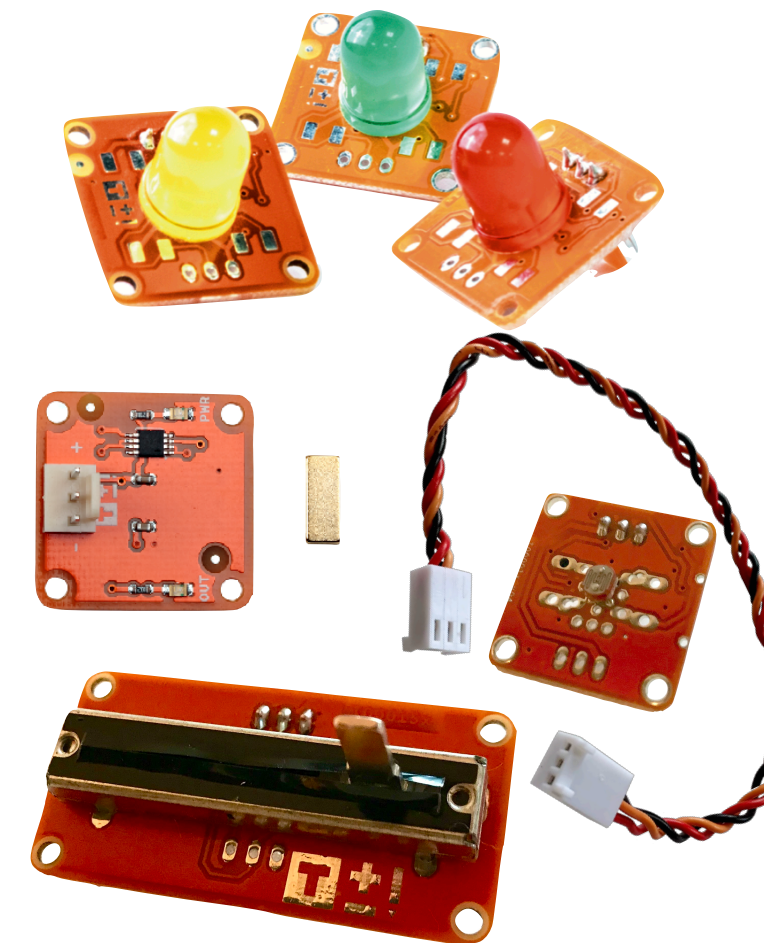
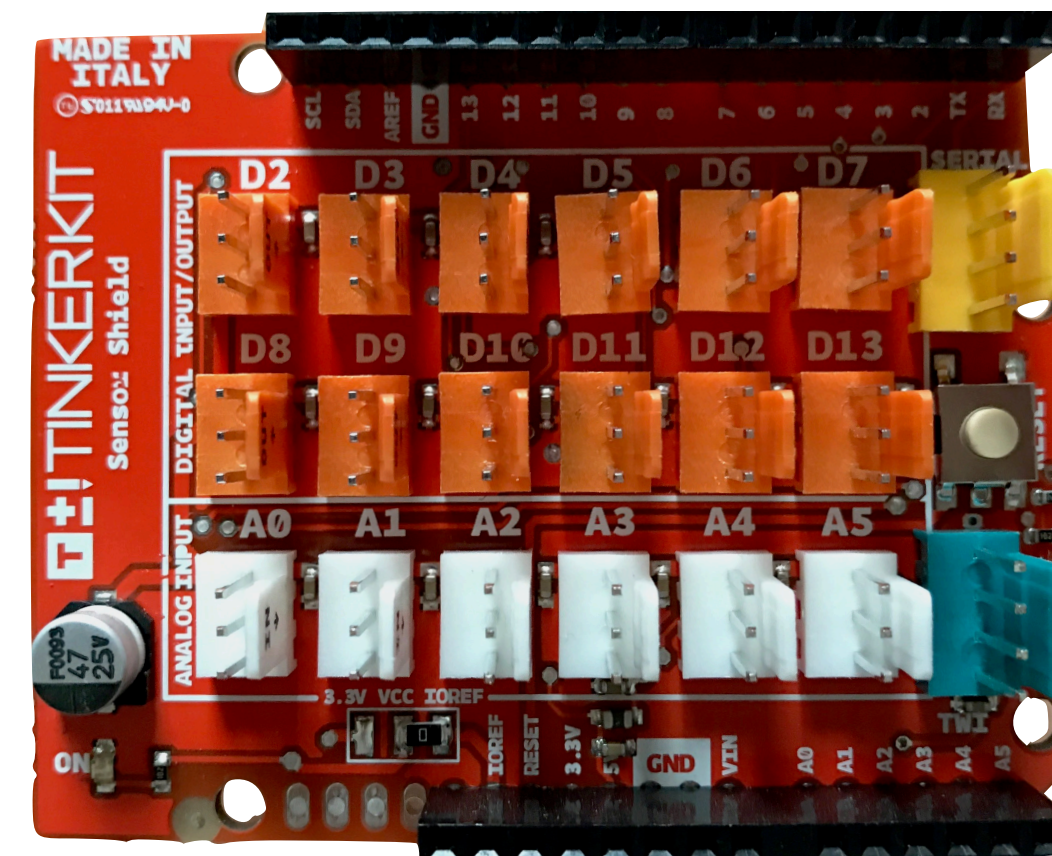
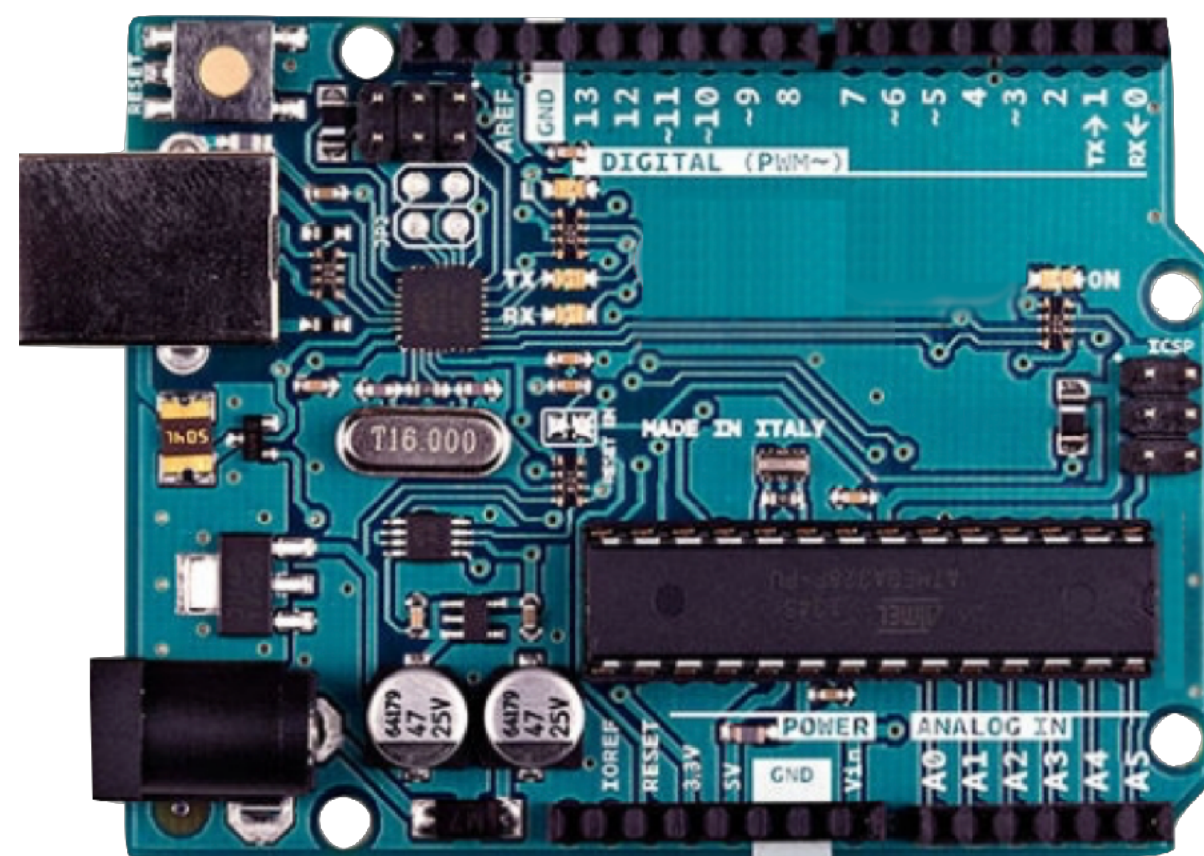
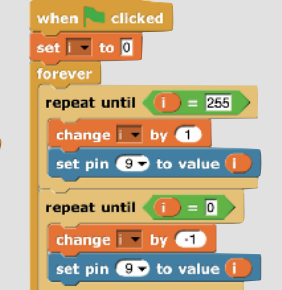
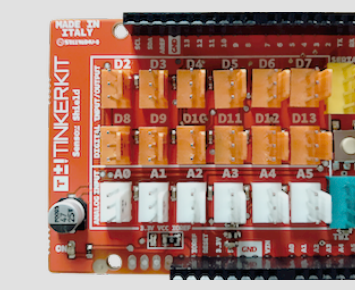
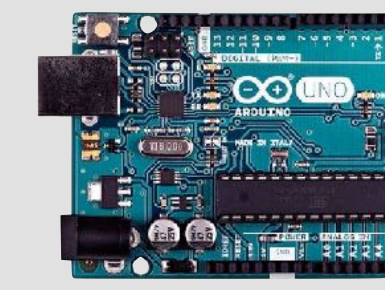




Einführung in Physical Computing



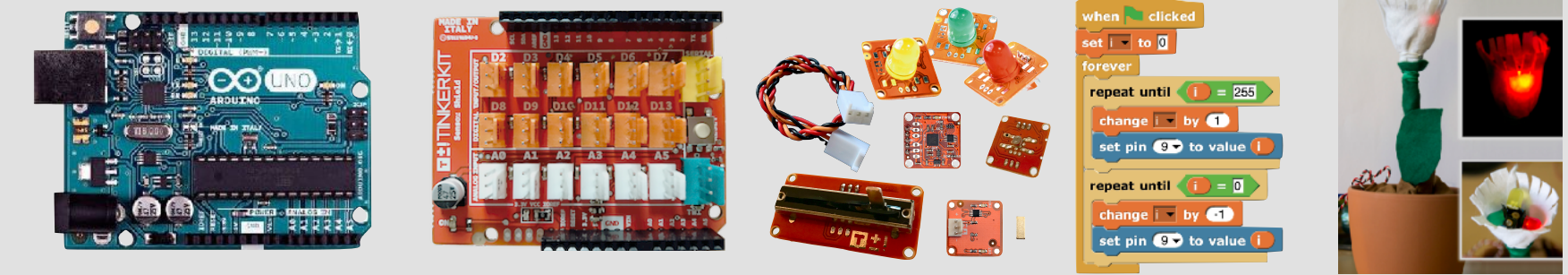
```
when clicked
set i to 0
forever
  repeat until i > 254
  change i by 1
  set pin 6 to value i
  repeat until i < 1
  change i by -1
  set pin 6 to value i
```



... mit Arduino, TinkerKit und Snap4Arduino

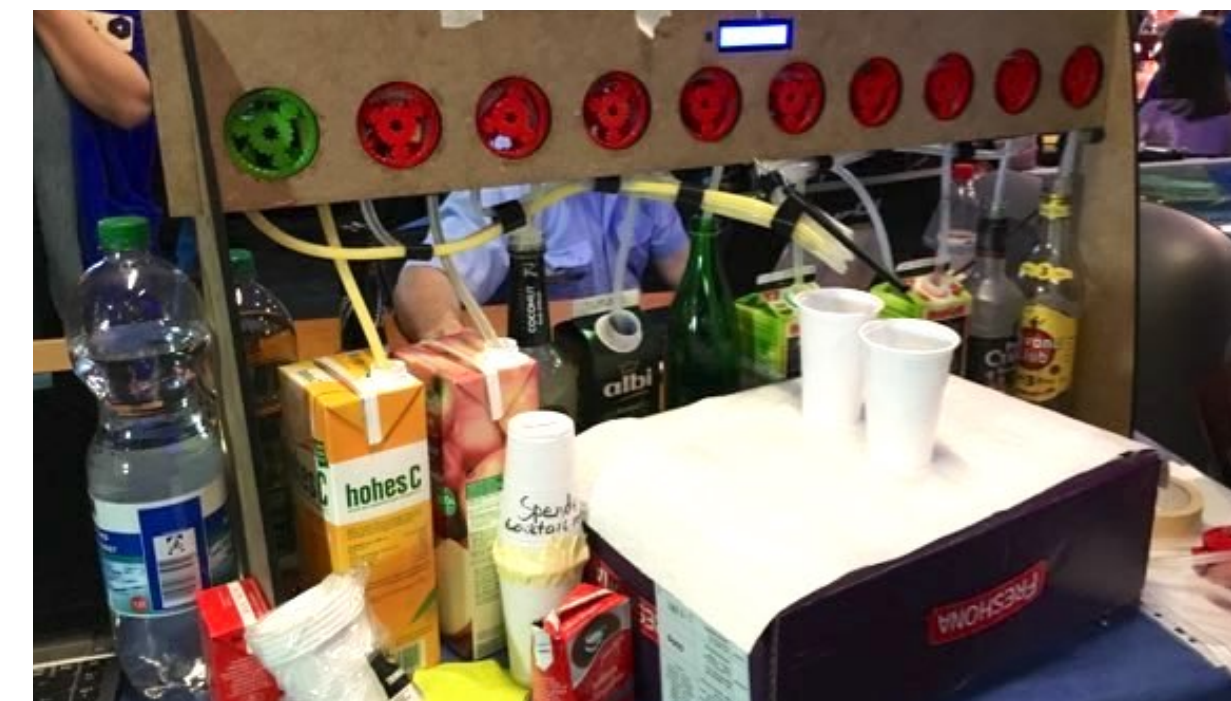
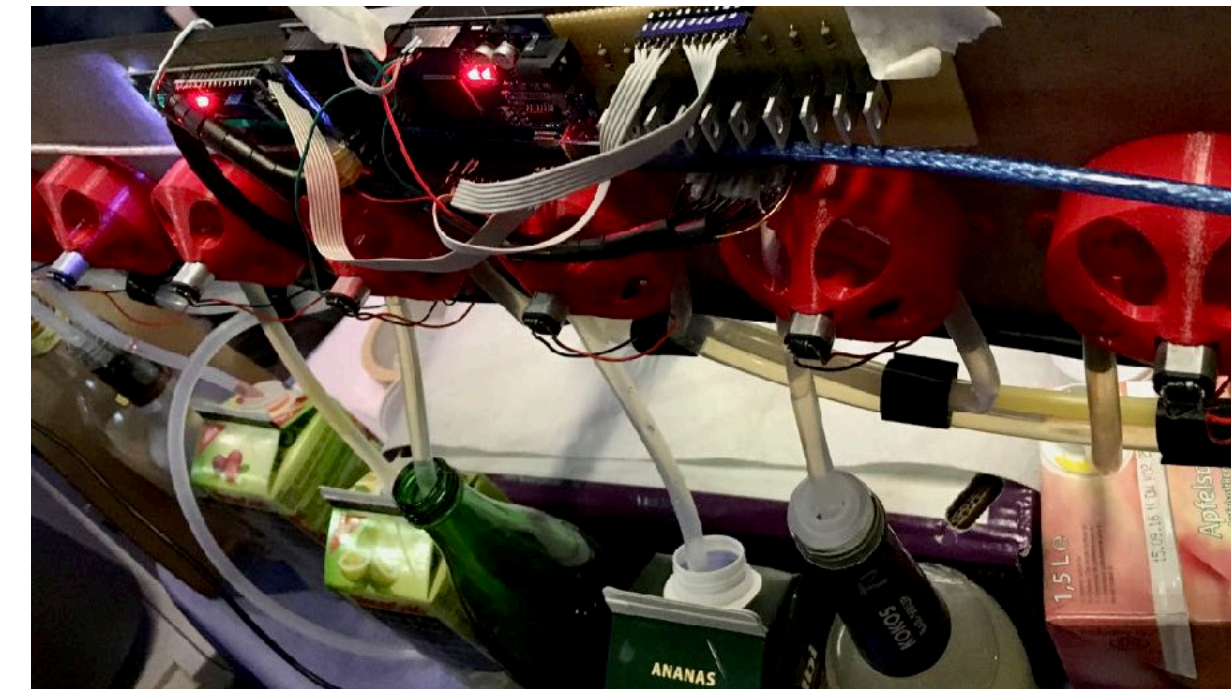
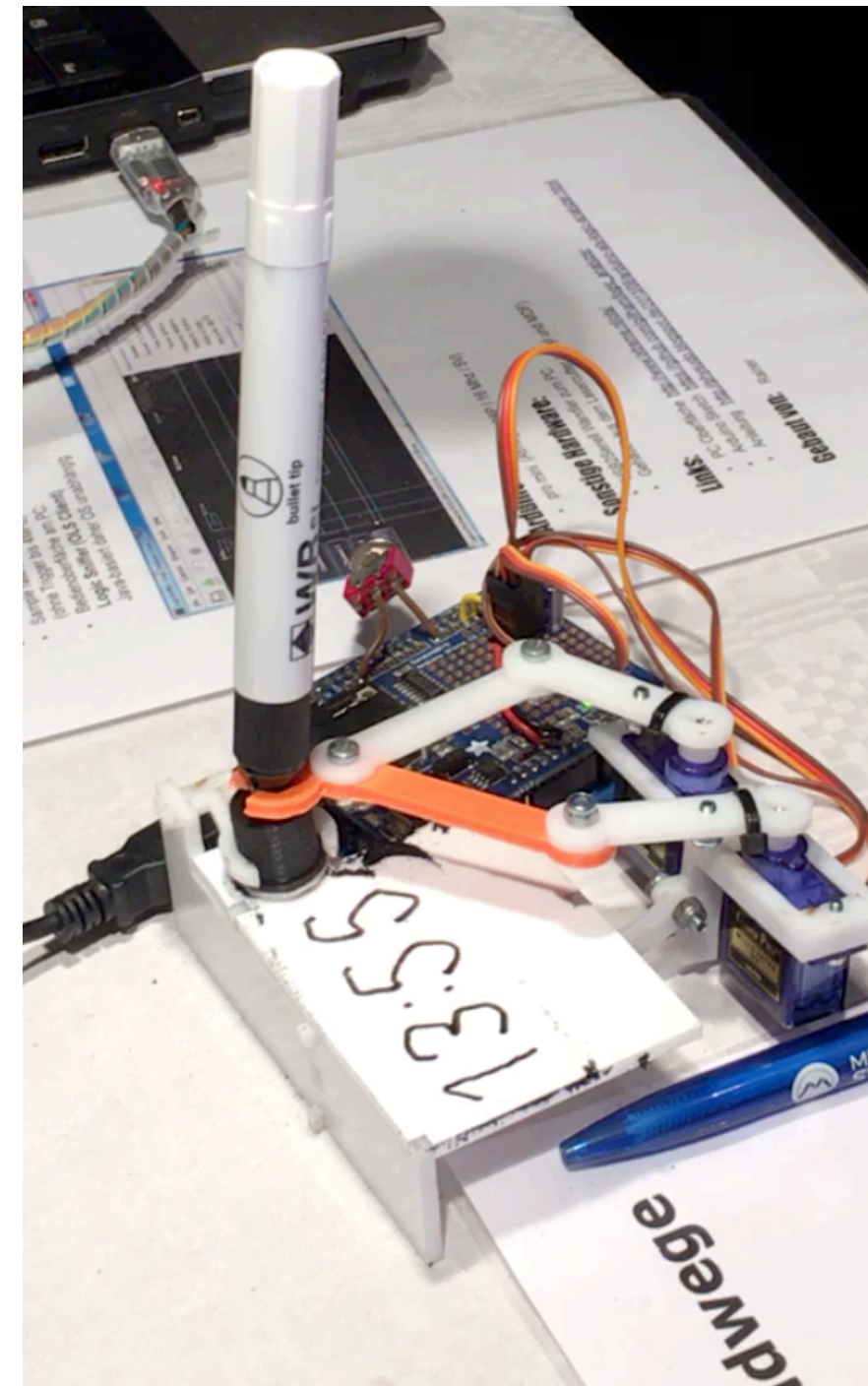


Was ist Physical Computing?



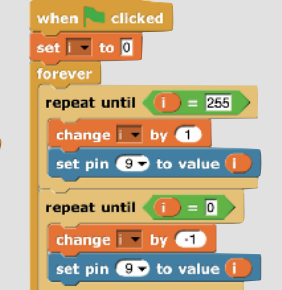
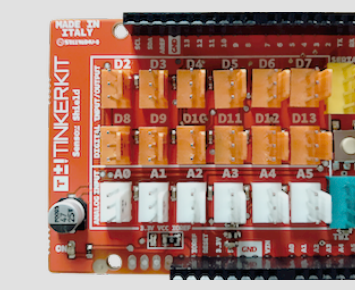
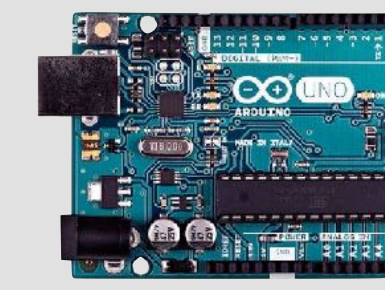
- Künstler und Designer nutzen programmierbare Hardware zur Herstellung interaktiver Objekte
- Interaktive Objekte kommunizieren über Sensoren und Aktoren mit Menschen und der “analogen Welt”
- Verhalten ist als Software auf einem Mikrocontroller implementiert

[in Anlehnung an Banzi 2011]

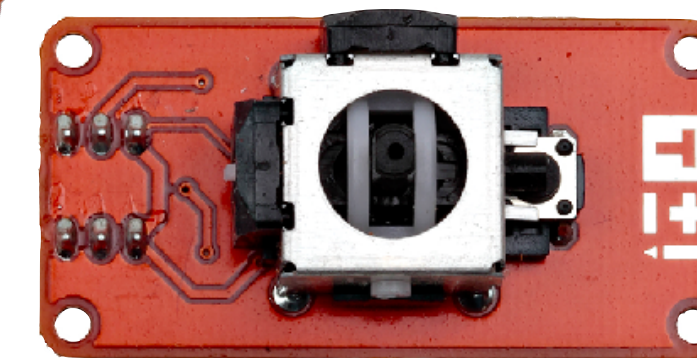
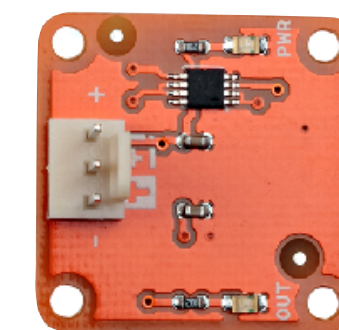
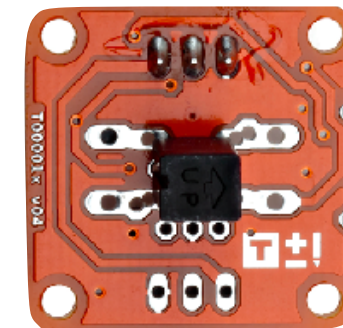
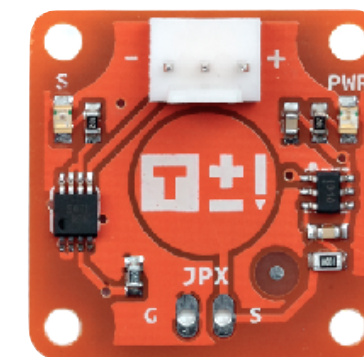
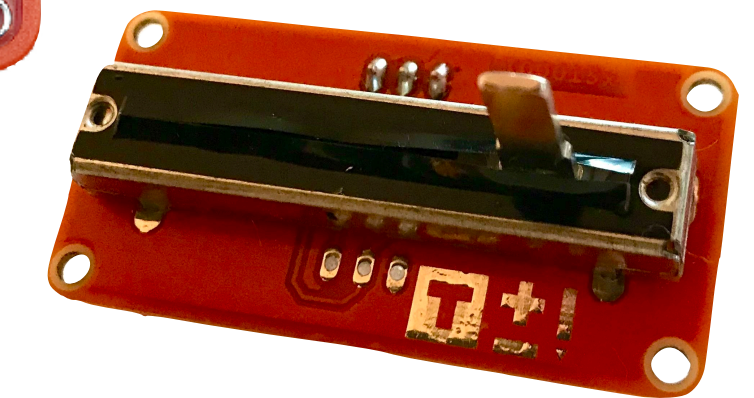
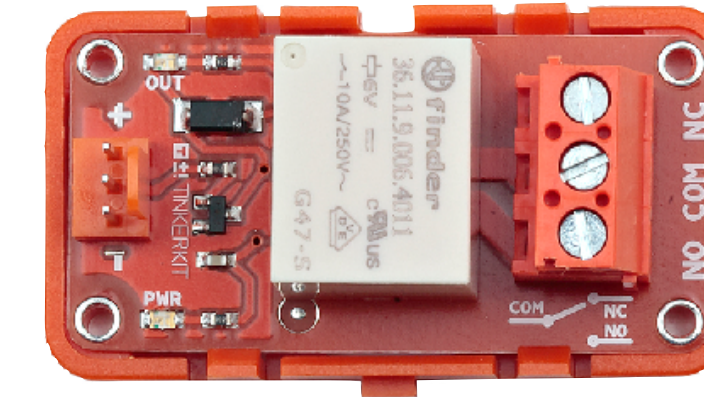
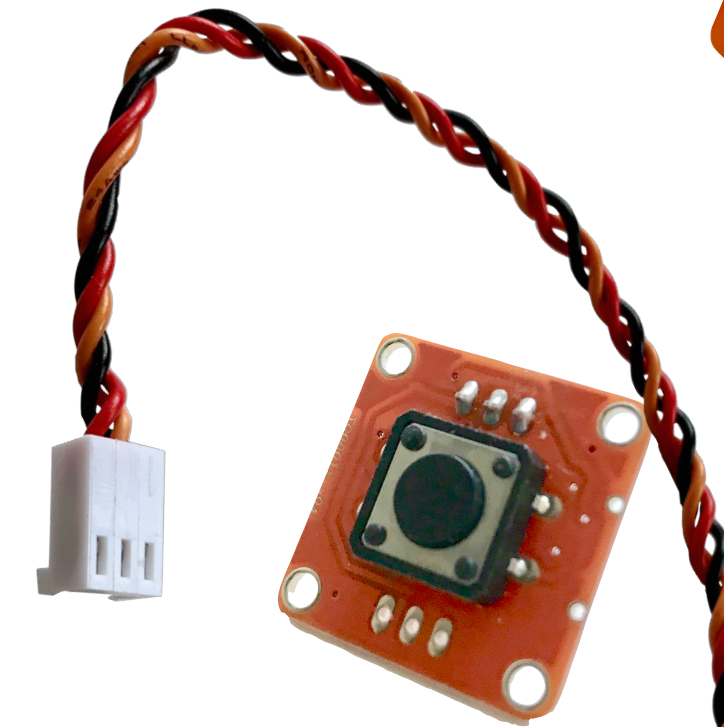
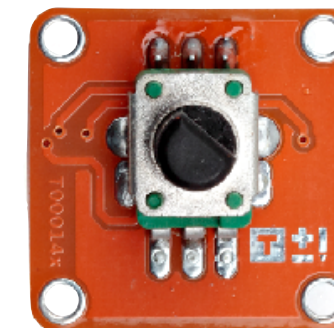
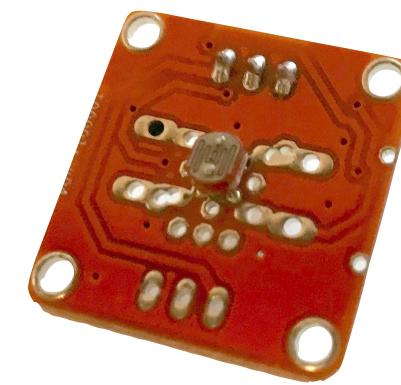
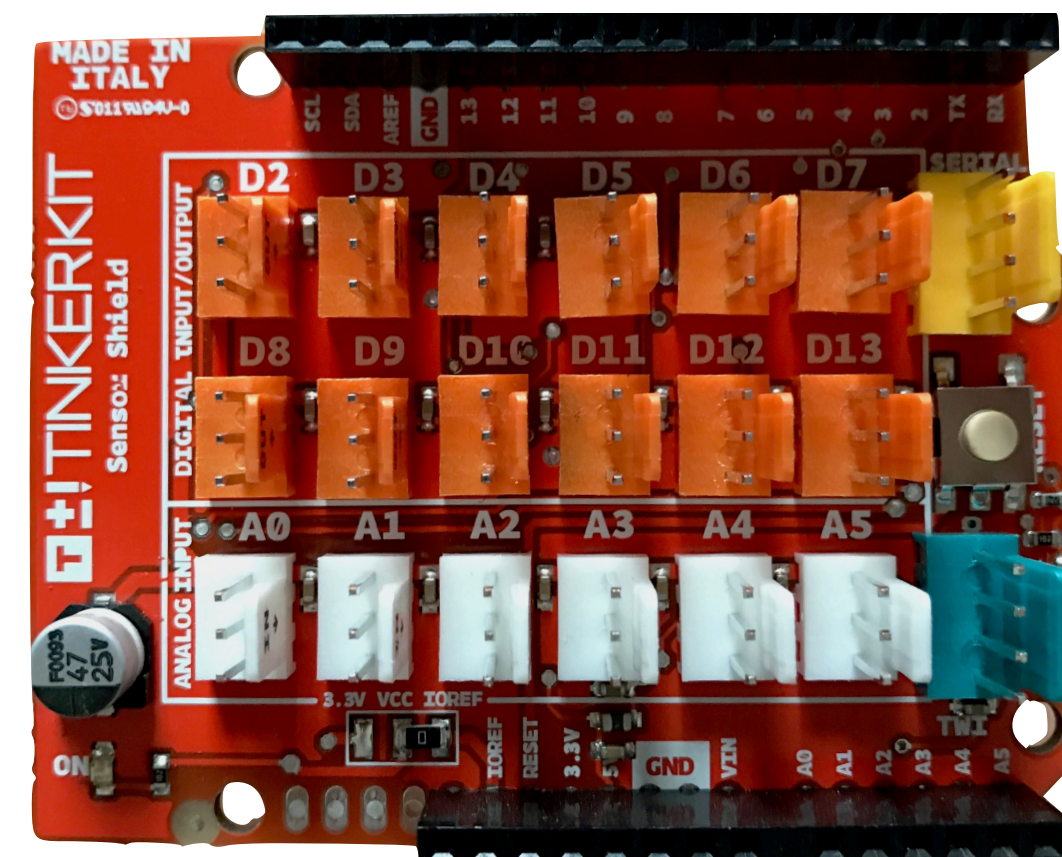
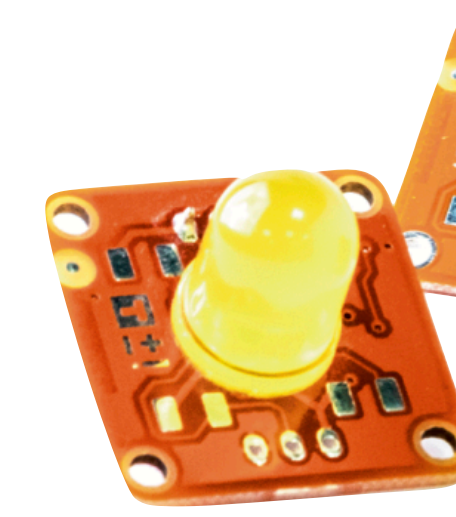
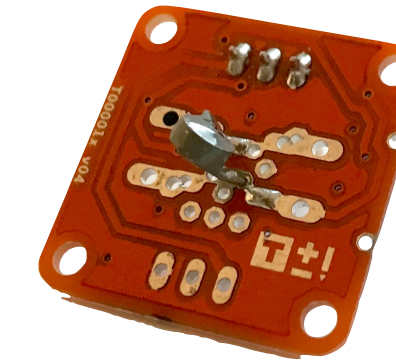
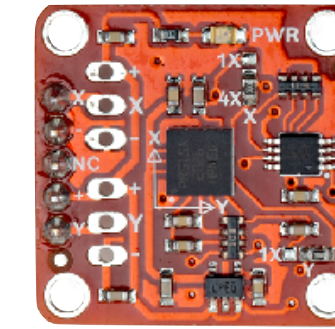
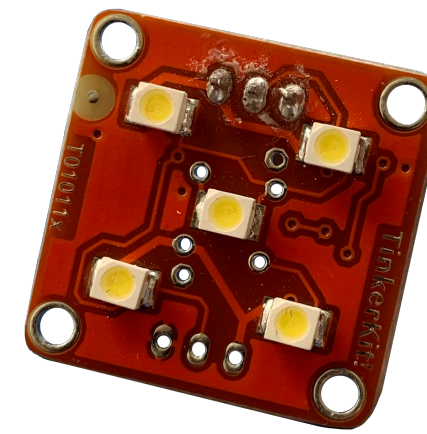
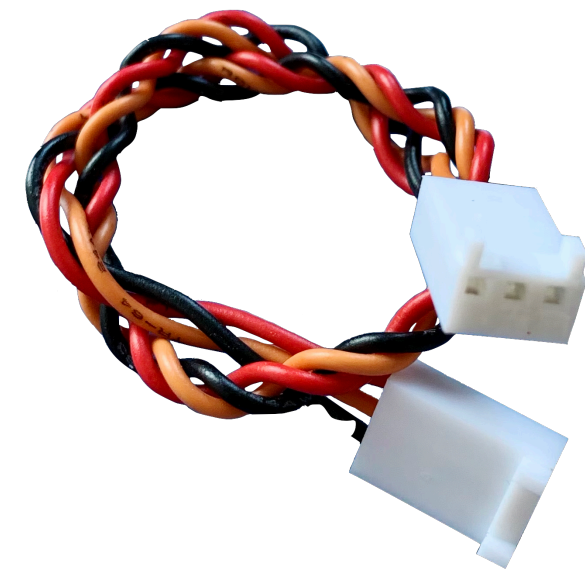
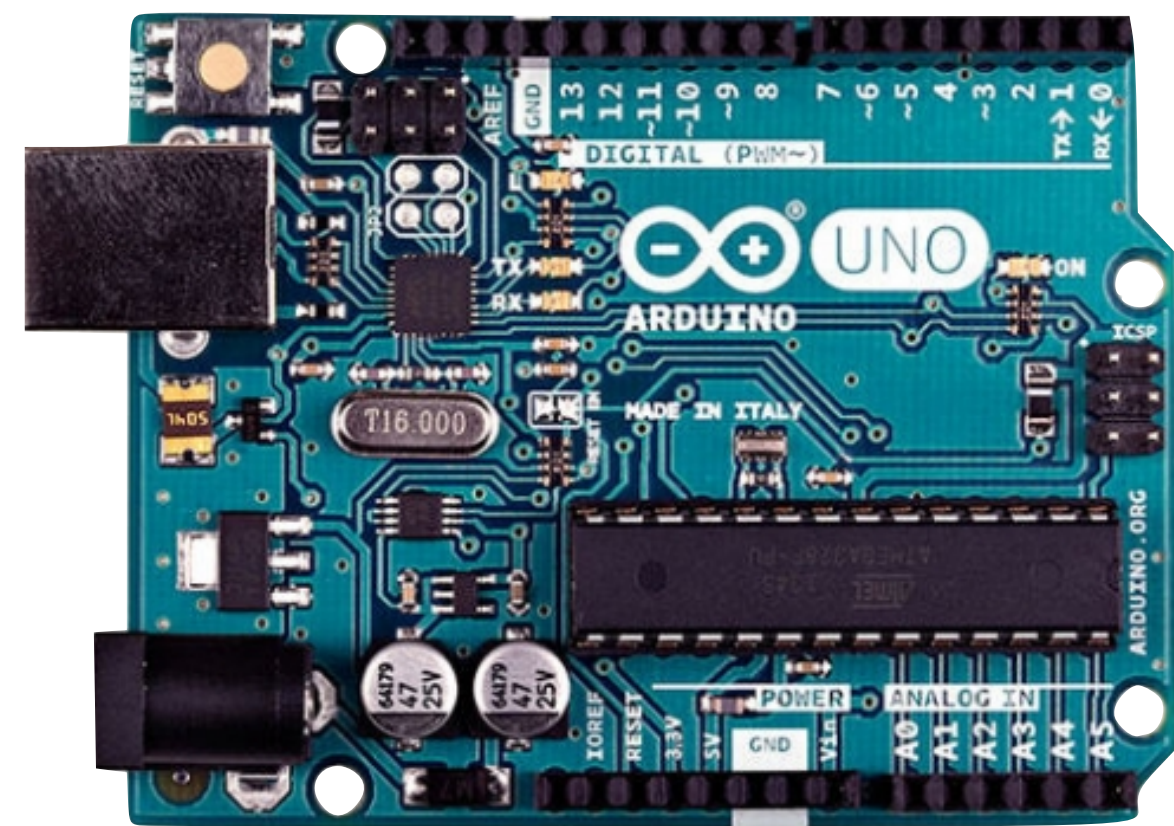




Mikrocontroller, Sensoren, Aktoren

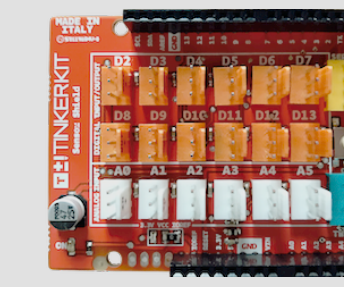
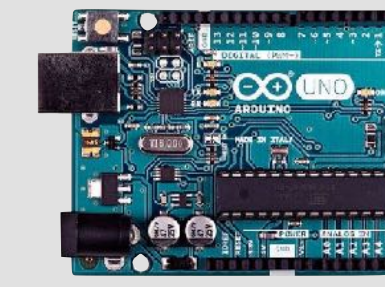


Arduino Uno und TinkerKit





Programmierumgebung Snap4Arduino



Snap4Arduino

DEMO lucky button

Motion, Looks, Sound, Pen, Other, Control, Sensing, Operators, Variables, Arduino

Sprite: draggable

Scripts, Costumes, Sounds

when I receive blink

repeat times

set random number to pick random 1 to 300

if random number < 100

set digital pin 2 to ✓

else

if random number < 200

set digital pin 3 to ✓

else

set digital pin 4 to ✓

wait seconds secs

set digital pin 2 to ✗

set digital pin 3 to ✗

set digital pin 4 to ✗

when I receive have luck

set lucky number to pick random 1 to 300

if lucky number < 100

set digital pin 2 to ✓

else

if lucky number < 200

set digital pin 3 to ✓

else

set digital pin 4 to ✓

wait 5 secs

when clicked

forever

set digital pin 2 to ✗

set digital pin 3 to ✗

set digital pin 4 to ✗

if analog reading 0 > 900

set times to 100

set seconds to 0.02

broadcast blink

set times to 50

set seconds to 0.03

broadcast blink

set times to 20

set seconds to 0.1

broadcast blink

broadcast have luck

random number 122

times 20

seconds 0.1

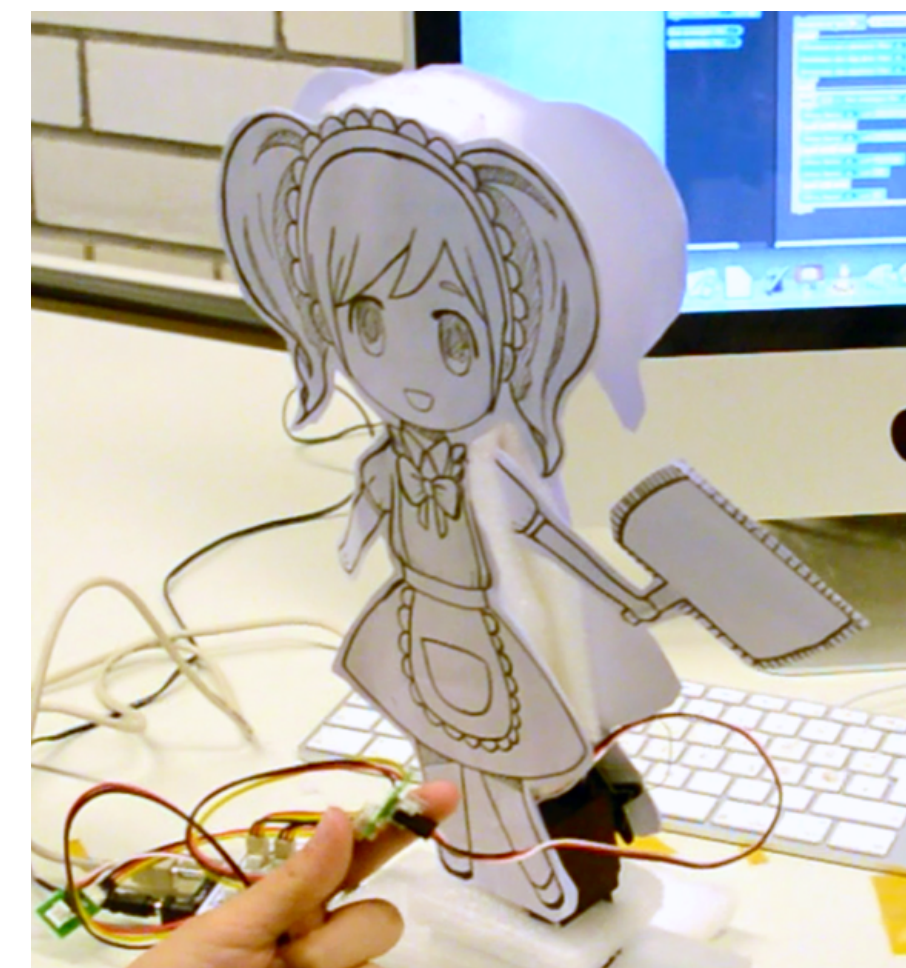
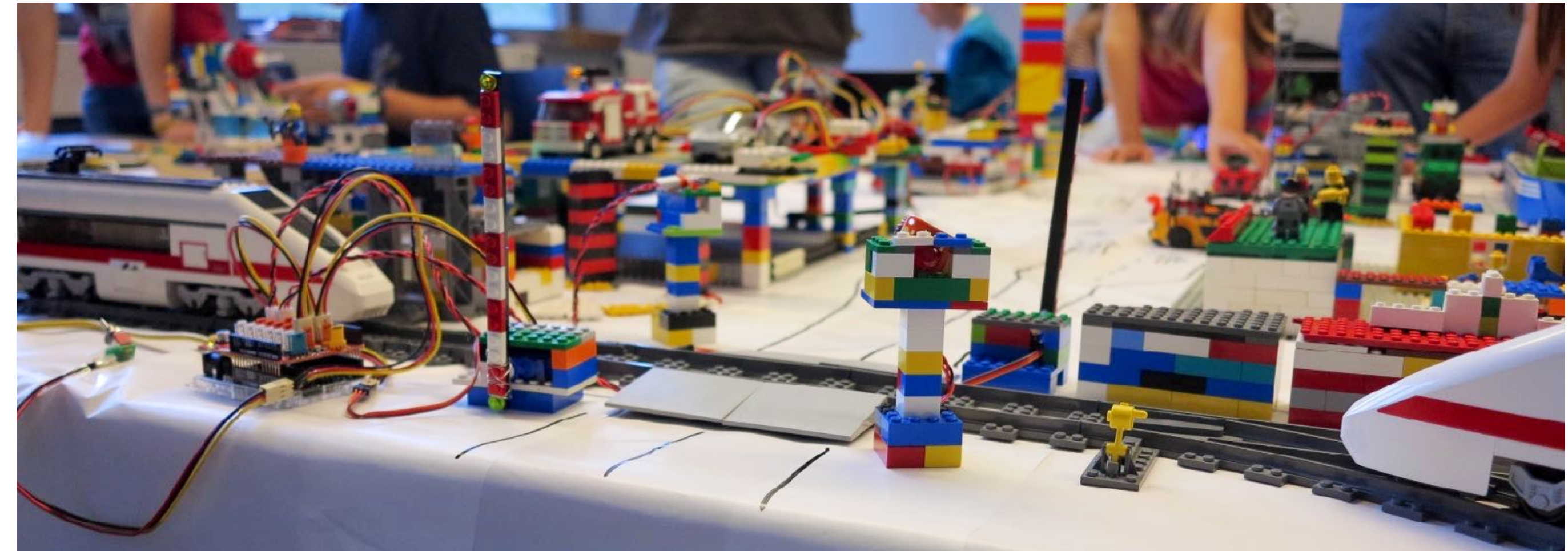
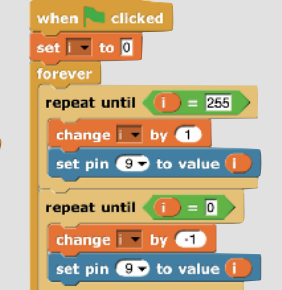
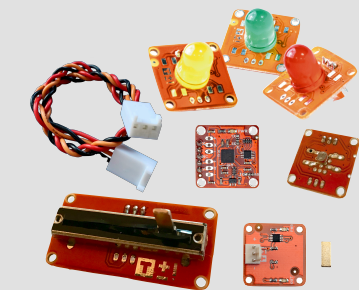
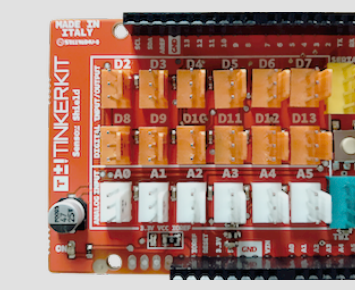
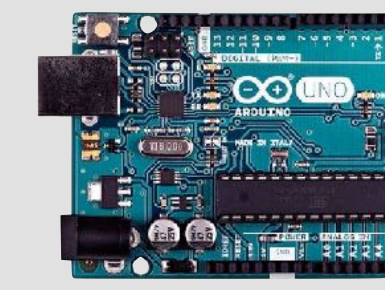
lucky number 0

Sprite

Stage

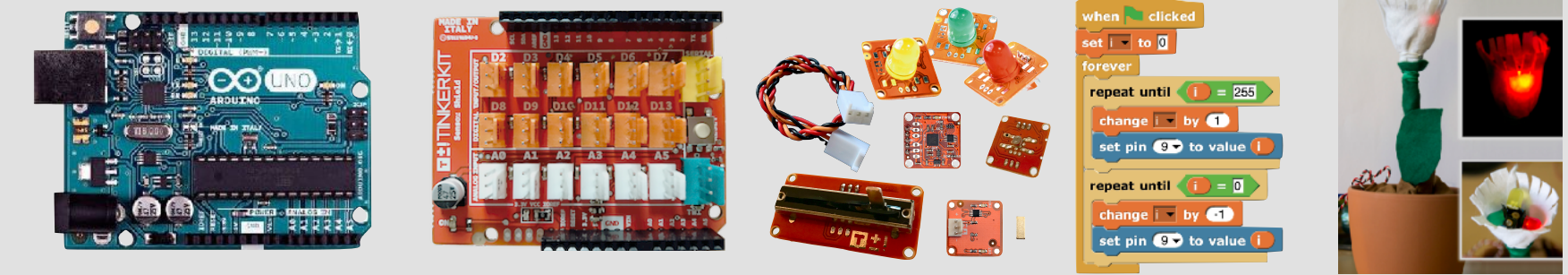


Beispielprojekte

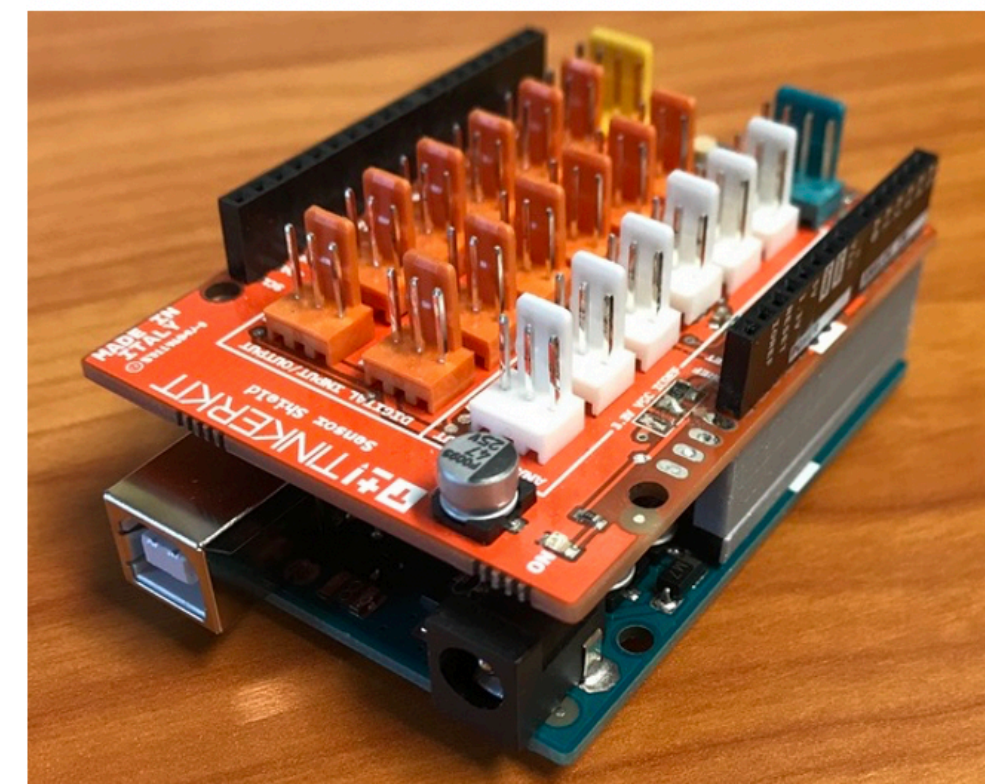
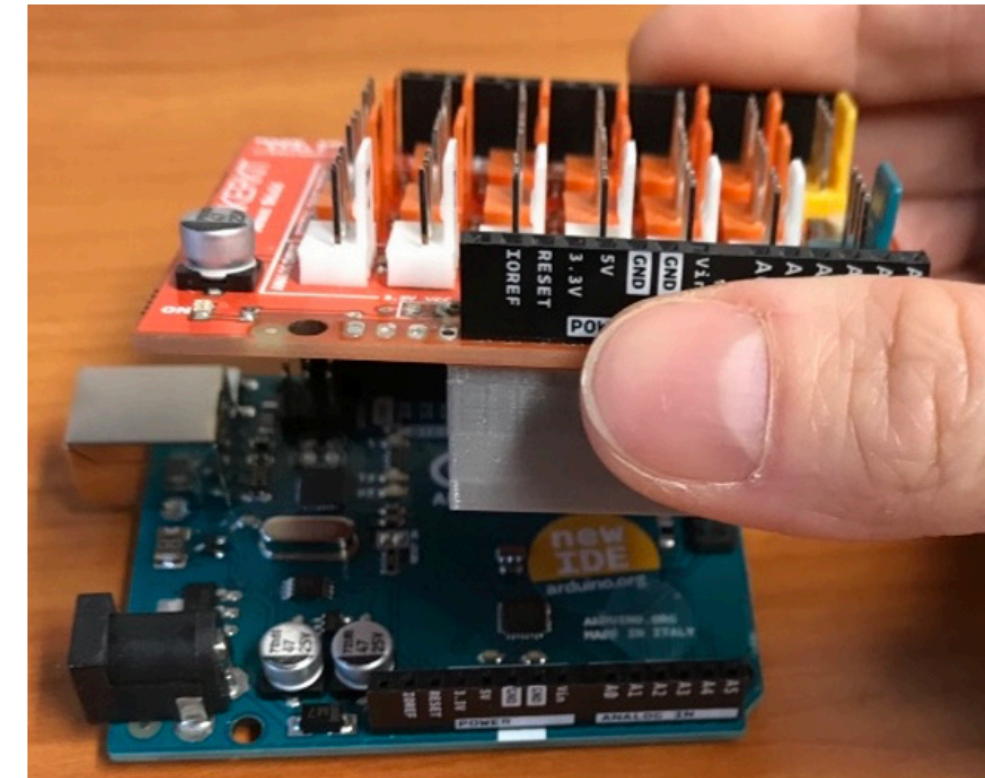




Vorbereitungen



- in Arduino-IDE modifizierte Standard-Firmata (FirmataSnap4Arduino.ino) auf Arduino hochladen ✓
- Shield auf Arduino stecken ✓
- per USB mit Computer verbinden
- Snap4Arduino starten
 - Library “Snap4Arduino-Blöcke.xml” einbinden (per Drag&Drop in den Skriptbereich ziehen)
 - im Bereich “Arduino” auf “mit Arduino verbinden” klicken



```

FirmataSnap4Arduino | Arduino 1.8.13
Upload Using Programmer
FirmataSnap4Arduino pitches.h rgb_lcd.cpp rgb_lcd.h
/*
Firmata is a generic protocol for communicating with microcontrollers
from software on a host computer. It is intended to work with
any host computer software package.

This is an extension of Arduino StandardFirmata to be used with Snap4Arduino
and corresponding blocks (LCD-Melody.xml).
It allows to control LCD displays and buzzers.

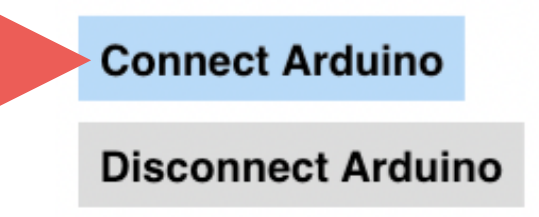
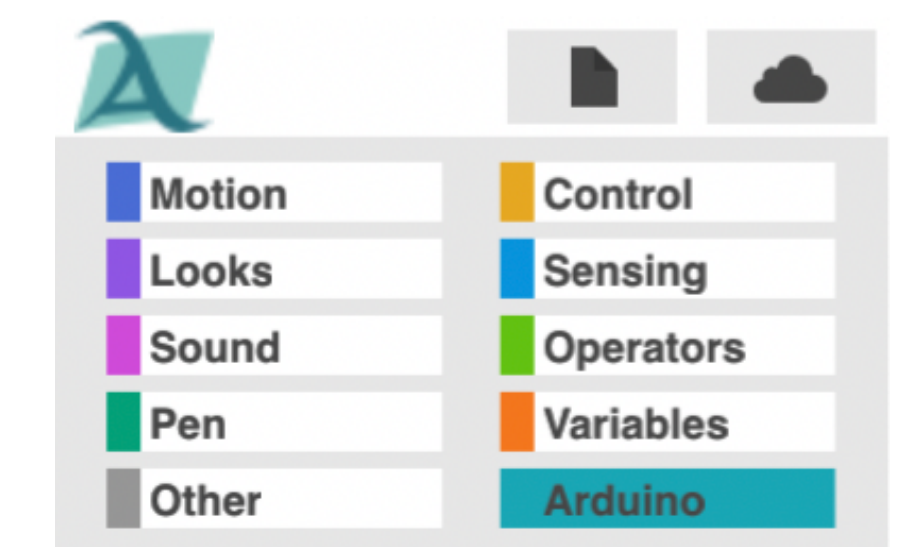
Last updated September 9th, 2020 by Mareen Przybylla (kontakt@mareen.ch)
*/

#include <Servo.h>
#include <Wire.h>
#include <Firmata.h>
#include "rgb_lcd.h"
#include "pitches.h"

rgb_lcd lcd;

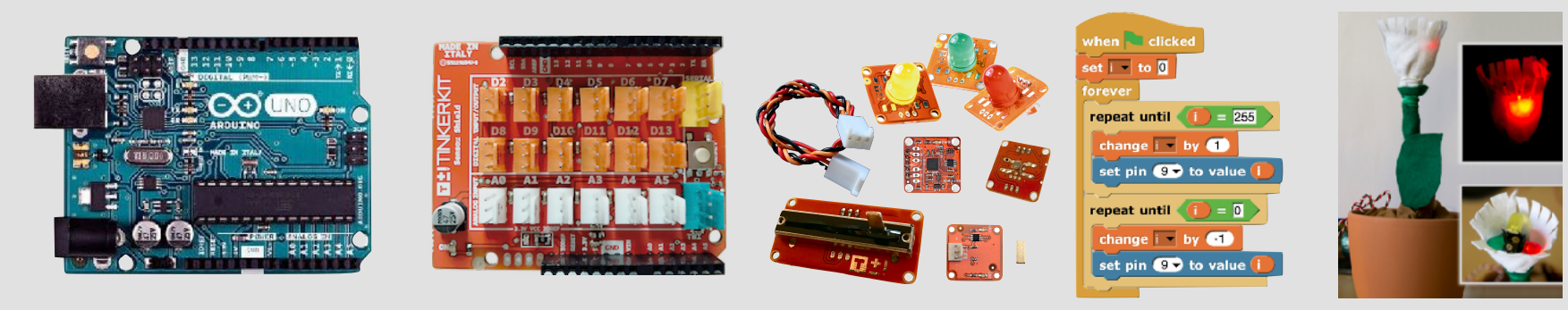
#define I2C_WRITE           B00000000
#define I2C_READ           B00001000
#define I2C_READ_CONTINUOUSLY B00010000
#define I2C_STOP_READING  B00011000
#define I2C_READ_WRITE_MODE_MASK B00011000
#define I2C_10BIT_ADDRESS_MODE_MASK B00100000
#define I2C_END_TX_MASK   B01000000
#define I2C_STOP_TX       1
#define I2C_RESTART_TX    0
#define I2C_MAX_QUERIES   8
#define I2C_REGISTER_NOT_SPECIFIED -1

// the minimum interval for sampling analog input
  
```




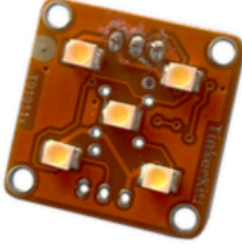



Ablauf



- Stationenlernen zum Kennenlernen der Bauteile:
 - (1) Taster und digitale Sensoren, (2) Analoge Sensoren, (3) LEDs, (4) Servomotoren, (5) Piezo-Summer, (6) LCD-Displays
- Regeln:
 - 2 Schüler(innen) pro Gruppe
 - Stationen in beliebiger Reihenfolge bearbeiten
 - Jeder füllt Laufzettel aus, zusätzliche Notizen machen
 - Wenn fertig: melden zum Vorführen
 - Materialien nach Verwendung wieder zur Station

Bei den LED-Modulen handelt es sich um Aktoren, die sowohl binär als auch mittels PWM angesteuert werden können:

	Farbige LEDs in verschiedenen Größen
	Hi-Power-LEDs
	Anschlusskabel (rot-orange-schwarz)

Zum Ein- oder Ausschalten der LEDs wird in Snap4Arduino der folgende Block aus der „Arduino“-Kategorie genutzt:



Hier werden der zu sendende Status (*an* oder *aus*) und der Pin, an den die LED angeschlossen ist, eingetragen.

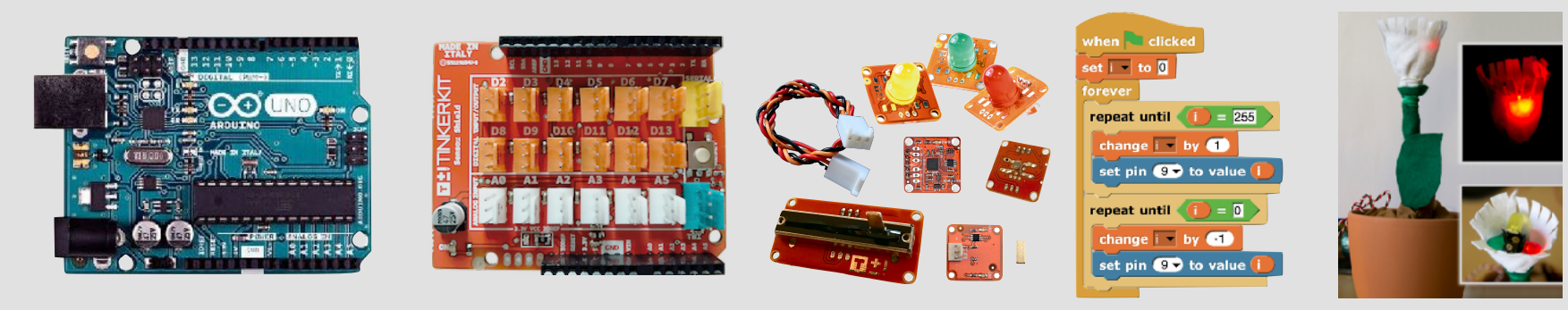
Um die Helligkeit einer LED steuern zu können, wird in Snap4Arduino der folgende Block aus der „Arduino“-Kategorie genutzt:



Hier wird der zu sendende Helligkeitswert (zwischen 0 und 255) und der Pin, an den die LED angeschlossen ist, eingetragen. Beachte: eine solche pseudanaloge



Ablauf



- Stationenlernen zum Kennenlernen der Bauteile:

- (1) Taster und digitale Sensoren, (2) Analoge Sensoren, (3) LEDs, (4) Servomotoren, (5) Piezo-Summer, (6) LCD-Displays

- Regeln:

- 2 Schüler(innen) pro Gruppe
- Stationen in beliebiger Reihenfolge bearbeiten
- Jeder füllt Laufzettel aus, zusätzliche Notizen machen
- Wenn fertig: melden zum Vorführen
- Materialien nach Verwendung wieder zur Station

Aufgaben:

- 1) SchlieÙe eine LED an den Pin D3 vom Arduino an.
 - a. Lasse die LED leuchten und schalte sie anschließend wieder aus.
 - b. Was bewirkt das folgende Programm?

```
Wenn  angeklickt
fortlaufend
  Setze digitalen Pin 3 auf 
  warte 1 Sek.
  Setze digitalen Pin 3 auf 
  warte 1 Sek.
```

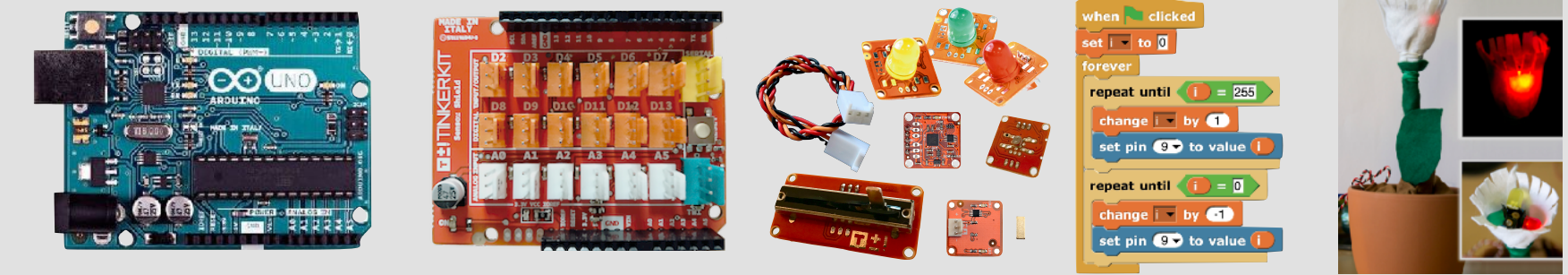
bzw.

```
when  clicked
forever
  set digital pin 3 to 
  wait 1 secs
  set digital pin 3 to 
  wait 1 secs
```


- 2) Schreibe ein Programm, das zwei LEDs im Wechsel an- und wieder ausschaltet.
- 3) Schreibe ein Programm, das eine LED im Sekundentakt immer heller werden lässt.
- 4) Für welche beispielhaften Anwendungszwecke können die LEDs dienen?





Ablauf



- Stationenlernen zum Kennenlernen der Bauteile:
 - (1) Taster und digitale Sensoren, (2) Analoge Sensoren, (3) LEDs, (4) Servomotoren, (5) Piezo-Summer, (6) LCD-Displays
- Regeln:
 - 2 Schüler(innen) pro Gruppe
 - Stationen in beliebiger Reihenfolge bearbeiten
 - Jeder füllt Laufzettel aus, zusätzliche Notizen machen
 - Wenn fertig: melden zum Vorführen
 - Materialien nach Verwendung wieder zur Station

 **Laufzettel – Stationenlernen Physical Computing** von: _____ (Name)

Station	Aufgaben	Notizen	Einschätzung	✓
1: Taster und digitale Sensoren (binärwert)	<input type="checkbox"/> Kippschalter <input type="checkbox"/> weitere digitale Sensoren <input type="checkbox"/> Anwendungsmöglichkeiten	<hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/>	 	
2: Analoge Sensoren (wertdiskret)	<input type="checkbox"/> Schiebewiderstand <input type="checkbox"/> weitere analoge Sensoren <input type="checkbox"/> Anwendungsmöglichkeiten	<hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/> <hr/>	 	

Prof. Dr. Mareen Przybylla
 Pädagogische Hochschule Schwyz
