



E2 Informatik 2

Versuch 3 – die krasse Ampel und eine Eieruhr

Aufgabe 1 (Erweiterung der Ampelschaltung von v2-7 mit LCD):

Wir verwenden in diesem Versuch 3 die gleiche Schaltung wie bei Versuch 2, also:

Bauen Sie eine Schaltung 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.

Erstellen Sie ein neues Arduino-Projekt mit dem Namen: **v3-ampel-krass**

Zusätzlich soll ein Liquid Crystal Display (LCD) verwendet werden. Lesen Sie zum grundsätzlichen Verständnis und zur Verschaltung des LCDs das Dokument 'LCD-I2C-real.pdf' aus dem Ordner 'Praktikum' im LEA-Kurs.

In dem neuen Projekt sollen auch die Beispielprogramme aus dem o.a. Dokument ausprobiert werden.

Löschen Sie danach alle Zeilen in der loop-Funktion,

Integrieren Sie jetzt den Programmcode Ihrer Ampelschaltung **v2-7** in das Programm in Ihrem neuen Projekt v3-ampel-krass. Dafür müssen verschiedene Programmteile kopiert werden (Variablen, Inhalte der setup-Funktion, Inhalte der loop-Funktion).

Sorgen Sie dann dafür, dass die Ampelphase (0 – 9) **nicht mehr auf dem seriellen Monitor**, sondern auf dem **LCD** ausgegeben wird.

Hinweis: Eine Ausgabe sollte nur dann geschehen, wenn der Ausgabewert sich auch ändert. Nicht in jedem Durchgang des loops!

Aufgabe 2 (Zustände schalten / Flankenerkennung):

- Ändern Sie das Programm so, dass eine **Dauer-Rotphase** mit einer **einmaligen** (kurzen) Betätigung des linken Tasters **an- und wieder ausgeschaltet** werden kann.

Hierzu implementieren Sie eine Signal-**Flankenerkennung** mittels der Verwendung einer Hilfsvariable, in der Sie den jeweils letzten Zustand des Tasters (gedrückt / nicht gedrückt) nach der Abfrage speichern, um ihn bei erneuter Betätigung des Tasters als Vergleichswert nutzen zu können. Beachten Sie weiterhin, dass der Taster **entprellt** werden muss.

Nutzen Sie hierzu die Hinweise aus dem Dokument zu Vorlesung 3.



Aufgabe 3 (Programmierung einer elektronischen Eieruhr – v3-eieruhr):



Wir verwenden in dieser Aufgabe als Basis die gleiche Schaltung wie bei der Aufgabe 1 und 2. Erstellen Sie ein neues Arduino-Projekt mit dem Namen 'v3-eieruhr'.

Zusätzlich verschalten Sie einen Lautsprecher an Pin A0 des Arduino prinzipiell genau so, wie die LEDs. Verwenden Sie hier einen 220 Ohm Widerstand. Mit dem Lautsprecher, können simple piepende Töne ausgegeben werden.



Mini Lautsprecherkapsel, ca. 160hm Impedanz, (max. zulässiger Spulenstrom ca 25mA), NICHT MIT SUMMER VERWECHSELN!



Einleitung:

In dieser Aufgabe soll eine elektronische Eieruhr auf dem Mikrocontroller-Board programmiert werden, um die Kochzeit von Frühstückseiern zu überwachen.

Hierzu soll das LCD verwendet werden. Auf diesem soll die Restkochzeit im folgenden Format angezeigt werden :

- m:ss (z.B. 6:35 für 6 Minuten und 35 Sekunden)

Sie können gerne kreativ bei der restlichen Gestaltung des Displays sein... 😊

Nach Ablauf der voreingestellten Kochzeit, sollen die drei Ampel-LEDs blinken und für eine Dauer von 5 Sekunden ein akustisches Signal durch den Lautsprecher ausgegeben werden.



Programmbeschreibung:

Durch die Taster 1 (links) und 2 (rechts) wird die Kochzeit für die Eier vorgewählt. Hierbei ist **keine** Flankenerkennung gefordert.

Nach dem Start der Eieruhrsoftware soll die Anzeige auf 0.00 stehen, ohne dass ein Alarm ertönt. Der Anwender kann nun eine der zwei vorgegebenen Kochzeiten mit den Tasten 1 oder 2 auswählen:

- 1 : perfect egg: 5:20 Kochzeit, grüne LED leuchtet
- 2: slushy egg: 0:15 Kochzeit, rote LED leuchtet

Das 'slushy egg' dient hier natürlich nur zum Testen des Alarms ☺

Nachdem die Kochzeit durch die Taster gewählt wurde, zeigt das Display die Restkochzeit an, die **einmal pro Sekunde** vermindert wird (Countdown-Timer). Wird in dieser Phase erneut eine der Tasten gedrückt, so soll die Eieruhr wieder auf den entsprechenden Startwert zurückgesetzt werden.

Realisieren Sie die **zeitliche Taktung von ca. einer Sekunde** mit Hilfe der Arduino-Funktion **millis()** (ohne delay-Funktion und ohne Zählvariable).

Am Ende der Kochzeit – die Anzeige steht auf 0:00 – sollen alle 3 Ampel-LEDs blinken und es soll zusätzlich für 5 Sekunden ein Alarmsignal mit 500 Hz über den Piezo-Lautsprecher ertönen.

Hierzu dürfen Sie den Aufruf folgender Arduino-Funktion verwenden:

tone(A0, frequency, duration);

Beachten Sie beim Aufruf der **tone** Funktion, dass die **duration** in **Millisekunden** angegeben werden muss. Der Funktionsaufruf hält das Programm nicht an der Stelle an – auch dies sollte berücksichtigt werden, d.h. dass der Aufruf der Funktion **nur einmal** erfolgen sollte.

Es sollten jetzt die folgenden Projekte vorhanden sein:

- V3-ampel-kraass
- V3-eieruhr

Wenn Sie bis hierhin alle Aufgaben erfolgreich bearbeiten konnten, haben Sie den Laborversuch 3 hiermit bestanden.

Viel Erfolg !!!