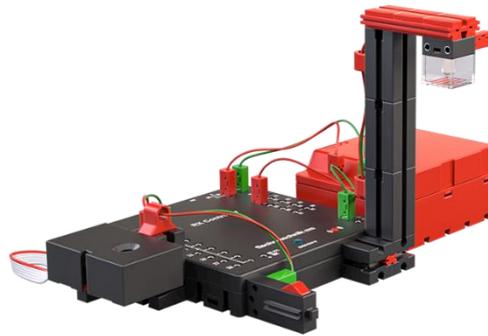


Dämmerungsschalter



Mit diesem Modell erarbeitest du dir weitere Grundlagen der Programmierung und der im Modell verbauten Elemente insbesondere des Gestensensors.

Im Modell werden folgende Sensoren und Aktoren verwendet:

Gestensensor	LED (Aktor)
	

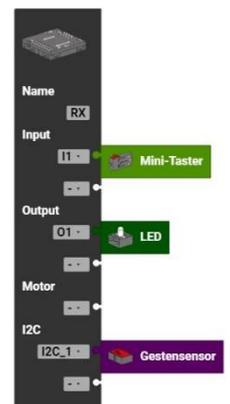
Die Erklärung zu den Bauteilen findest du auf der Startseite.

Das Modell „Dämmerungsschalter“ gliedert sich in 2 Programmieraufgaben:

Aufgabe 1 Twilight_switch_1.ft	<u>Programmierlevel 3</u> Ab einem bestimmten Dunkelheitswert schaltet die LED ein oder aus
Aufgabe 2 Twilight_switch_2.ft	<u>Programmierlevel 3</u> Ab einem bestimmten Dunkelheitswert schaltet die LED ein oder aus. Bei eingeschalteter LED soll diese im 1-Sekunden Rhythmus blinken. Mit dem Taster soll die LED bei Helligkeit eingeschaltet werden können.

Controllerkonfiguration

Bevor du mit der Aufgabe beginnst, musst du noch den Controller konfigurieren. Füge einen RX-Controller, einen Minitaster, eine LED und den Gestensensor aus den Blöcken „Eingang“, „Ausgang“ und „I2C“ ein.



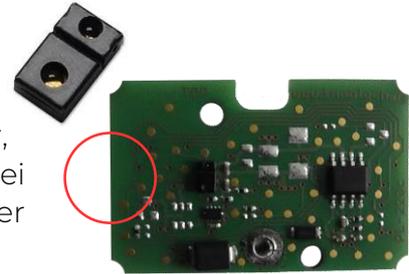
Wichtig: Schalte auf das Programmierlevel (Lernstufe) 3 um.

Aufgabe 1

Bevor du mit dem Programmieren beginnst, hier noch ein paar Informationen zum Baustein:

Gestensensor

Eigentlich ist der APDS-9960 Sensor ein kleiner, rechteckiger schwarzer Elektronikbaustein mit zwei linsenförmigen Öffnungen der auf einer Elektronikplatine mit anderen Bauteilen verbaut ist.



Warum findet dieser Baustein Verwendung in den fischertechnik-Modellen? Auf kleinstem Raum vereinigt er einen Abstands-, Bewegungs-, Farb- und Lichtsensoren.

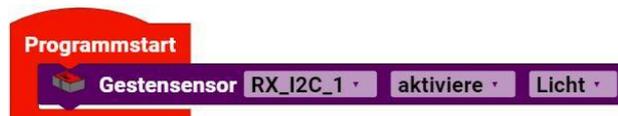
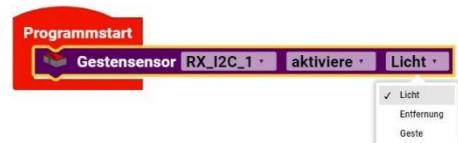


Für diesen Baustein findest du unter ROBO Pro Coding einen eigenen Befehlsblock.



Diesen findest du unter „Sensoren“ – „I2C“.

Schon einmal vorab: Wenn du den Befehl in dein Programm einbindest, kannst du durch Klick auf das Pfeilchen nach „Licht“ ein Kontextfenster öffnen, in dem du die weiteren Funktionen wie „Entfernung“ und „Geste (Bewegung)“ auswählen kannst.



```
1 import user.lib.controller as controller
2 from fischertechnik.logging import log as print
3
4 async def run(instance):
5     controller.init(instance)
6     controller.RX_I2C_1_gesture_sensor.enable_light()
```

Bei jedem Programmbeginn musst du den Sensor initialisieren. Dazu fügst du ihn im freien Bereich des „Programmstart“ ein.

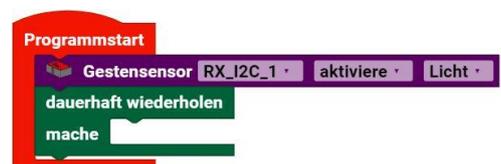


Nach der Initialisierung des Sensors folgt der



Schleifenbefehl „dauerhaft wiederholen mache“. Den Befehl findest du im Befehlsblock

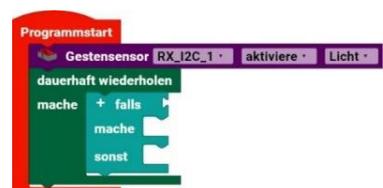
„Verarbeitung“ – „Schleifen“. Füge ihn in deinen Programmablauf ein.



Der nächste Befehl, den du benötigst, ist die Abfrage des vom



Gestensensor ermittelten Lichtwertes (Umgebungslicht). Als Richtwert versuche es mit dem Wert „< 512“ – experimentiere mit verschiedenen Werten. Füge zuerst aus dem Befehlsblock „Verarbeitung“ – „Logik“ den Befehl „falls mache sonst“ ein.





Jetzt musst du noch die 3 Freistellen definieren. Zuerst musst du den Befehl zur Abfrage des Richtwertes am Gestensensor festlegen. Diesen findest du unter „Sensoren“ - „I2C“ - „ist Gestensensor ...“.



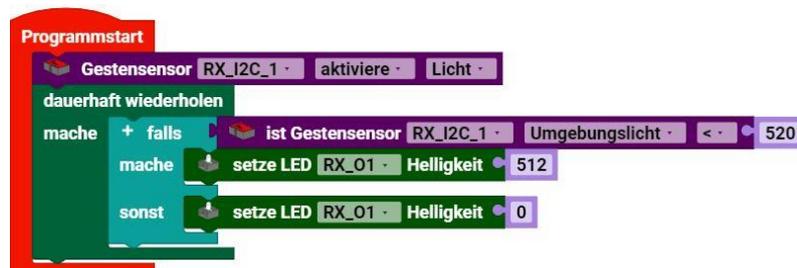
Ziehe den Befehl an die Einfügestelle und ändere die Parameter auf „Umgebungslicht (Klick auf das Pfeilchen nach RGB red)“ - „<“ - „512“ ab.



Ist der Lichtwert „< 512“ soll die LED leuchten. Ist der Wert „>512“ soll die LED ausgehen. Füge den Befehl aus dem Block „Aktoren“ - „Ausgang“ den Befehl „setze LED xx ...“ an die beiden Einfügestellen ein. Lege den Helligkeitswert im ersten Befehl mit „512“ - volle Helligkeit und den Wert im zweiten Befehl mit „0“ - dunkel fest.



Helligkeitswert im ersten Befehl mit „512“ - volle Helligkeit und den Wert im zweiten Befehl mit „0“ - dunkel fest.



So, jetzt kannst du das Programm mit dem Namen **Twilight_switch_1** benennen. Speichere es mit diesem Namen auf deinem Rechner ab. Übertrage das Programm auf deinen Controller und teste es. Teste auch mit verschiedenen Lichtparametern.

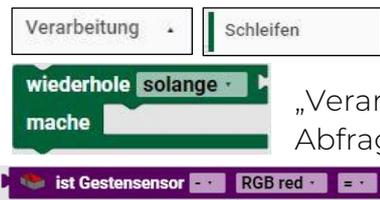
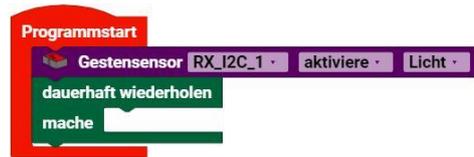
Wichtig: Es kann beim Test vorkommen, dass die LED blinkt. Dies kann am eingestreuten Umgebungslicht liegen. Versuche es in diesem Fall einfach in einem relativ dunklen Raum.

Aufgabe 2

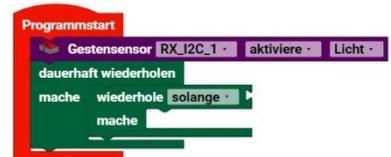
Ändere dein Programm für die 2. Aufgabe so ab, dass die LED, wenn sie im Leuchtmodus ist, blinkt. Erweitere im zweiten Schritt das Programm so, dass du die LED bei ausreichendem Umgebungslicht mit dem Taster ein- und ausschalten kannst.

Dazu musst du die LED nochmals in den Ablauf einbinden, eine Wartezeit von 1 Sekunde einfügen und die LED wieder ausschalten.

Lösche aus dem vorherigen Programm alle Befehle die sich in der „Schleife – dauerhaft wiederholen mache.“



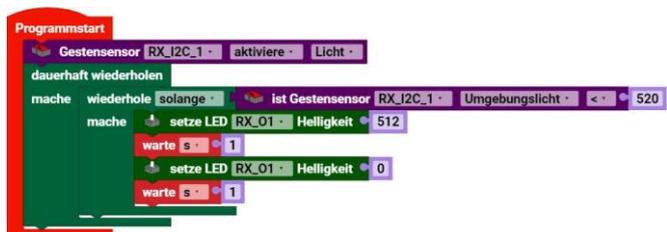
Füge den Befehl „wiederhole xxx mache“ aus dem Befehlsblock „Verarbeitung“ – „Schleifen“ ein. An die Abfrageposition „solange“ kommt der Befehl „ist Gestensensor ...“.



Hier musst du wieder die Parameter „Umgebungslicht“ – „<“ – „520“ setzen.



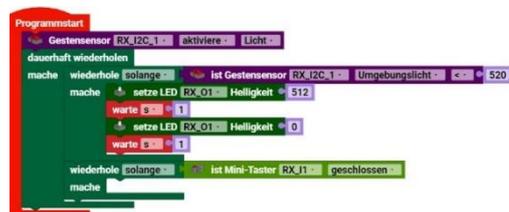
In dem Bereich „mache“ werden die Blöcke für die Blinkfunktion erstellt. Zuerst soll die LED leuchten, nach „1“ Sekunde wieder ausgehen. Sie soll „1“ Sekunde aus bleiben bevor die Schleife wieder durchlaufen wird. Die LED-Schaltbefehle kennst du schon aus der vorherigen Aufgabe. Den „warte x“ Befehl findest du im Befehlsblock „Verarbeitung“ – „Util“. Füge ihn, wie in der Darstellung gezeigt, in dein Programm ein.



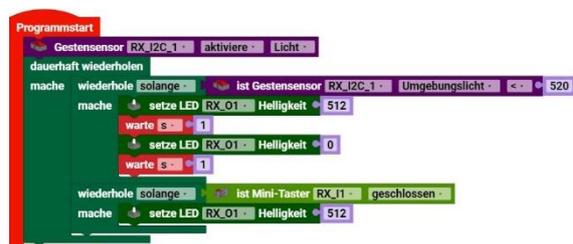
Jetzt muss noch der Taster abgefragt werden. Dazu fügst du unter den „wiederhole mache“ - Befehl einen weiteren ein. An seiner Abfragestelle wird der Mini-Taster abgefragt, ob er geschlossen ist.



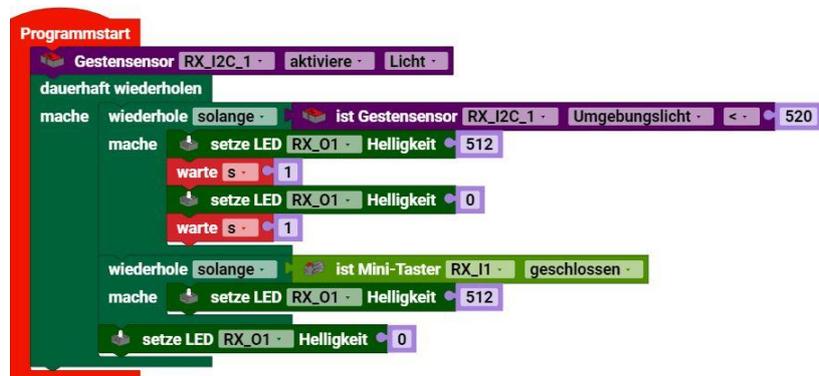
Aus dem Block „Sensoren“ – „Eingang“ benötigst du den Befehl „ist Mini-Taster xxx geöffnet“. Diesen fügst du hinter „solange“ ein. Ändere noch von „geöffnet“ auf „geschlossen“ um.



Ist die Abfrage des Tasters mit „geschlossen“ erfüllt, soll unter „mache“ die LED leuchten. Dazu duplizierst du den Befehl an die freie Stelle im Programm.



Das Ausschalten der LED wird nach der zweiten „Wiederholschleife“ festgelegt. Dazu duplizierst du nochmals den LED-Befehl und fügst ihn unter der „Wiederholschleife“ ein. Setze den Wert der Helligkeit auf „0“.



Teste das Programm mit deinem Modell. Speichere es, wenn alles funktioniert hat, unter dem Namen [Twilight_switch_2](#).

Im nebenstehenden Bild kannst du dir das Programm in der Programmiersprache „Python“ einmal anschauen und die Befehle mit den Blockbefehlen vergleichen.

```

1 import asyncio
2 import user.lib.controller as controller
3 from fischertechnik.logging import log as print
4
5 async def run(instance):
6     controller.init(instance)
7     controller.RX_I2C_1_gesture_sensor.enable_light()
8     while True:
9         while controller.RX_I2C_1_gesture_sensor.get_rgb()[3] < 520:
10             controller.RX_01_led.set_brightness(int(512))
11             await asyncio.sleep(1)
12             controller.RX_01_led.set_brightness(int(0))
13             await asyncio.sleep(1)
14             await asyncio.sleep(0.01)
15         while controller.RX_I1_mini_switch.is_closed():
16             controller.RX_01_led.set_brightness(int(512))
17             await asyncio.sleep(0.01)
18         controller.RX_01_led.set_brightness(int(0))
19         await asyncio.sleep(0.01)
20

```

Demontiere anschließend das Modell und baue das nächste Modell auf.