

# Kommunikation über USART und TWI

---

## 1 Ziel der Aufgabe:

In dieser Laborübung soll der Umgang mit den Schnittstellen des Mikrocontrollers erlernt werden. Dazu wird ein Beschleunigungssensor der Firma Bosch eingesetzt. Dieser kommuniziert mit dem Mikrocontroller über den TWI-Bus. Um die Daten zu veranschaulichen, sollen diese an einen PC über USART übermittelt werden. Auf dem PC wird eine in MATLAB geschriebene Oberfläche verwendet, welche bereits existiert.

Damit eine Übersichtlichkeit im Programm gegeben ist, soll in verschiedenen Files gearbeitet werden. Da eine gesamte Implementierung zu umfangreich wäre, werden einige Funktionen zur Verfügung gestellt.

## 2 Vorbereitung für die Aufgabe:

- Informieren über den TWI-Bus (I<sup>2</sup>C-Bus):
- Informieren über die serielle Schnittstelle USART  
Wie funktioniert die Schnittstelle auf dem STK 500?
- Installieren der MATLAB Compiler Runtime für Release R2012a:  
<http://www.mathworks.de/products/compiler/mcr/index.html>
- Informieren über die Funktionsweise eines Beschleunigungssensors und die Register des BMA020
- Erstellen eines AVR-Projekts mit dem Namen „Beschleunigungssensor“
- Austauschen des Files Beschleunigungssensor.c mit dem gegebenen File
- Einbinden aller Quelldateien
- Auseinandersetzen mit den Quelldateien

## 3 Aufgabe:

### 3.1 TWI-Bus

Für den TWI-Bus ist bereits eine große Anzahl von Funktionen gegeben. Wie der Headerdatei TWI.h jedoch zu entnehmen ist, soll es später nur drei Zugriffsfunktionen geben.

```
void TWI_InitTWI(uint8_t BitRate, uint8_t PrescalerHighBit, uint8_t PrescalerLowBit);  
uint8_t TWI_RecvData(uint8_t SlaveAdress, uint8_t RegisterAdress);  
void TWI_SendData(uint8_t SlaveAdress, uint8_t RegisterAdress, uint8_t Data);
```

- Mit der ersten Funktion soll das TWI-Modul initialisiert werden. Berechnen Sie nach der Vorschrift aus dem Datenblatt, welche Werte beim Aufruf der Funktion in der main-Funktion übergeben werden müssen. Es soll eine Frequenz von 100kHz gewählt werden.
- Schreiben Sie die Funktionen TWI\_SendData und TWI\_RecvData mit Hilfe der bereitgestellten Funktionen. Verwenden Sie in der Funktion TWI\_RecvData die Funktion TWI\_RepeatedStartCondition.

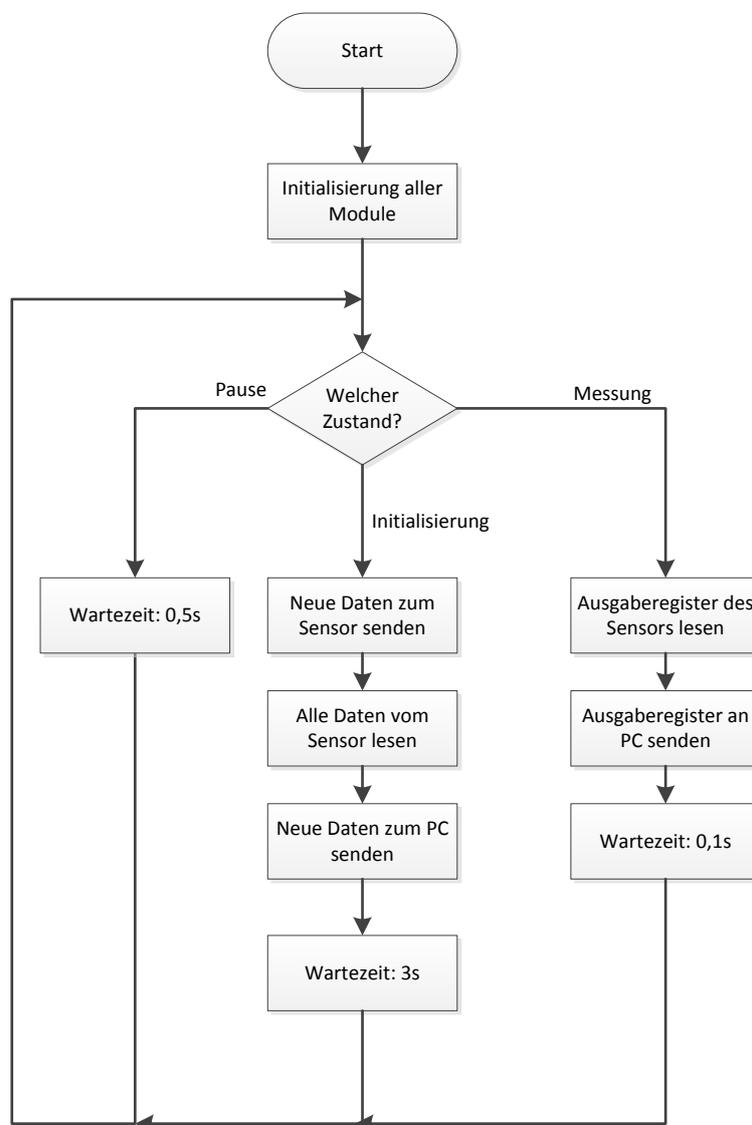
### 3.2 USART-Kommunikation

Für die USART-Kommunikation sind nur die Namen der Funktionen gegeben. Schreiben Sie die beiden Funktionen im File USART.c. Alle notwendigen Angaben finden Sie in den Kommentaren zu den Funktionen.

Die Interrupt-Service-Routine ist bereits im File Beschleunigungssensor.c implementiert.

### 3.3 main-Funktion

Implementieren Sie den folgenden Programmablaufplan in der main-Funktion. Machen Sie sich Gedanken darüber, ob es sinnvoll ist die verschiedenen Zustände durch unterschiedliche LEDs anzeigen zu lassen. Die benötigten Funktionen finden Sie in dem File DATA\_SYNC.c.



Nun kann das gesamte Programm mit der Oberfläche auf dem PC getestet werden.