



Aufgabenblatt 4 Termine: 19.05./22.05.

Gruppe	
Name(n)	Matrikelnummer(n)

Kommunikation mit Peripherie jeglicher Art gehört zum festen Bestandteil des Funktionsumfanges eingebetteter Systeme. Ihre Aufgabe ist es die Lösungen vorheriger Aufgaben über die Einbindung von Kommunikationsschnittstellen auf mehrere Mikrocontroller zu verteilen. Für die Lösung der Aufgaben werden jeweils zwei Mikrocontroller benötigt. Arbeiten Sie bitte gemeinsam mit einer anderen Gruppe zusammen und teilen Sie die Arbeit sinnvoll untereinander auf.

Aufgabe 4.1

Entwerfen Sie ein Programm zum Austausch von Nachrichten zwischen den beiden Mikrocontrollern unter Verwendung der seriellen Schnittstelle. Verbinden Sie die beiden Systeme, indem Sie (**jeweils gekreuzt**) die TX und RX Anschlüsse sowie den GND Anschluss der beiden Mikrocontroller miteinander verbinden. Schliessen Sie an eines der Systeme einen Taster an und an das andere eine LED an (bzw. verwenden Sie die intern verdrahtete LED). Entwerfen Sie zunächst ein einfaches Kommunikationsprotokoll, das Ihnen folgenden Funktionsumfang bereitstellt:

1. Einmalige Betätigung des Tasters (1. Mikrocontroller) regelt mittels PWM die Intensität der LED (2. Mikrocontroller) hoch (bis zum Wert 255) und daraufhin runter (bis zum Wert 0). Das Interval für das Hoch- und Herunterregeln soll jeweils ca. 5 Sekunden betragen.
2. Ist eine der beiden Grenzen erreicht worden, so soll der 2. Mikrocontroller dem steuernden Kommunikationspartner dieses explizit mitteilen, indem die auf dem 1. Mikrocontroller integrierte LED 3x zum Blinken gebracht wird.

Achtung: Das Arduino Due Board bietet insgesamt 4 serielle Anschlüsse an. Da die erste serielle Schnittstelle (TX0 und RX0) für die Übertragung des Programmcodes sowie die allgemeine Kommunikation mit dem angeschlossenen Rechner verwendet wird, wird jedoch empfohlen von der Verwendung dieser Schnittstelle abzusehen.

Aufgabe 4.2

Verwenden Sie für die Lösung dieser Aufgabe den seriellen I²C (Inter-Integrated Circuit) Bus zur Anbindung der beiden Mikrocontroller. Informieren Sie sich über die Funktionsweise von I²C:

- Wikipedia: en.wikipedia.org/wiki/I2C, bzw. de.wikipedia.org/wiki/I2C
- oder tiefergehend unter: www.i2c-bus.org/de/i2c-bus

Das allgemeine Anschlussschema ist in Abbildung 1 skizziert. Lokalisieren Sie die Anschlüsse für die I²C Signale SCL und SDA auf dem jeweiligen Board und stellen Sie eine Verbindung durch **direkte** Verdrahtung her. **Achten Sie bitte darauf die Kabellängen möglichst gering zu halten**, da sonst die Verwendung von Pull-Up Widerständen für jede Signalleitung notwendig wird, um eine nutzbare Signalqualität zu ermöglichen. Vergessen Sie bitte darüber hinaus nicht die GND Anschlüsse beider Boards zu verbinden.

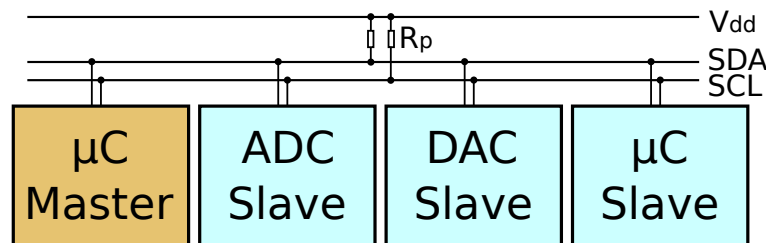


Abbildung 1: Allgemeines I²C Anschlussschema.

Verwenden Sie zur Ansteuerung der I²C Schnittstelle die **Wire** Funktionsbibliothek (`#include <Wire.h>`), die Bestandteil der Arduino Bibliotheken ist. Machen Sie sich mit dem Funktionsumfang dieser Bibliothek vertraut (www.arduino.cc/en/Reference/Wire). Schauen Sie sich insbesondere die folgenden Funktionen an:

* <code>Wire.requestFrom()</code>	arduino.cc/en/Reference/WireRequestFrom
* <code>Wire.beginTransmission()</code>	arduino.cc/en/Reference/WireBeginTransmission
* <code>Wire.endTransmission()</code>	arduino.cc/en/Reference/WireEndTransmission
* <code>Wire.onReceive()</code>	arduino.cc/en/Reference/WireOnReceive
* <code>Wire.onRequest()</code>	arduino.cc/en/Reference/WireOnRequest

Machen Sie einen der Mikrocontroller zum *Bus-Master* und den anderen zum *Bus-Slave* und entwerfen Sie für die Boards jeweils ein Programm, dass ein Kommunikationsprotokoll implementiert, welches folgenden Funktionsumfang anbietet:

1. Der Bus-Master soll die integrierte LED des Bus-Slaves im 2-Sekunden-Takt ein- und ausschalten.
2. Für Debugging-Zwecke soll das Resultat jeder Ein- bzw. Ausschaltaktion vom Bus-Master beim Slave angefragt und das Ergebnis auf über den seriellen Monitor ausgegeben werden.

Aufgabe 4.3

Basierend auf der vorherigen Aufgabe, gilt es eine neue Lösung für die Aufgabenstellung 3.3 zu implementieren, bei der das Slave-Board mit dem Gyroskop-Modul verbunden wird und der Bus-Master mit dem anzusteuernenden Servo. Entwerfen Sie dazu ein Kommunikationsprotokoll, das dem Bus-Master ermöglicht beim Bus-Slave die Winkelgeschwindigkeit um eine von Ihnen gewählte Achse des Gyroskops abzufragen.