

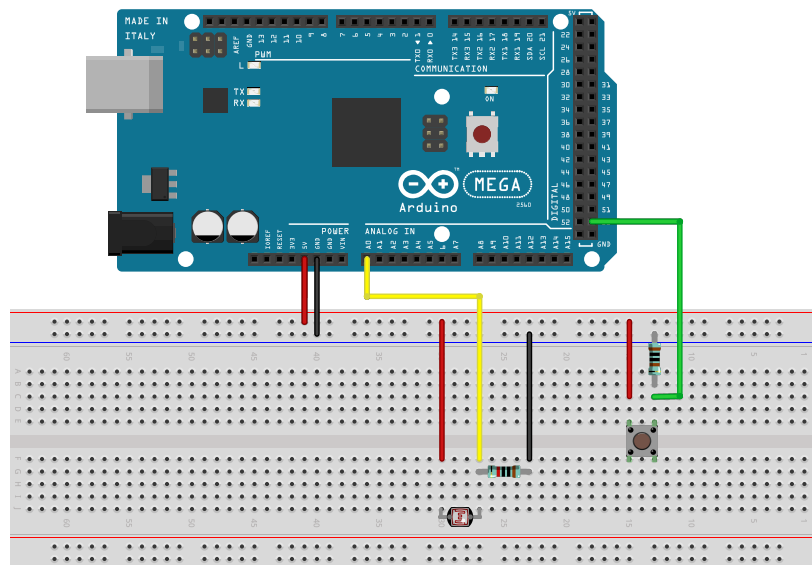
Abgabe

Geben Sie die Lösung bis **Montag, 22. Mai** um 23:59 als PDF-Dokument in einem ZIP, inklusive aller Quelldateien, per E-Mail an Ihren Tutor ab. Die Abgabeformalitäten sind die Gleichen wie auf Blatt 1.

1 Messreihen (6 Punkte)

In den vergangenen Vorlesungen haben Sie Schleifen und Arrays kennengelernt. Auf diesem Blatt werden Sie Programme mithilfe dieser Konstrukte implementieren.

In dieser Aufgabe geht es darum Daten von einem Licht-Sensor auszulesen und in einem Array abzuspeichern. Bauen Sie dazu zuerst die dargestellte Schaltung auf.



fritzing

Der Taster hat einen 10kOhm Pull-Down-Widerstand dazu geschaltet, auch der Licht-Sensor hat einen solch hohen Widerstand.

1. Beim ersten Drücken des Tasters beginnt eine Messung.
2. Beim zweiten Drücken des Tasters, oder nach maximal 30 Sekunden, ist die Messung abgeschlossen.
3. Während einer Messung soll alle 10 Millisekunden ein Messpunkt angelegt werden. Dazu muss Ihr Programm den Sensor-Wert abfragen und in einem Array hinterlegen.
4. Ist die Messung abgeschlossen, so sollen das Minimum, das Maximum und der Durchschnittswert aller Messpunkte auf der seriellen Konsole ausgegeben werden.

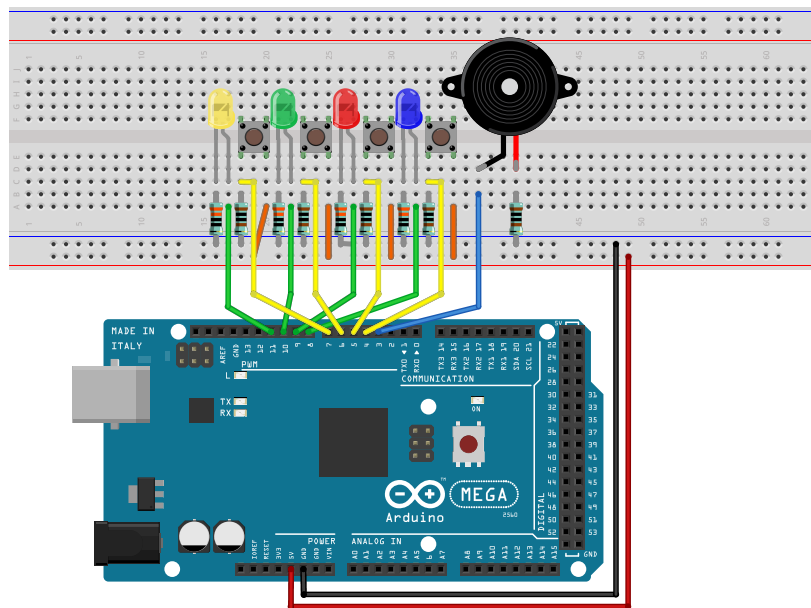
2 Senso (5 Punkte + 1 Punkt + 2 Bonuspunkte)

2.1 Die Implementierung (5 Punkte)

In dieser Aufgabe werden Sie eine Variante des beliebten Spiels *Senso* implementieren. Bei *Senso* gibt ein Spieleautomat eine Ton- und Lichtsequenz vor, die der Spieler wiederholen muss. Jede Runde wird die Sequenz um ein Element (Ton und Farbe) länger und die Abspielgeschwindigkeit erhöht sich. Sobald der Spieler die Sequenz nicht mehr korrekt wiederholen kann, hat er verloren.

Sie werden jedoch nicht das Spiel selbst implementieren, sondern eine Variante davon. Der Unterschied zum Original ist, dass Sie und der Computer die Rollen tauschen: Sie können per Knopfdruck die abzuspielende Sequenz vorgeben und Ihr Arduino wird diese abspielen und mit Lichtern animieren.

1. Schließen Sie LEDs, Taster und den Piezo-Lautsprecher wie in der Abbildung gezeigt an. (Hinweis: Der Piezo-Lautsprecher sieht in der Abbildung anders aus. Sie erkennen ihn daran, dass er am blauen Kabel angeschlossen ist.)



Bei den LED-Widerständen handelt es sich wieder um ca 200Ohm Widerstände, die Taster sind mit 10kOhm Widerstände auf Ground geschaltet. Hinter dem Piezo-Lautsprecher, welcher an einem PWM-fähigen Pin angeschlossen ist, ist ein 1kOhm Widerstand.

2. Implementieren Sie den Ablauf wie unten beschrieben.
3. Der erste Taster ist der ersten LED (blau) zugeordnet, usw.
4. Der erste Taster soll einen Ton von 220Hz erzeugen. Die anderen Taster 440, 660, 880Hz entsprechend.
5. Zum Abspielen von Tönen schauen Sie sich `tone` und `noTone` auf der Arduino-Website¹ an.

¹<http://www.arduino.cc/en/Reference/HomePage>

Ablauf:

1. Warten Sie auf einen Tastendruck bei einem der 4 Taster.
2. Spielen Sie den entsprechenden Ton und lassen Sie die entsprechende LED aufblinken.
3. Abspielen und Aufblinken passiert dabei parallel und dauert beides 750ms.
4. Merken Sie sich den gedrückten Taster in einem Array.
5. Spielen Sie die bereits gemerkte Sequenz ab. Jeder Ton, bzw. jedes Aufblinken der entsprechenden LED soll dabei 500ms dauern.
6. Danach wird wieder bei Punkt 1 begonnen.

2.2 Arraygröße (1 Punkt)

Was fällt Ihnen hierbei bezüglich der Arraygröße auf? Kann es zu Problemen kommen?

2.3 Bonus: Ringspeicher (2 Bonuspunkte)

In dieser Aufgabe sollen Sie dem Array, in welchem die Eingaben gespeichert werden, eine feste Größe von 25 geben. Außerdem sollen Sie das Programm derart anpassen, dass wenn das Array voll ist, das älteste gespeicherte Element “gelöscht” wird – wodurch wieder ein Eintrag im Array frei wird.

Dies hat zum Effekt, dass jeweils nur die maximal 25 letzten Tastendrucke wiederholt werden.