

Abgabe

Geben Sie die Lösung bis **Donnerstag, 07. Mai** um 11:00 als PDF-Dokument per E-Mail an Ihren Tutor ab. Die Abgabeformalitäten sind die Gleichen wie auf Blatt 1.

1 Automatische Ampelsteuerung

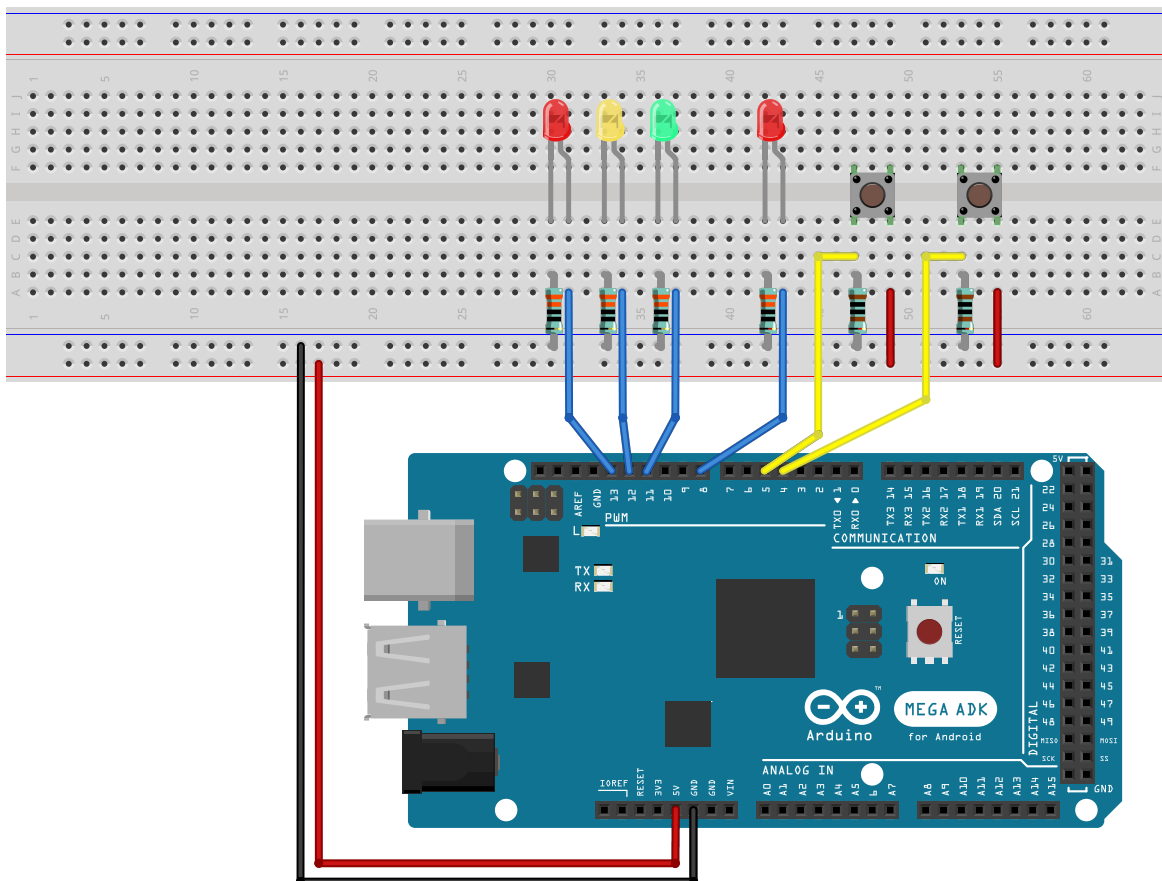
Auf diesem Übungsblatt geht es darum, den Einsatz von `delay(ms)`, `millis()` und `if () {}` an einem praktischen Projekt zu erproben.

Es geht um die Realisierung einer Ampel, die Automobilen anzeigt, ob sie fahren dürfen oder nicht.

Bauen Sie dazu zunächst die Ampelschaltung wie in der Abbildung gezeigt auf. Achten Sie dabei besonders auf die Polung der LEDs: Der kürzere Pin der LED ist dabei die Kathode und zeigt in Richtung der Erde (GND). Auf der Abbildung ist die Kathode durch den **nicht** abgeknickten Pin dargestellt. Außerdem ist es wichtig die LEDs nur zusammen mit einem Widerstand (330Ω) anzuschließen. Die Taster werden mit einem $1k\Omega$ -Widerstand geerdet.

Zunächst soll die Ampel rein zeitgesteuert sein. Ein Ampelübergang läuft in Zeitschritten von 1 Sekunde wie folgt ab: Rot auf Rot-Gelb, auf Grün, auf Gelb, und zurück auf Rot.

Zur Realisierung der Verzögerung sollen Sie dabei die Funktion `delay(ms)` einsetzen, die die in Millise-



fritzing

kunden übergebene Zeit lang wartet.

2 Interaktive Ampelsteuerung

Erweitern Sie Ihre Ampelsteuerung aus der vorherigen Aufgabe:

1. An Pin 5 ist ein Taster mit Pull-Down-Widerstand angeschlossen (siehe Abbildung).
2. Die Ampel wird weiterhin Automobilen anzeigen, ob sie fahren dürfen oder nicht.
3. Die Ampel bleibt jetzt dauerhaft grün, solange kein Fußgänger die Straße überqueren möchte.
4. Wenn der Taster gedrückt wird, wechselt die Ampel nach rot. Der Übergang läuft weiterhin ab, wie in Aufgabe 1 beschrieben. (Jede Phase 1 Sekunde)
5. Die Ampel soll nach jeder Rotphase für mindestens 1 Sekunde grün sein.
6. Tasterdrücke werden also nie ignoriert, aber unter Umständen erst zeitverzögert ausgeführt.

Damit ein Tastendruck immer erkannt werden kann, können Sie nicht mittels `delay(ms)` warten. Benutzen Sie `millis()` um die aktuelle Zeit abzufragen und schalten Sie die Ampel dementsprechend.

Tipps:

1. Versuchen Sie erst Ihren Code aus Aufgabe 1 derart umzuschreiben, dass er `millis()` anstatt `delay(ms)` verwendet und bauen Sie **danach** den Taster ein.
2. Es empfiehlt sich die Verwendung einer Variable, die speichert, wann die letzte Rotphase abgeschlossen war.
3. Es empfiehlt sich außerdem die Verwendung einer Variable, die speichert, ob der Taster gedrückt und noch nicht verarbeitet wurde.

3 Bonus: Nachtschaltung

Erweitern Sie Ihre Ampelsteuerung mit einem Nachtmodus:

1. An Pin 4 wird analog zum Fußgänger-Taster ein weiterer Taster angeschlossen.
2. Bei erstmaligen Druck des Tasters wechselt die Ampel in den Nachtmodus.
3. Im Nachtmodus sind die grüne und alle roten LEDs aus. Die gelbe LED blinkt (250 ms an, 250 ms aus).
4. Bei nochmaligen Druck des Tasters wechselt die Ampel wieder in den Tagmodus.
5. Die Ampelsteuerung soll erst in den Nachtmodus wechseln, wenn die Ampel auf Rot steht.