

## Aufgabenblatt 05 Termine: KW 20

Gruppe	
Name(n)	Matrikelnummer(n)

### 5 Elektrische Ansteuerung eines Displays

Flüssigkristalldisplays (LCDs) sind ein häufig verwendetes Ausgabegerät für eingebettete Systeme. Entsprechend des Bedarfes werden einfache Zeichen-Displays oder auch grafische Displays eingesetzt. Für die späteren Aufgaben der Übungen zu ES werden Sie ein grafisches 1.8" TFT-LCD (Thin-film-transistor liquid-crystal display) verwenden, um darauf grundlegende Funktionalitäten zur visuellen Ausgabe zu implementieren.

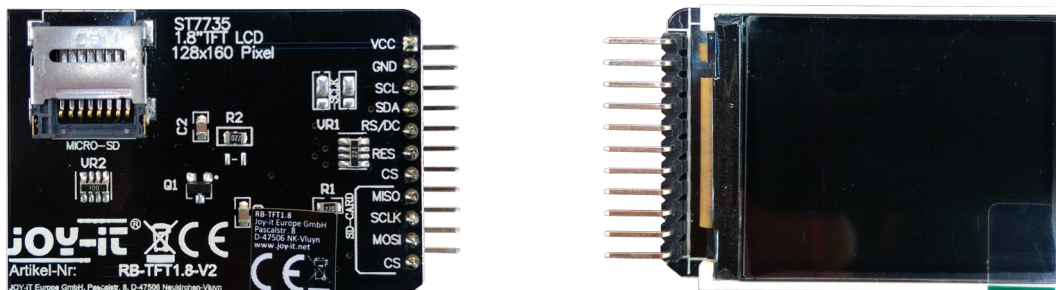


Abbildung 1: 1.8" TFT-Display mit zusätzlichem SD-Card-Slot.

Die Schnittstelle dieses Displays wird durch den Displaycontroller (Sitronix ST 7735R) bereitgestellt. Setzen Sie sich mit dem **Datenblatt** des Displaycontrollers auseinander. Der Displaycontroller verfügt über eine parallele, eine serielle 3-Draht und eine serielle 4-Draht Schnittstelle. Über die äußere Beschaltung des Displaycontrollers auf dem Breakout-Board des in den ES Übungen verwendeten Displays (siehe Abb. 1) ist die serielle 4-Draht Schnittstelle hardwareseitig fest konfiguriert. Abb. 2 zeigt das vereinfachte Timing-Diagramm dieser Schnittstelle.

#### Aufgabe 5.1 Diskussion eines Anschlussbeispiels

Diskutieren Sie das in Abb. 3 skizzierte Beispiel zum Anschluss des Displays an einen Arduino.

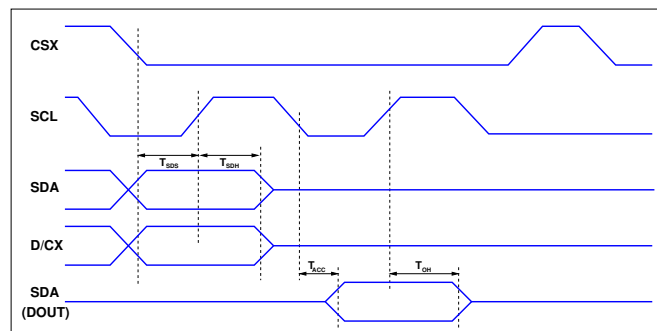


Abbildung 2: Vereinfachtes Timing-Diagramm der 4-Draht Schnittstelle des ST7735R; Leitungen: CSX – Chip-Select, SCL – Serial Clock Line (Taktsignal), SDA – Serial Data Line (Datenleitung), D/CX – Data/Command; Timingangaben:  $T_{SDS}$  – Data setup time,  $T_{SDH}$  – Data hold time,  $T_{ACC}$  – Access time,  $T_{OH}$  – Output disable time (vgl. [Datenblatt ST7735R](#), S.25)

Für erste Überlegungen gehen Sie davon aus, dass für die Anschlüsse des Displays seitens des Arduinos normale digitale I/O-Pins verwendet werden.

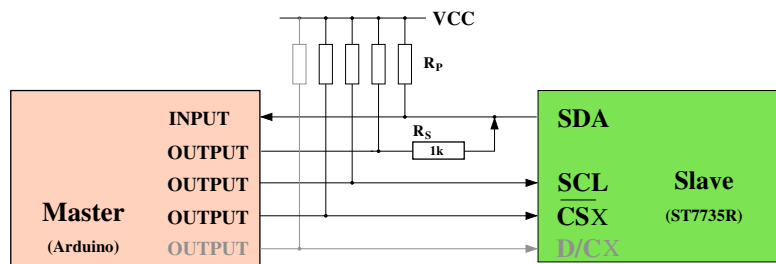


Abbildung 3: Anschlussvorschlag des Displays mit ST7735R-Controller

- Wäre diese Beschaltung grundsätzlich möglich und sinnvoll?
- Welche Bedeutung hätte dann der Widerstand  $R_S$ ?
- Könnte er auch fortgelassen werden?
- Was können Sie über die Dimensionierung von  $R_S$  aussagen?
- Welche Bedeutung haben die Pullup-Widerstände  $R_P$ ?
- Können die Pullup-Widerstände fortgelassen werden?  
Und falls ja, welche und unter welchen Voraussetzungen?

Die in eingebetteten Systemen eingesetzten Prozessoren stellen i. d. R. verschiedene serielle Schnittstellen, die für die Ansteuerung von Sensoren und Aktuatoren üblicherweise benötigt

werden, bereits zur Verfügung. So stellen auch die Prozessoren der Arduino-Boards folgende **serielle Protokolle** bereit:

- **UART**: Universal Asynchronous Receiver/Transmitter  
Beispielsweise **RS-232**: Ein Kommunikationspartner pro Schnittstelle, asynchroner Datentransfer
- **I<sup>2</sup>C**: Inter-Integrated Circuit  
Serieller Bus, (multi-) **Master/Slave** Kommunikation, synchroner Datentransfer
- **SPI**: Serial Peripheral Interface  
Serieller Bus, **Master/Slave** Kommunikation, synchroner Datentransfer

### **Aufgabe 5.2**      Serielle Busprotokolle

Machen Sie sich mit den o. g. seriellen Schnittstellen und ihren (Bus)Protokollen vertraut. Die referenzierten Wikipedia-Seiten bieten schon einen guten Überblick.

Lässt sich eine dieser Schnittstellen zur Ansteuerung des Displays über die serielle 4-Draht Schnittstelle des ST 7735R-Displaycontrollers (siehe Abb. 2) verwenden? Sie werden feststellen, dass keine dieser Schnittstellen auf den ersten Blick mit der seriellen 4-Draht Schnittstelle des ST 7735R-Displaycontrollers kompatibel ist. Berücksichtigen Sie bei Ihrer Einschätzung nicht nur die Hardwareverbindungen, sondern auch die Protokolle der Schnittstellen.

Welche dieser Schnittstellen halten sie ggf. für geeignet, das in den ES-Übungen verwendete Display anzusteuern. Begründen Sie Ihre Entscheidung.