

Versuch 2 – Ampel-Steuerung mit Sicherheitsfunktion

Aufgabe 1: Ampelschaltung, Port I/O, Sicherheitsfunktion

Bauen Sie eine neue Schaltung (Arduino-Programmname: v2-1-ampeltest) zur Ansteuerung von drei Ampel-LEDs und zur Abfrage von zwei Tastern auf. Achten Sie hierbei darauf, dass Port D auf dem Arduino Board ohne Verschaltung bleibt (Vorlage in Vorlesung 3, Folie 9 – aber **kein Taster an Pin 13 !!!**).

Port D soll später noch für Erweiterungen verwendet werden!

Zunächst sollen Sie dafür sorgen, dass die rote Ampel-LED leuchtet, wenn der Taster an Pin 12 dauerhaft gedrückt wird. Solange der Taster gedrückt bleibt, soll die Ampel rot anzeigen und somit jeglichen Unfall von vorne herein ausschließen.

Nutzen Sie hierzu die **Programmvorlage** in Vorlesung 3, Folie 11 als Grundlage und nehmen Sie Änderungen an folgenden Stellen vor:

- a) Ändern Sie das Programm zunächst so, dass auf dem Arduino Uno die Pins 8-10 auf Ausgang geschaltet werden.
- b) Ändern Sie dann die Aktivierung der internen Pull-Up Widerstände so, dass diese an den Pins 11-12 aktiviert werden.
- c) Ändern Sie schließlich die Abfrage des Tasters auf Pin 12 des Arduino Uno – dies sollte Ihr linker Taster sein.

Bei gedrücktem Taster an Pin 12 sollte nur die rote LED leuchten und auch wieder ausgehen, wenn der Taster losgelassen wird.



Aufgabe 2: Ausgaben auf dem seriellen Monitor

Speichern Sie Ihr Projekt/Programm unter dem Namen: v2-2-ampel

Bei der Arbeit mit einem an den PC angeschlossenen Arduino Uno gibt es die Möglichkeit, während des Programmablaufs Texte oder Inhalte von Variablen in einem Fenster auf dem PC anzeigen zu lassen.

Hierzu gibt es im Menü 'Werkzeuge' den Punkt 'Serieller Monitor'. Ein Klick darauf öffnet dann das entsprechende Fenster.

Fügen Sie nun in Ihrem Programm folgende Zeilen hinzu:

In der setup-Funktion:

```
Serial.begin(9600);  
Serial.println("Hello World!");
```

Zwischen setup-Funktion und loop-Funktion:

```
int i = 0;
```

In der loop-Funktion:

```
i++;  
  
Serial.println(i);
```

Lassen Sie das Programm laufen, im seriellen Monitor sollte zum Programmstart einmal "Hello World!" ausgegeben werden und in Folge der Inhalt der Variablen i, der sich in jedem Durchlauf des loops um 1 erhöht.

Jetzt, wo Sie wissen wie das funktioniert, können Sie alle Zeilen außer 'Serial.begin(9600);' wieder löschen und den Aufruf von Serial.println(...); bei der Bearbeitung der nächsten Aufgabe sinnvoll nutzen.

Aufgabe 3: Ampelsteuerung mit Anzeige eines Phasenzählers

Programmieren Sie eine einfache Ampelsteuerung mit Hilfe der delay-Funktion.

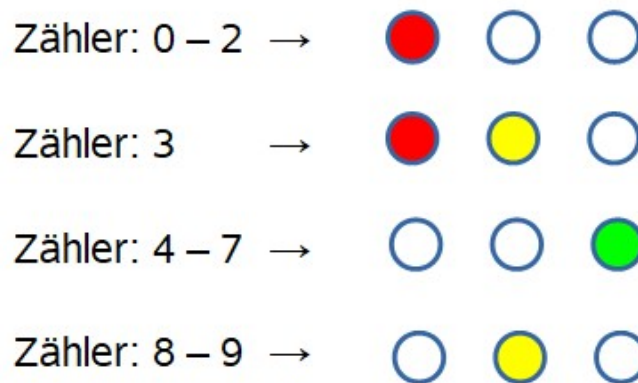
Erweitern Sie dazu das bisherige Projekt/Programm (siehe Vorlesung 3, Folie 12ff).

- a) Programmieren Sie einen Zähler mit dem Namen 'phase', der von 0 – 9 hochzählt, dann zurück auf Null springt und erneut bis 9 zählt, usw. Die Zählgeschwindigkeit sollte bei ungefähr einer Sekunde liegen.

Zeigen Sie den aktuellen Zählerstand auf dem **seriellen Monitor** an.

- b) Implementieren Sie die Ampelsteuerung mithilfe der drei Ampel-LEDs.

Dabei sollen die folgenden Ampel-Phasen realisiert werden:



Kleiner Tipp: In Vorlesung 3, Folie 14, 15 und 16 gibt es drei verschiedene Vorlagen zur Realisierung.

Testen Sie Ihr Programm und überprüfen Sie, ob die 'Sicherheitsfunktion' (linker Taster) noch funktioniert. Die brauchen Sie nämlich für Aufgabe 6 und 7.

Aufgabe 4: Ampel mit variabler Geschwindigkeit

Erweitern Sie das bisherige Projekt/Programm mit der folgenden Funktionalität:

- Das dauerhafte Drücken des rechten Tasters führt dazu, dass die Ampel langsamer läuft:
-> (ein Zählschritt dauert nicht 1, sondern 5 Sekunden).

Kleiner Tipp: der Übergabewert beim Aufruf der delay-Funktion kann auch eine Variable sein.



Aufgabe 5: erneutes Testen der Sicherheitsfunktion

Verändern Sie nun die Verzögerungszeit des Zählers (der von 0-9 zählt) von einer Sekunde auf 5 Sekunden, indem Sie den rechten Taster dauerhaft drücken.

Testen Sie jetzt noch einmal die 'Sicherheitsfunktion' (linker Taster)
– was fällt Ihnen auf? Reagiert das System immer **sofort**?

Notieren Sie in Ihrem Programmcode kurz (oben, als ‚Kommentar‘ mit //), welches Verhalten Sie beobachten können.

Das Projekt v2-2-ampel ist damit beendet.

Aufgabe 6: (Verbesserung der Sicherheitsfunktion)

Speichern Sie Ihr Projekt/Programm unter dem Namen: v2-3-ampel-krass

Die Sicherheitsfunktion soll jetzt so verbessert werden, dass das Schalten auf rot bei gedrückter linker Taste **direkt** passiert, **ohne Wartezeit bzw. Verzögerung!**

Dazu ändern Sie Ihr Programm so ab, dass die delay-Funktion nicht mehr verwendet wird. Verwenden Sie statt dessen eine Zählvariable vom Datentyp **long int**, z.B. mit dem Namen ‚counter‘, die für eine getaktete Ausführung zuständig sein soll.

Kleiner Tipp: In Vorlesung 3, Folie 27 gibt es eine Vorlage zur Realisierung eines Zeittakts mit Hilfe einer Zählvariablen.

Es sollten jetzt die folgenden Programme vorhanden sein:

- V2-1-ampeltest
- V2-2-ampel
- V2-3-ampel-krass

Wenn Sie bis hierhin alle Aufgaben erfolgreich bearbeiten konnten, haben Sie den Laborversuch 2 hiermit bestanden und werden das Teilstat erhalten.

Viel Erfolg !!!