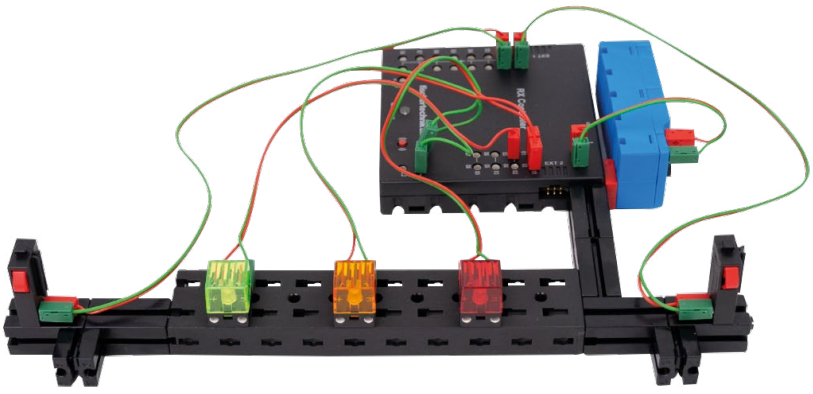
Die Reaktionsfähigkeit ist wichtig, um beispielsweise im Sport schnell reagieren zu können. Baut ein Spiel, dass eure Reaktionsfähigkeit testet.

AUFGABE:

build: Schließt zwei Taster und drei LEDs an den RX Controller an.
code: Lasst die mittlere LED nach einer zufälligen Zeitspanne leuchten. Die Person, die als erste ihren Taster betätigt, bekommt einen Punkt und ihre LED geht kurz an. Anschließend beginnt der nächste Reaktionstest.

SCHALTUNG:

Eingänge:
I1: Taster
I2: Taster

Ausgänge:
O1: LED (grün)
O2: LED (gelb)
O3: LED (rot)

PROGRAMMCODE: