



Aufgabenblatt 00 Termine: KW 15

Gruppe	
Name(n)	Matrikelnummer(n)

Die Übungen zur Vorlesung „Eingebettete Systeme“ sollen den Einstieg in den Umgang und die Programmierung eines Mikrocontroller-Systems anhand der Lösung kleiner praktischer Aufgabenstellungen ermöglichen, um so das im Rahmen der Vorlesung erworbene methodisch-theoretische Wissen auf praktischen Problemstellungen anzuwenden. Hierfür wird ein Mikrocontroller-Board, ein Arduino DUE-Board, und die für die jeweilige Aufgabenstellung erforderliche Zusatzhardware (Taster, Servos, Display, GPS u. ä.) benötigt.

In den Übungen selbst werden die Lösungen auf realer Hardware (hier wird ein Arduino DUE-Board nebst Peripherie verwendet) erprobt.

Das **Arduino DUE**-Board basiert auf einem **32-bit ARM Cortex-M3 RISC Prozessor**, dem **SAM3X** basiert.

store.arduino.cc/products/arduino-due

Machen Sie sich bitte zunächst mit der **Pinbelegung** des Arduino DUE Boards vertraut, bevor Sie mit der Lösung der Aufgaben beginnen. Entsprechende Übersichten finden Sie Online unter: [Arduino DUE Pinbelegung](#).

Arduino Entwicklungsumgebung

www.arduino.cc/en/Main/Software

Zur Programmierung wird grundsätzlich die Arduino IDE verwendet. Die Installation der Arduino-IDE auf ihrem Rechner – falls noch nicht vorhanden – wird Bestandteil der Aufgabe 0.1 dieses Blattes sein.

Online Hilfe

docs.arduino.cc/language-reference/

Die Online-Referenz des Arduino Framework ist nicht immer perfekt, stellt jedoch eine wertvolle Quelle für grundlegende Information zur Syntax und Semantik enthaltener Funktionen dar. Vereinzelt sind Code-Beispiele enthalten, welche die Verwendung der jeweiligen Funktion darstellen.

Empfehlung: Sollte eine der Funktionen des Arduino Framework unverständlich erscheinen, schlagen Sie diese zuerst in der Online-Referenz nach!

Genereller Hinweis: Speichern Sie die Lösung zu jeder Aufgabenstellung gesondert ab. Dieses erleichtert einerseits die Zuordnung von Aufgabe zu Lösung und andererseits hilft es Ihnen bei der Bearbeitung von Aufgaben, die teilweise über Aufgabenblätter hinweg aufeinander aufbauen. Berücksichtigen Sie bei der Entwicklung der Lösungen auch eine spätere Wiederverwendung des Codes. **Kommentare im Code sind ausdrücklich erwünscht!**

Die grundlegende Programmstruktur eines Arduino Sketches zeigt folgendes Beispiel:

```
// Deklaration/Definition von Variablen mit globaler Sichtbarkeit

uint8_t pin_led = 13;           //Nr. des ext. Pins
bool ledState = LOW;           //Zustandsvariable

void toggle_ledstate() {
    ledState = !ledState;
}

void setup()
{
    // Anweisungen der einmaligen, initialen Konfiguration des Mikrocontrollers

    // Beispiel: Konfiguration des digitalen I/O Anschlusspins
    // mit der Nummer 13 (hier: als Ausgangspin)
    pinMode(pin_led, OUTPUT);
    digitalWrite(pin_led, ledState);
}

void loop()
{
    // Anweisungen, die der Mikrocontroller innerhalb einer Iteration der
    // nicht terminierenden Hauptschleife ausführen soll

    // Beispiel: Wechsel des Pegels am Ausgangspin 13 (400ms-on; 200ms-off)
    toggle_ledstate();
    digitalWrite(pin_led, ledState);
    delay(400);
    toggle_ledstate();
    digitalWrite(pin_led, ledState);
    delay(200);
}
```

Listing 1: Programmstruktur eines Arduino Sketches

Folgende Funktionen werden in diesem Sketch verwendet:

```
* pinMode(<pin>, <mode>)           → pinMode
* digitalWrite(<pin>, <value>)    → digitalWrite
* delay(<ms>)                     → delay
```

Aufgabe 0.1 Installation Arduino-IDE

Ziel dieser Aufgabe ist die Bereitstellung der Voraussetzungen zur Bearbeitung der weiteren Aufgaben der Übung. Hierfür benötigen sie die Arduino-IDE. Die Arduino Software wird für alle gängigen Plattformen bereitgestellt. Auf den Poolrechnern kann sie mit dem Befehl `/informatik/tams/software/arduino-1.8.19/arduino` gestartet werden. Zudem empfehlen wir, immer den selben Poolrechner zu verwenden.

Auf anderen Rechnern folgen Sie der [Anleitung auf der Arduino-Website](#) und installieren sie die aktuelle Version. Hier steht die letzte Version der Arduino IDE 1.8.X, die Version 1.8.19, oder die aktuelle Version der Arduino IDE 2 zur Verfügung. Beide Varianten sind für die ES-Aufgaben gleichermaßen geeignet. Der Vorteil der IDE 2 liegt in der Möglichkeit des Hardware-Debuggings, was allerdings Boardseitig ein ST-Link Interface voraussetzt, welches der verwendete Arduino DUE nicht zur Verfügung stellt.

Wir raten grundsätzlich davon ab, Arduino aus den Ubuntu-Paketquellen zu installieren.

Stellen Sie sicher, dass das Arduino DUE-Board unterstützt wird (Tools->Board->Board Manager: Arduino ARM (32-bits) Boards).