

# Praktikum Eingebettete Systeme

WS 2015/16

## Übungsblatt 1

**Abnahme:** 03.11.2015 um 15:15 Uhr

Hinweise: Alle nachfolgenden Aufgaben beziehen sich auf die in Abbildung 3 bis 5 gezeigte Hardware. Diese steht Ihnen im Hardwarelabor P535 zur Verfügung. Auf den Laborrechner ist für die Lösung der Praktikumsaufgaben die Arduino IDE installiert.

### Aufgabe 1: Fragen im Vorfeld

1. Was versteht man unter "Prellen eines Tasters" und welche Problematik liegt dem Prolleffekt zugrunde? Wie können Sie diesem Effekt entgegenwirken? Gehen Sie dabei kurz auf eine Hard- und eine Softwarelösungen ein.
2. Sind die LEDs 1 bis 3 auf dem IoT-Shield aus Abbildung 3 high- oder lowaktiv? Welche Auswirkung hat die Antwort auf die Programmierung?
3. Die 7-Segment-Anzeige auf dem Board ist vom Typ: "Common Cathode". Was verbirgt sich hinter diesem Sachverhalt?
4. Das IoT-Shield verfügt an der Stelle "LDR" über einen Fotowiderstand. Erklären Sie die grundlegende Anschluss- und Funktionsweise.
5. Erläutern Sie die groben Unterschiede zwischen einem Mikrocontroller und einem Mikroprozessor-System.

### Aufgabe 2: LED Binärzähler

1. Erstellen Sie ein Arduino-Programm<sup>1</sup>, welches die 3 LEDs des IoT-Shields permanent zum Leuchten bringt.
2. Erweitern Sie das Programm so, dass Sie jetzt mit dem Taster S1 einen Binärzähler von  $000_b$  bis  $111_b$  über die 3 LEDs realisieren. Mit jedem Tastendruck soll der Zähler lediglich um 1 inkrementiert werden. Bei Überlauf soll der Zähler wieder bei 0 beginnen.

Im Zusammenhang zu dieser Aufgabe führen Sie eine (Software)-Entprellung des Tasters mittels zeitlicher Insensitivität durch<sup>2</sup>.

<sup>1</sup>Ein Programm im Arduino-Umfeld wird auch als "sketch" bezeichnet.

<sup>2</sup>Beachten Sie die bereits vorhandene, hardwareseitige Entprellung der Taster auf dem IoT-Shield

### Aufgabe 3: Ansteuerung der 7-Segment-Anzeige

Auf dem IoT - Shield ist eine 7-Segmentanzeige mit gemeinsamer Kathode vorhanden. Implementieren Sie die Ansteuerung dieser 7-Segmentanzeige und verknüpfen Sie schließlich den LED Binärzähler mit der Selbigen.

Um eine einheitliche Betrachtung der 7-Segmentanzeige vorauszusetzen, halten Sie sich bitte an die in Abbildung 1 und 2 aufgezeigten Konventionen:

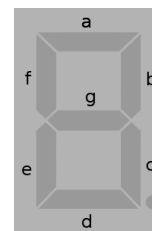


Abbildung 1: Segment-Definition

Anzeige	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

Abbildung 2: Dezimal zu Bit-Array-Definitionen

## **Aufgabe 4: Ansteuerung des Fotowiderstands**

Am Analogport 2 des Arduino wird über das IoT-Shield ein Fotowiderstand herausgeführt (IoT-Shield LDR-Port).

Vergewissern Sie sich, dass dieser Fotowiderstand vorhanden ist.

Realisieren Sie eine sequenzielle Abfrage dieses Analogwertes im Sekundentakt und geben Sie den Helligkeitswert prozentual über die serielle Schnittstelle aus<sup>3</sup>.

Hinweis: Der Arduino verfügt bereits über einen USB «->» UART Übersetzer. Die Textausgabe erfolgt somit direkt über den USB Port, den Sie bereits zum Beschreiben des Arduino nutzen.

## **Aufgabe 5: Zusammenfassende Übung**

Implementieren Sie einen Helligkeitsmesser wie in Aufgabe 3 mit dem Zusatz, dass die gemessene Helligkeit in zehn linear, proportionale Stufen eingeteilt und auf der einstelligen 7-Segmentanzeige angezeigt wird. Die Messpunkte sollen zudem nicht mehr im Sekundentakt erhoben werden, sondern nun bei Bedarf per Taster S1 (prellfrei).

---

<sup>3</sup>100% max. hell, 0% max. dunkel

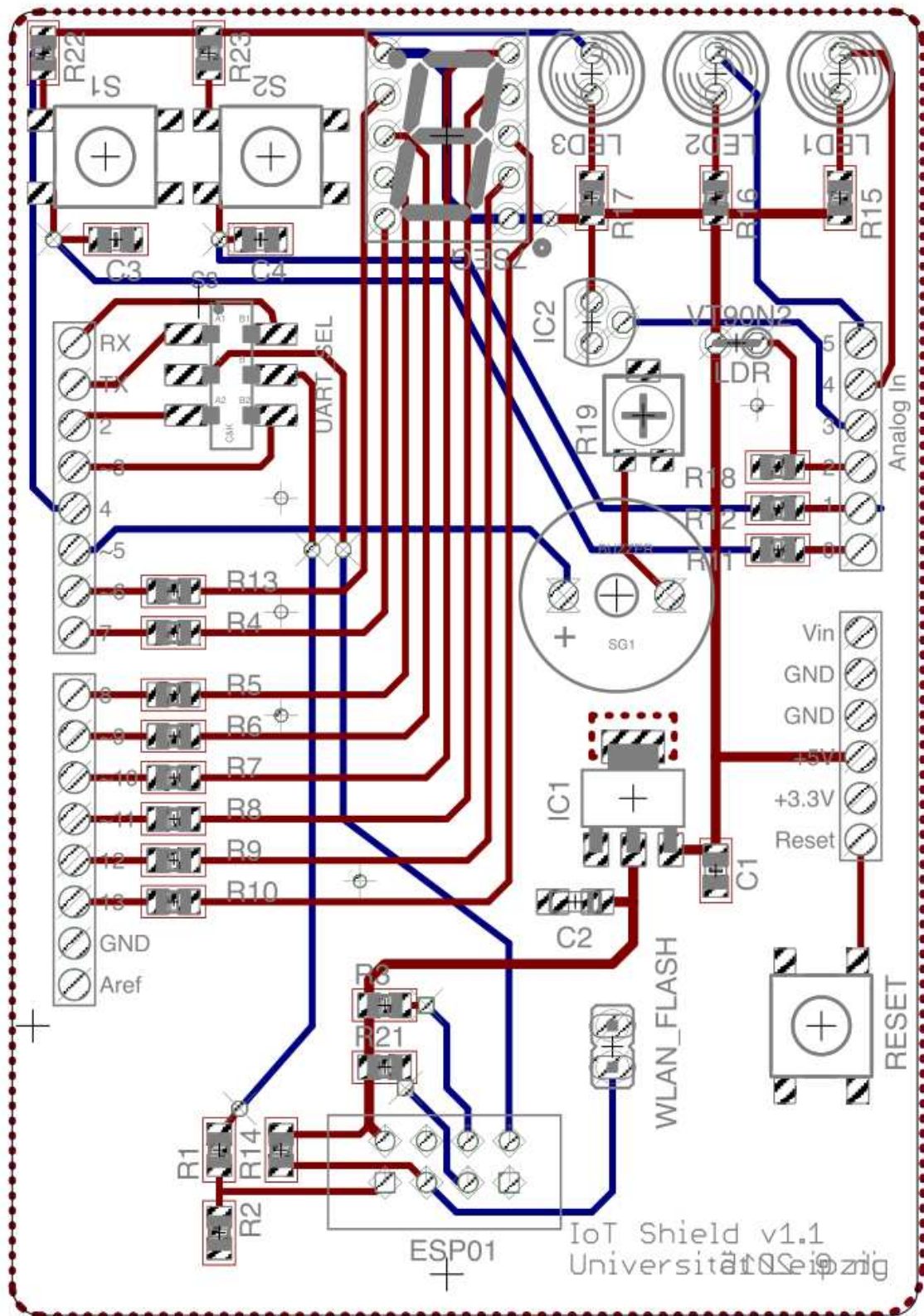


Abbildung 3: IoT Shield schematic

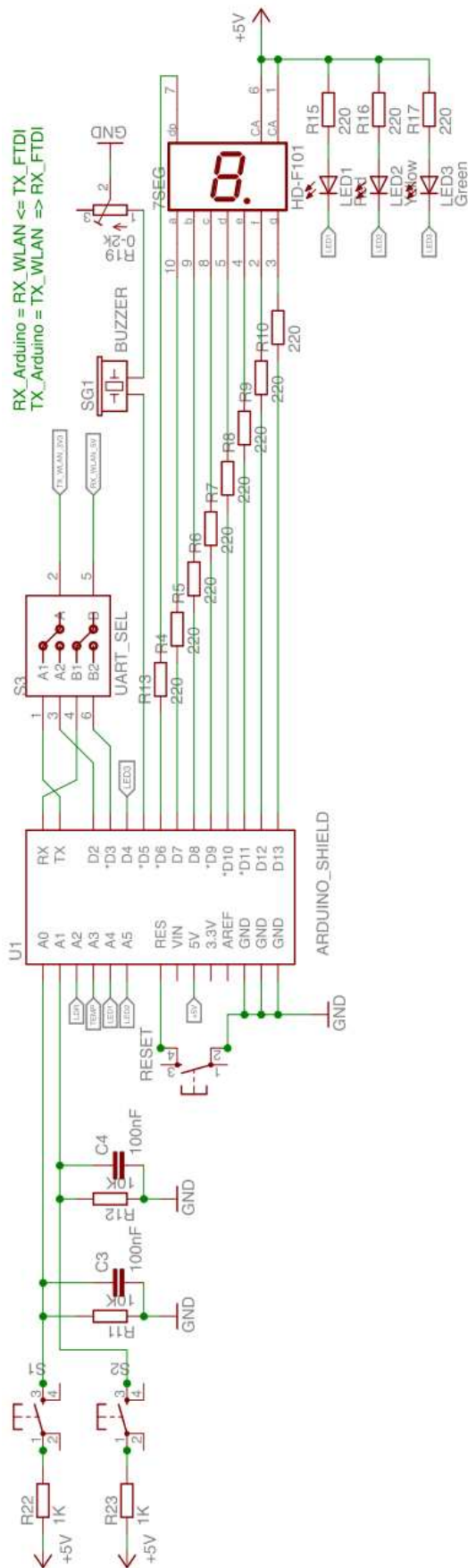


Abbildung 4: IoT Shield Pinout

# THE DEFINITIVE ARDUINO UNO PINOUT DIAGRAM

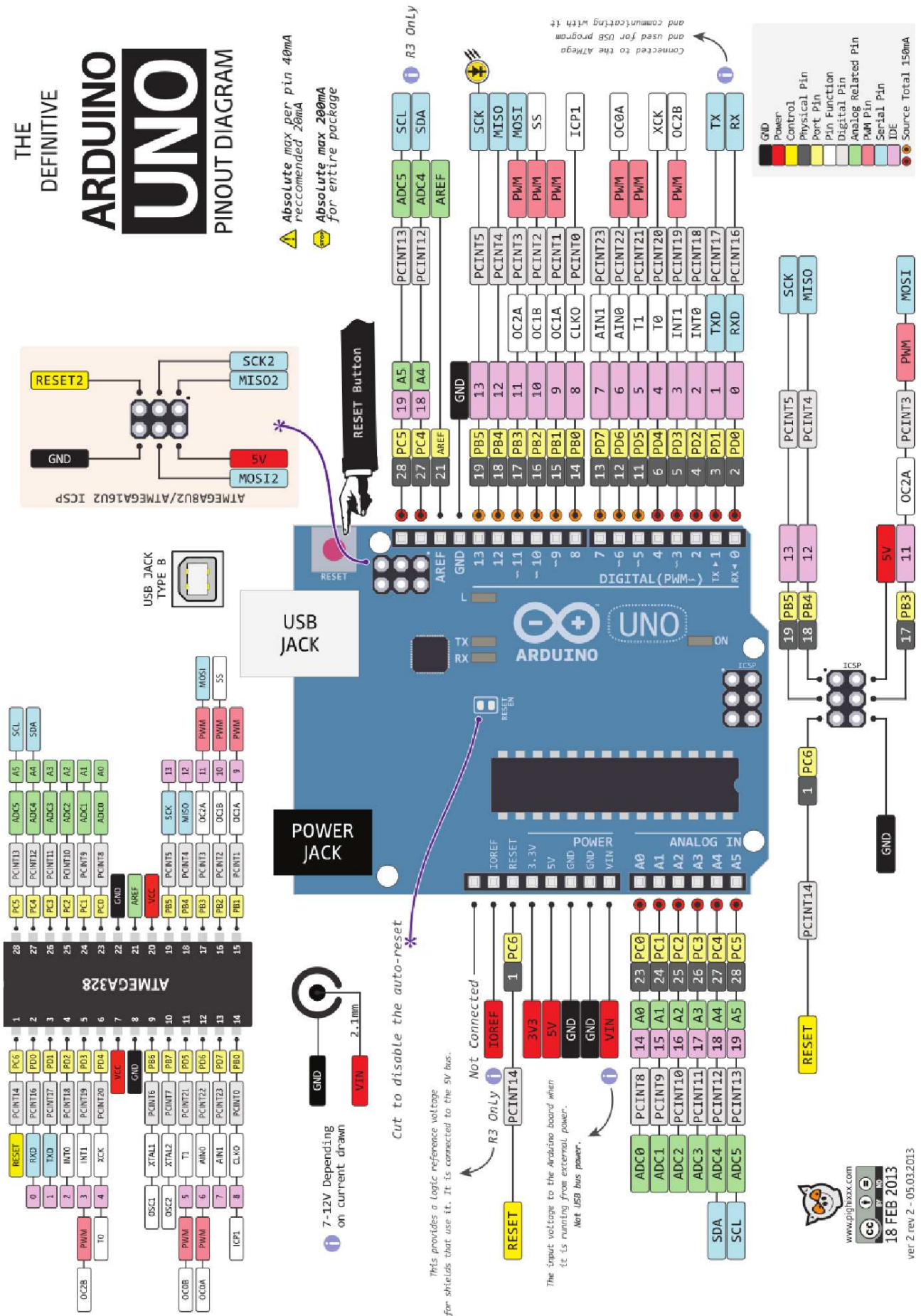


Abbildung 5: Arduino Uno Pinout Diagram