



E2 Informatik 2

Sommersemester 2025

Lernziele:

Ziel des Praktikums ist die Vermittlung von grundlegenden Kenntnissen über Mikrocontroller und deren Programmierung. Dazu werden im Praktikum einfache Programme für den Mikrocontroller ATmega 328p erstellt, getestet und mögliche Fehler durch systematisches Debugging eingegrenzt und beseitigt.

Ein Schwerpunkt des Praktikums ist es, verschiedene Peripherie-Komponenten des Mikrocontrollers anzusprechen und typische Aufgaben im Controller-Bereich kennen zu lernen. Wir verwenden hierzu einen Mikrocomputer:

- das Development-Board Arduino UNO.

Allgemeine Hinweise:

Das Praktikum umfasst 5 Termine zu jeweils 90 Minuten.

Die erfolgreiche Bearbeitung der Versuche 1-4 ist die Voraussetzung für die Erteilung einer Teilnahmebescheinigung (Testat).

Sie arbeiten in Gruppen von jeweils zwei (in Ausnahmen: drei) Studierenden, nur im Einzelfall bei entsprechender Vorkenntnis kann auch alleine gearbeitet werden.

Machen Sie sich bitte Notizen zu Ihren Versuchen und sichern Sie Ihre Programme, z.B. auf einem USB-Stick oder per E-Mail. Erkenntnisse aus dem Praktikum werden Ihnen auch bei der schriftlichen Modul-Prüfung (Klausur) helfen!

Die Laborversuche bauen aufeinander auf. Daher ist es wichtig, dass Sie Ihre Programme ordentlich kommentieren, damit Sie Teilprogramme (Funktionen) wiederverwenden können.

Sicherheitshinweise im Labor:

Die Mikrocontroller-Boards werden über den USB-Port des PCs versorgt und arbeiten mit einer sehr geringen Spannung von 5 bzw. 3,3 Volt. Daher sind keine besonderen Vorsichtsmaßnahmen zu treffen.

Vermeiden Sie dennoch unbedingt, die elektronischen Bauteile – besonders den Mikrocontroller-Chip – direkt mit Ihren Fingern zu berühren.

Durch elektrostatische Entladungen könnte es zu Defekten der elektronischen Bauteile kommen!



Versuch 1 – Erste Schritte mit dem Arduino

- **Einleitung:**

In diesem ersten Praktikum sollen Sie sich mit dem Mikrocomputer-System **Arduino UNO**, dem darauf verbauten **Mikrocontroller** Atmega328p und der Entwicklungsumgebung (**Arduino IDE**) vertraut machen.

Entnehmen Sie die Informationen zur Erstellung eines neuen Projekts und zur Arbeit mit der Arduino IDE bitte dem Dokument ‚Arduino-IDE.pdf‘, welches in LEA im Ordner ‚Praktikum‘ zur Verfügung steht.

Aufgabe 1: Testen Sie das Board und das Entwicklungssystem

Das folgende Programm zeigt Ihnen bereits, wie Sie eine LED ansteuern (ein-/ausschalten) und eine Programmverzögerung realisieren können.

```
void setup()
{
    pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
{
    digitalWrite(LED_BUILTIN, HIGH);
    delay(1000);
    digitalWrite(LED_BUILTIN, LOW);
    delay(1000);
}
```

Testen Sie dieses Programm, indem Sie den Programmcode in ein neu erstelltes Projekt kopieren, kompilieren und an das Arduino-Board übertragen. Speichern Sie das Projekt unter dem Namen ‚v1-blink‘.

Idealerweise sollte nun eine kleine, eingebaute LED auf dem Board im Sekundentakt blinken. (Also immer eine Sekunde an und eine Sekunde aus).

Aufgabe 2: Testen Sie eine erste Schaltung

Bauen Sie eine Schaltung zur Ansteuerung einer LED an Pin 8 (Vorlage in Vorlesung 1, Folie 17). Der Widerstand vor der LED sollte 220 Ohm betragen. Nutzen Sie die Programmvorlage aus Folie 18, um ihr Programm auf die Ansteuerung einer externen LED an Pin 8 abzuändern.

- Testen Sie dieses Programm auf Funktionalität – es soll jetzt wieder im 'Sekundentakt' blinken.
- Verkürzen Sie nun die Übergabewerte beim Aufruf der delay-Funktion. Beschleunigen Sie zunächst das Blinken etwas und verzehnfachen Sie dann die Ursprungsgeschwindigkeit.
- Erhöhen Sie die Geschwindigkeit des Blinkens noch weiter – welcher Effekt fällt Ihnen ab einer gewissen Geschwindigkeit auf? Notieren Sie im Programmcode als Kommentar bei welchen Übergabewerten für delay Sie dabei angelangt sind.

Aufgabe 3: Testen Sie ein Lauflicht mit 4 LEDs

Erweitern Sie Ihre Schaltung mit drei weiteren LEDs an Pin 9-11 (Vorlage in Vorlesung 1, Folie 19) und speichern Sie das bestehende Projekt unter dem Namen ‚v1-lauflicht‘. Das vorgegebene Programm „v1-lauflicht“ (Folie 20) zeigt Ihnen bereits, wie Sie ein einfaches Lauflicht realisieren können.

- Testen Sie dieses Programm und notieren Sie oben als Kommentar im Programm, wieviele Bytes des Programmspeicherplatzes laut Arduino IDE für diesen Sketch verwendet werden. Diese Angabe bezieht sich auf die Größe des Maschinencodes Ihres Programms (Merke: Programm == Sketch).
- Schreiben Sie **eine Funktion** mit dem Code für das Lauflicht. In der loop-Funktion soll dann nur noch diese zusätzliche Funktion aufgerufen werden.
- Verändern Sie den Code der Funktion, indem Sie eine for-Schleife für die Realisierung des Lauflichts verwenden. Notieren Sie jetzt nochmals, wieviele Bytes für diesen geänderten Sketch im Programmspeicher verwendet werden.



Aufgabe 4: Ein durch Taster gesteuertes Lauflicht mit 4 LEDs

Speichern Sie ihr Projekt unter dem Namen ‚v1-knight-rider‘.

Verschalten Sie zusätzlich **drei Drucktaster** in der Betriebsweise ‚active low‘ an den Pins 7, 6 und 5. In Vorlesung 1, Folie 30 finden Sie eine Vorlage zur korrekten Verschaltung **eines Tasters**.

Hinweis: es werden keine externen Widerstände für die Taster benötigt, da wir die internen Pull-Up Widerstände der digitalen I/O Pins des ATmega 328p Controllers verwenden.

Diese müssen per Programmcode aktiviert werden – wie das grundsätzlich (für einen Port-Pin) geht, sehen Sie in Vorlesung 1, Folie 31.

Dieses Programm kann auch zum Testen eines Tasters an Pin 7 in Kombination mit einer LED an Pin 8 verwendet werden:

- Testen Sie zunächst die Vorlage aus Folie 31 und überprüfen Sie, ob die LED an Pin 8 korrekt schaltet, wenn der Taster an Pin 7 gedrückt wird.
- Erweitern Sie dann die **setup-Funktion** so, dass die Pull-Up-Widerstände für alle drei Taster eingeschaltet werden.
- Nehmen Sie dann die Vorlage in Vorlesung 1, Folie 33 als Grundlage für ihre neue **loop-Schleife**.
- Erweitern Sie den Programmcode jetzt so, dass die folgende finale Funktionalität erreicht wird (siehe hierzu auch Vorlesung 1, Folie 32):
 - (a) Solange der Taster 1 gedrückt wird, bewegt sich ein Lauflicht von *links nach rechts*. D.h. nacheinander werden die LEDs *D1, D2, D3, D4, D1*, usw. angesteuert.
 - (b) Solange der Taster 2 gedrückt wird, bewegt sich ein Lauflicht von *rechts nach links*. D.h. nacheinander werden die LEDs *D4, D3, D2, D1, D4*, usw. angesteuert.
 - (c) Wird der Taster 3 gedrückt gehalten, soll das Lauflicht automatisch nach jedem Durchlauf seine Richtung wechseln – (der „Knight Rider Mode“) siehe folgendes Video, Sekunden 9-14:

<https://www.youtube.com/watch?v=9beWnnpQsZE>

Damit würde sich folgender Ablauf für die LEDs ergeben:

D1, D2, D3, D4, D3, D2, D1, D2, ...

- (d) Werden beide Taster 1 und 2 zusammen gedrückt, sollen alle LEDs (*D1 – D4*) dauerhaft leuchten („Lampentest“).

Viel Erfolg !!!