

## Übungen 2    Arduino Entwicklung

Quellen: Moodle, Internet, [www.arduino.cc](http://www.arduino.cc), Bibliothek

Beachten Sie unbedingt die folgende Regel für alle folgenden Praktika:

- Die Ergebnisse von den Versuchen/Aufgaben müssen den Laborbetreuern präsentiert werden
- Sie sind für die Vorführung und die Protokollierung der erfolgreichen Versuche durch die Laborbetreuer verantwortlich.
- Eine nicht protokollierte Aufgabe gilt als nicht bearbeitet!

Beachten Sie die zu arbeitenden Aufgaben:

- Alle Gruppen bearbeiten die Aufgaben 2.1 und 2.2
- Die Gruppen Lok, Weiche, Schranke bearbeiten die Aufgaben 2.3-2.8
- Die Gruppen RFID und Magnetkontakt bearbeiten die Aufgaben 2.13-2.18

Beachten Sie, dass sich die Hardware-Boards der Teams leicht unterscheidet:

- Zur Programmierung der Controller ist die gleiche Software (Arduino-IDE) zu verwenden.
- Die eigentliche Programmierung der Controller Arduino bzw. ESP32 ist sehr identisch.
  - Der wesentliche Unterschied ist die WLAN Fähigkeit des ESP32-Controllers. Der Arduino verfügt über keine Netzwerk-Verbindungen.
- Die Teams Lok, Schranke und Weiche verwenden einen Arduino.
- Die Teams RFID und Magnetkontakt verwenden eine ESP32.

### **Aufgabe 2.1      Arduino IDE**

Sie brauchen für die Bearbeitung der Aufgaben eine Arduino-IDE. Sie haben die folgende Möglichkeit:

- Sie installieren die Arduino-IDE auf ihrem privaten PC.
- Sie können einen der MACs im Labor 333B verwenden, auf dem die Software bereits installiert ist.

Machen Sie sich mit der installierten Arduino IDE vertraut.

- Was bedeutet IDE?
- Aus welchen drei Teilen besteht ein Arduino Sketch?
- Welche Konfigurationen in der Arduino IDE sind notwendig?
- Wie speichern Sie einen Sketch dauerhaft ab?
- Wie starten Sie einen Sketch auf dem Arduino?

### **Aufgabe 2.2      Arduino Upload in Schaltbild**

Im PDF-Schaltbild des Arduino Uno sollen sie die folgenden Elemente farblich markieren.

- den Reset Schalter des Arduino
- das Kommunikationsmodul des Arduino
- den MicroController auf dem Arduino

- den Kommunikationspfad, über den der Sketch beim Hochladen auf den MicroController übertragen wird

Laden Sie das bearbeitete PDF über Moodle hoch!

## **Aufgaben für die Gruppen Lok, Weiche und Schranke**

### **Aufgabe 2.3      Arduino Hello World**

1. Erstellen Sie mit der Arduino IDE ein erstes Hello-World Arduino Projekt, das „Hello STTS Arduino World“ über den Seriellen Port ausgibt.
2. Verfolgen Sie die Ausgabe mit Hilfe des Serial Monitors aus dem Plugin
3. Woran erkennen Sie, dass gerade ein neues Programm hochgeladen wird?
4. Dokumentieren Sie ihre Entwicklungsschritte!
5. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.4      Arduino Blink**

1. Erstellen Sie ein Programm für das STTS Arduino-System, das eine LED im Sekundentakt blinken lässt.
2. Verwenden Sie in diesem einfachen Programm die Funktion „delay(msec)“!
3. Dokumentieren Sie ihr Vorgehen!
4. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.5      Arduino Switch**

1. Erstellen Sie ein Programm für das STTS-Arduino-System, das die Position eines Tasters und eine DIP-Schalters über die serielle Verbindung ausgibt.
2. Dokumentieren Sie ihr Vorgehen!
3. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.6      Arduino Switch Blink**

1. Erstellen Sie ein Programm für das STTS-Arduino-System, das abhängig vom Zustand des Tasters die LED mit einer Frequenz von 1 Herz blinkt oder ganz ausschaltet!
2. Vermeiden Sie die Verwendung des „delay()“ Befehls!
3. Dokumentieren Sie ihr Vorgehen!
4. Laden Sie ihr Sketch auf Moodle hoch!
5. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.7      Arduino Lauflicht**

1. Erstellen Sie ein Programm für das STTS-Arduino-System, das die 8 LEDs, die mit dem rechten DIP Schalter verbunden, sind in einer Lauflichterkette nacheinander ansteuert.
  - a. Dabei sollen folgenden Eingaben berücksichtigt werden

- i. DIP Schalter 1-7 legt die Anzahl der wandernden LED fest
    1. Wenn der DIP Schalter 1-7 ausgeschaltet ist, soll nur eine LED leuchten
    2. Wenn der DIP Schalter 1-7 eingeschaltet ist, sollen zwei LEDs leuchten
  - ii. DIP Schalter 1-8 legt die Art der Lichterkette fest (Invertierung)
    1. Wenn der DIP Schalter 1-8 ausgeschaltet ist, wandern leuchtende LEDs
    2. Wenn der DIP Schalter 1-8 eingeschaltet ist, wird die Ausgabe invertiert. Das heißt nicht die helle LED wandert, sondern alle LEDs sind an und die dunklen LEDs wandern. Abhängig von DIP Schalter 1-7 (siehe i und ii) wandern dann eine oder zwei „dunkle“ LEDs
  - iii. Wenn ein Taster gedrückt gehalten wird, ändert sich die Laufrichtung.
2. Dokumentieren Sie ihr Vorgehen!
  3. Laden Sie ihr Sketch auf Moodle hoch!
  4. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### Aufgabe 2.8      Arduino „Analoger“ Ausgang - PWM

1. Entwickeln Sie ein Programm, das abhängig vom Binär-Wort der DIP Schaltern die Helligkeit der LED ändert!
2. Können Sie die Auflösung des PWM Ausgangs verändern?
3. Dokumentieren Sie ihr Vorgehen!
4. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

## Aufgaben für die Gruppen RFID und Magnetkontakt

Für die Bearbeitung der Aufgaben mit dem ESP32 sind wenige Vorarbeiten notwendig, da Sie für diese Aufgaben die WLAN-Hardware des ESP32 verwenden sollen. Führen Sie die folgenden Schritte aus:

- Starten Sie die Arduino-IDE und öffnen Sie die Voreinstellungen.
- Tragen Sie im Textfeld „zusätzliche Bordverwalter-URLs“ diese URL ein:
  - [https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)
- Öffnen Sie den „Boardverwalter und Werkzeuge“, suchen Sie nach ESP32 und installieren Sie die benötigte Toolchain.
- Für die Kommunikation mit MQTT installieren Sie sich noch die Bibliothek „PubSubClient Version 2.7.0“ mit dem Bibliothekverwalter. Die Dokumentation finden Sie unter <https://pubsubclient.knolleary.net/>
- Sie müssen eine WLAN Verbindung herstellen. Sie haben die folgenden Möglichkeiten
  - Sie kennen für ihr lokales WLAN-Netzwerk die SSID und das entsprechende Passwort

- Sie sind im STTS-Labor der DHBW und verbinden sich mit der SSID „ITBahn“ und dem zugehörigen Passwort „geheim007“.
- Im Netzwerk ist ein Master-MQTT-Broker (**TODO IP: 192.168.100.39**) erreichbar. Sie können die Nachrichten, die dieser Server erhalten hat mit Hilfe eines Webbrowser beobachten.

### **Aufgabe 2.13      ESP32 Hello World**

1. Erstellen Sie mit der Arduino IDE ein erstes Hello-World Arduino Projekt, das „Hello STTS Arduino World“ über den Seriellen Port ausgibt.
2. Verfolgen Sie die Ausgabe mit Hilfe des Serial Monitors aus dem Plugin
3. Woran erkennen Sie, dass gerade ein neues Programm hochgeladen wird?
4. Dokumentieren Sie ihre Entwicklungsschritte!
5. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.14      ESP32 Blink**

1. Erstellen Sie ein Programm für das STTS Arduino-System, das eine LED im Sekundentakt blinken lässt.
2. Verwenden Sie in diesem einfachen Programm die Funktion „delay(msec)“!
3. Dokumentieren Sie ihr Vorgehen!
4. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.15      ESP32 Switch**

1. Erstellen Sie ein Programm für das STTS-ESP32-System, das die Position eines Tasters über die serielle Verbindung ausgibt.
2. Dokumentieren Sie ihr Vorgehen!
3. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### **Aufgabe 2.16      ESP32 Switch Blink**

1. Erstellen Sie ein Programm für das STTS-ESP32-System, das abhängig vom Zustand des Tasters die LED 1 mit einer Frequenz von 1 Herz blinkt oder ganz ausschaltet! (siehe Datenblatt ESP32)
2. Vermeiden Sie die Verwendung des „delay()“ Befehls!
3. Dokumentieren Sie ihr Vorgehen!
4. Laden Sie ihr Sketch auf Moodle hoch! (unter Aufgabe 2.6)
5. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### Aufgabe 2.17      ESP32 MQTT Nachricht senden

1. Erstellen Sie zuerst ein Programm für das STTS-ESP32-System, das beim Drücken des Tasters T1 eine Nachricht mit dem Inhalt „Hello STTS World“ an das Topic „hello“ auf dem Master MQTT Broker (IP: 141.72.191.235, Port: 1883) schickt.
  - a. Stellen Sie sicher, dass beim kurzen Betätigen des Tasters T1 nur eine Nachricht verschickt wird.
  - b. Stellen Sie sicher, dass beim dauerhaften Drücken des Tasters T1 nur max.2 Nachrichten pro Sekunde (0,5 Hz) verschickt werden.
  - c. Sie können die MQTT-Nachrichten in Ihrem Browser betrachten, in dem Sie sich mit der Webseite [http://141.72.189.70:8080/test\\_client](http://141.72.189.70:8080/test_client) verbinden. Sie müssen dazu entweder in den Laboren der Hochschule oder mit dem VPN verbunden sein.
    - i. Wählen Sie „Connect“
    - ii. Geben Sie unter Subscriptions im Feld „topic filter“ den Topicnamen „hello“ ein und klicken sie auf „Subscribe“
2. Erstellen Sie ein weiteres Programm, das
  - a. abhängig von den betätigten Tastern sollen die LEDs auf dem Board leuchten:
    - i. Ist kein Taster (T1, T2, T3) gedrückt, sind alle LED dunkel.
    - ii. Ist ein Taster gedrückt, soll eine LED1 grün leuchten.
    - iii. Sind zwei Taster gleichzeitig gedrückt, soll die LED2 grün leuchten.
    - iv. Sind drei Taster gleichzeitig gedrückt, soll die LED3 rot leuchten.
  - b. abhängig von den betätigten Tastern Nachrichten an den Master-MQTT-Broker sendet.
    - i. Wenn sich der Zustand eines Schalters ändert, soll immer eine Nachricht geschickt werden.
    - ii. 2 Sekunden nach einer gesendeten Nachricht soll erneut eine neue Nachricht geschickt werden
    - iii. Der Inhalt der Nachricht setzt sich abhängig von den Tastern T1, T2 und T3 aus den folgenden Bestandteilen zusammen:
      1. Gruppennamen
      2. Anzahl der gedrückten Taster
      3. Liste der gedrückten Taster (z.B bei gedrücktem Schalter T1 und T3 die Liste „[T1, T2]“
    - iv. Wenn das Senden der Nachricht fehlschlägt sollen alle LEDs der Boards mit 2 Hz rot blinken!
3. Dokumentieren Sie ihr Vorgehen!
4. Laden Sie ihr Sketch auf Moodle hoch! (unter Aufgabe 2.7)
5. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!

### Aufgabe 2.18      ESP32 „Analoger Ausgang PWM

1. Entwickeln Sie ein Programm, das abhängig vom Binär-Wort der DIP Schaltern die Helligkeit der LED ändert!
2. Können Sie die Auflösung des PWM Ausgangs verändern?
3. Dokumentieren Sie ihr Vorgehen!
4. Präsentieren Sie das Ergebnis den Praktikumsbetreuern!