

Mikroprozessoren Praktikumsbericht

Abgabe 2: 7 Segment Anzeige und AD Wandler



VRINDA PRAKASH SARAF
768043

- ✎ **Schriftliche Aufgabe V-1:** Ermitteln Sie aus dem Datenblatt des μC die garantierte minimale Ausgangsspannung für einen High-Pegel am Ausgang bei einem Ausgangsstrom von 6 mA und bei einem Ausgangsstrom von 20 mA. Geben Sie die verwendete Tabelle an.

Tabelle auf der Seite 94.

Table 55. Output voltage characteristics

Symbol	Parameter	Conditions	Min	Max	Unit
$V_{OL}^{(1)}$	Output low level voltage for an I/O pin	CMOS port ⁽²⁾ $I_{IO} = +8 \text{ mA}$ $2.7 \text{ V} \leq V_{DD} \leq 3.6 \text{ V}$	-	0.4	V
$V_{OH}^{(3)}$	Output high level voltage for an I/O pin		$V_{DD}-0.4$	-	
$V_{OL}^{(1)}$	Output low level voltage for an I/O pin	TTL port ⁽²⁾ $I_{IO} = +8 \text{ mA}$ $2.7 \text{ V} \leq V_{DD} \leq 3.6 \text{ V}$	-	0.4	V
$V_{OH}^{(3)}$	Output high level voltage for an I/O pin		2.4	-	
$V_{OL}^{(1)}$	Output low level voltage for an I/O pin	$I_{IO} = +20 \text{ mA}$ $2.7 \text{ V} \leq V_{DD} \leq 3.6 \text{ V}$	-	1.3 ⁽⁴⁾	V
$V_{OH}^{(3)}$	Output high level voltage for an I/O pin		$V_{DD}-1.3$ ⁽⁴⁾	-	
$V_{OL}^{(1)}$	Output low level voltage for an I/O pin	$I_{IO} = +6 \text{ mA}$ $1.8 \text{ V} \leq V_{DD} \leq 3.6 \text{ V}$	-	0.4 ⁽⁴⁾	V
$V_{OH}^{(3)}$	Output high level voltage for an I/O pin		$V_{DD}-0.4$ ⁽⁴⁾	-	
$V_{OL}^{(1)}$	Output low level voltage for an I/O pin	$I_{IO} = +4 \text{ mA}$ $1.7 \text{ V} \leq V_{DD} \leq 3.6 \text{ V}$	-	0.4 ⁽⁵⁾	V
$V_{OH}^{(3)}$	Output high level voltage for an I/O pin		$V_{DD}-0.4$ ⁽⁵⁾	-	

2.1

$V_1 :-$

Bei unserem Board

$$V_{DD} = 3,3 \text{ V}$$

$$\begin{aligned} 6 \text{ mA} :- \quad V_{\min} &= V_{DD} - 0,4 \text{ V} \\ &= 3,3 - 0,4 \\ &= 2,9 \text{ V} \end{aligned}$$

$$\begin{aligned} 20 \text{ mA} :- \quad V_{\min} &= V_{DD} - 1,3 \text{ V} \\ &= 3,3 - 1,3 \text{ V} \\ &= 2 \text{ V} \end{aligned}$$

- ✎ **Schriftliche Aufgabe V-2:** Speziell bei unserem Anwendungsfall „Ansteuern der Segmente“ würde auch ein ungepuffertes Schieberegister für die Bereitstellung der zusätzlichen Ausgänge genügen. Überlegen und beschreiben Sie, was bei der Verwendung eines ungepufferten Schieberegisters am oben beschriebenen Ablauf geändert werden muss.

V2

Verwendung eines ungepufferten Schieberegisters :-

- i) Anzeigestelle „n“ wird deaktiviert
- ii) „n“ wird auf neuen Wert $(n+1) \% 4$ getestet.
- iii) Neues Segmentmuster wird in Register geladen und ~~ist~~ warten.
- iv) Der Counter wartet bis 7 takt wo die Werte in alle Register geladen.
- v) Anzeigestellen „n“ wird aktiviert.
- vi) Abwarten und dann prozess wiederholen

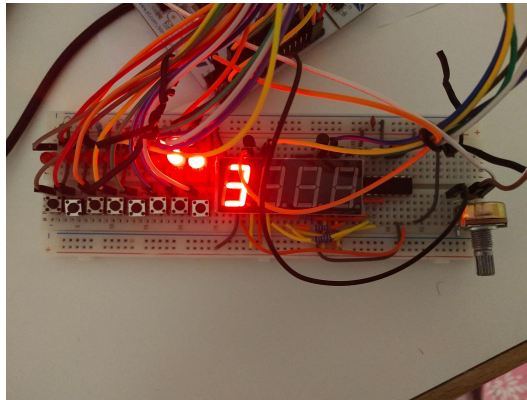
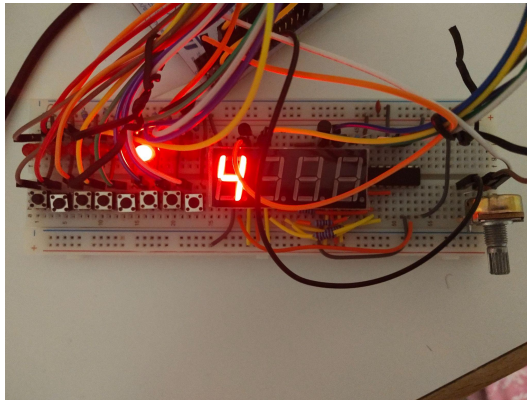
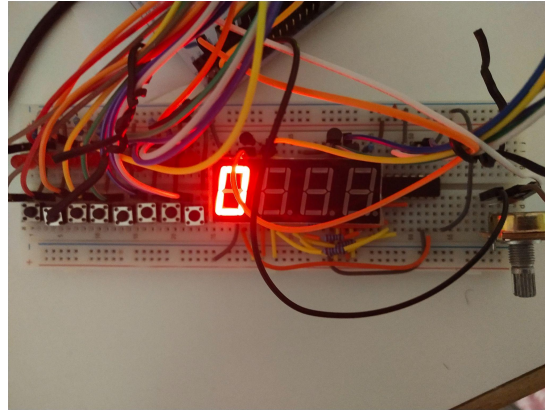
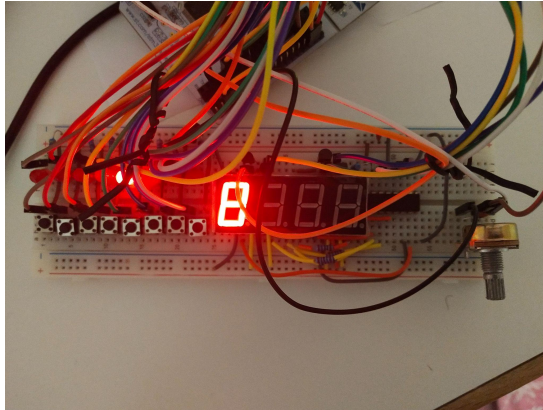
- ✎ **Schriftliche Aufgabe V-3:** Erstellen Sie eine Tabelle, die dokumentiert, welches Bit (0 bis 7) des Wertes, den Sie `ssdSegment`s zuweisen, welchem Segment der Anzeige („a“ bis „g“ bzw. Dezimalpunkt) zugeordnet ist.

V3

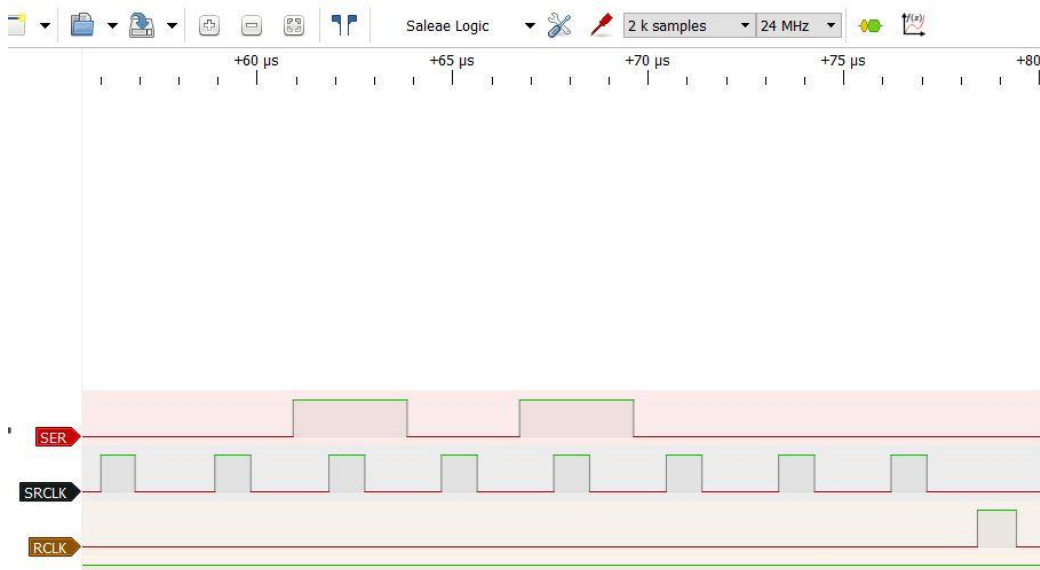
a	Bit 0 → b
f . b	Bit 1 → d
e g c	Bit 2 → e
d . c	Bit 3 → g
	Bit 4 → c
	Bit 5 → . (dezimalpunkt)
	Bit 6 → f
	Bit 7 → a

✎ **Schriftliche Aufgabe V-4:** Dokumentieren Sie Ihren erfolgreichen Test durch 4 Aufnahmen, **auf denen auch Ihr Studentenausweis (lesbar!)** zu sehen ist. Die Aufnahmen müssen die Tests für die letzten vier Ziffern Ihrer Matrikelnummer dokumentieren.

8043 :-



- ✎ **Schriftliche Aufgabe V-5:** Nehmen Sie eine Messung für die Darstellung der Ziffer „0“ vor (keine Taste gedrückt). Kopieren Sie einen Screen-Shot des vollständigen PulseView-Fensters in Ihre Labordokumentation. Bestimmen Sie aus Ihrer Messung die Zeit, die benötigt wird, um neue Werte an die Ausgänge des Schieberegisters zu übertragen. Wie können Sie sinnvoll abschätzen, welche Zeit für das Einlesen der Taster benötigt wird? Wie könnten Sie die Zeit genauer bestimmen?



5.1

Buffer Register befüllen:
ca. $(78 \mu s - 56 \mu s) \approx 22 \mu s \Rightarrow$ Zeit vom erstem
SRCLK bis RCLK

Das Einlesen der Taster (`key.read()`) benötigt weniger Zeit als $22 \mu s$. Das schreiben in das Register nimmt natürlich viel Zeit.

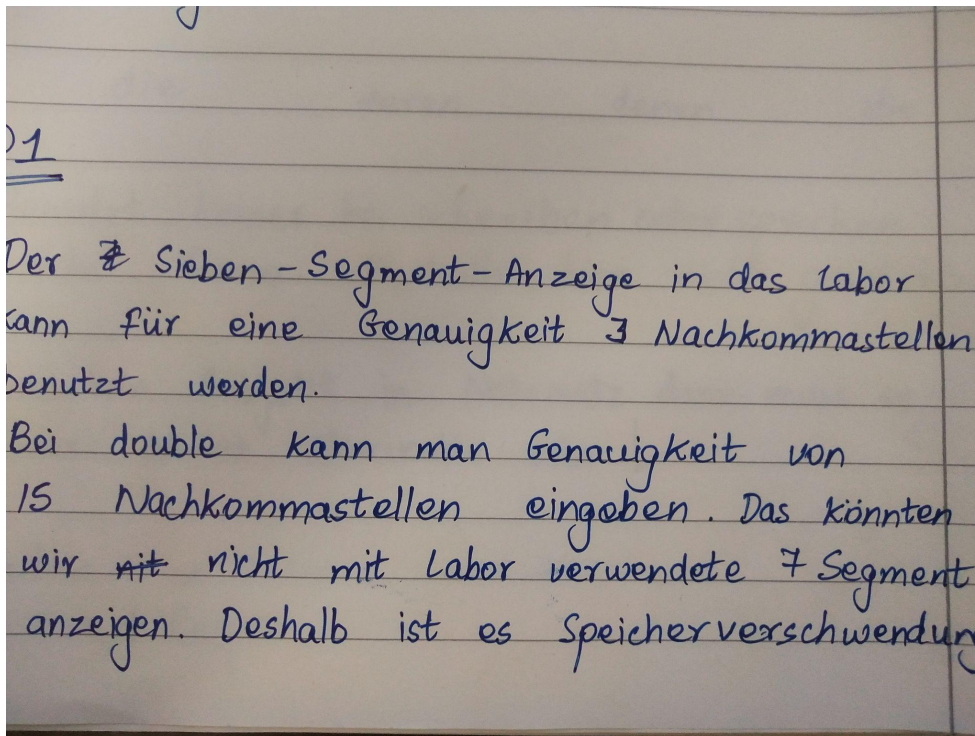
5.2

Die genauer Zeit ~~kann~~ für Zustandsabfrage könnte man mit das Logic Analyser ~~erfassen~~ erledigen.

- ① Zeit wo das LED leuchtet
- ② Zeit wann das Taster gedrückt
- ③ Dann die differenz zwischen beide.

Natürlich gibts hier Störungen bzw. Verzögerungen in Leitungen.

- ✎ **Schriftliche Aufgabe D-1:** Begründen Sie (vollständige Sätze!), warum es sinnlos ist, den Parameter von `setValue` als `double` zu deklarieren.



- ✎ **Schriftliche Aufgabe D-2:** Speicher in einem Mikrocontroller ist endlich. Berechnen Sie mit Hilfe des Datenblatts, wie viele Werte Sie in Ihrem gleitenden Mittelwertfilter maximal speichern könnten, wenn Ihnen das gesamte RAM ausschließlich für diesen Zweck zur Verfügung stünde.

Reference Manual Seite 213-

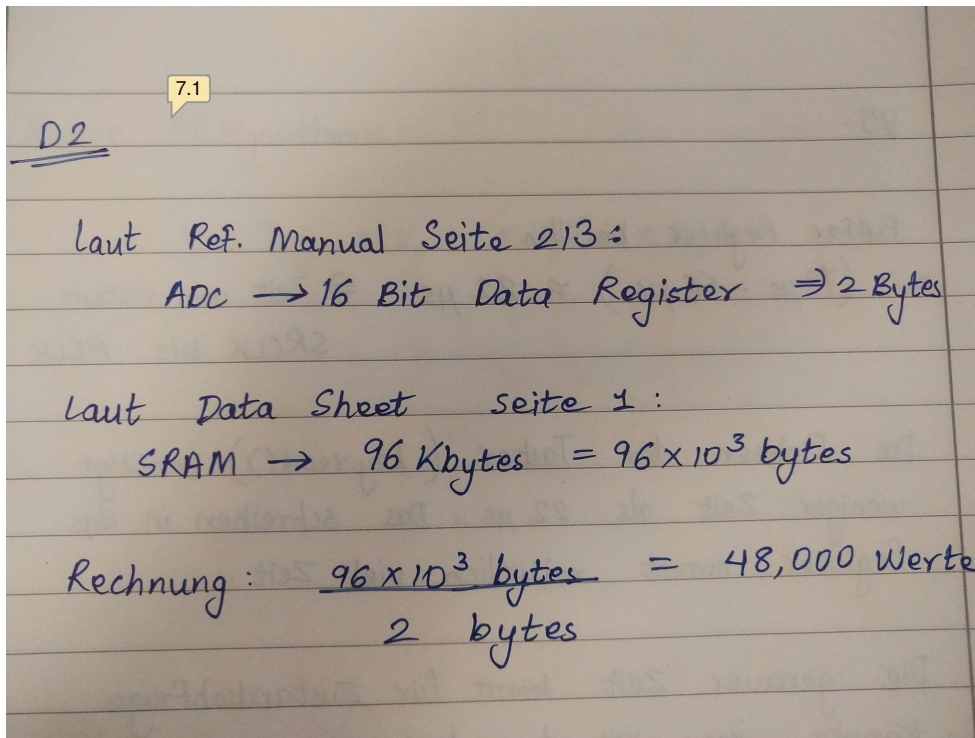
11.1 ADC introduction

The 12-bit ADC is a successive approximation analog-to-digital converter. It has up to 19 multiplexed channels allowing it to measure signals from 16 external sources, two internal sources, and the V_{BAT} channel. The A/D conversion of the channels can be performed in single, continuous, scan or discontinuous mode. The result of the ADC is stored into a left- or right-aligned 16-bit data register.

The analog watchdog feature allows the application to detect if the input voltage goes beyond the user-defined, higher or lower thresholds.

Data Sheet Seite 1 :-

- Memories
 - up to 512 Kbytes of Flash memory
 - up to 96 Kbytes of SRAM



✎ **Schriftliche Aufgabe D-3:** Häufig gibt es im Umfeld der Embedded-Programmierung Vorgaben, dass Puffergrößen Zweierpotenzen sein sollen. Welchen arithmetischen Ausdruck in Ihrer Implementierung ^{7.2} können Sie durch welche Bit-Operation ersetzen (und damit effizienter machen) wenn die Größe des Puffers immer eine Zweierpotenz wäre?

----> `float calculated_average_volt = (calculated_average >> exponent) * 3.3f`
>> Diese Bit operation dividiert. Der exponent nimmt hier den Wert 16.

✎ **Schriftliche Aufgabe D-4:** Geben Sie an, bei welchem Wert für N die Anzeige bei Ihnen stabil wird.

----> In meiner Implementierung bleibt die Anzeige stabil bei einem Wert von N = 5000 ^{7.3}

✎ **Schriftliche Aufgabe D-5:** Geben Sie an, bei welchem Wert für den Faktor die Anzeige bei Ihnen stabil wird.

----> In meiner Implementierung bleibt die Anzeige stabil bei den Faktor = 0.01

Index der Kommentare

- 2.1 im Prinzip ja. Als Minimalwert berechne ich mit VDDmin allerdings in beiden Fällen 1,4 V
- 5.1 das ist richtig. Das Einlesen der Taster geht etwas schnelle. Wie viele Mikrosekunden etwa?
- 5.2 Die Zeit zum Einlesen der Taster ist unabhängig davon, ob ein Taster gedrückt wird oder nicht. Messen kann man sie z.B. wie oben. Man zeichnet mit und ohne ("auskommentiert") die Anweisung zum Einlesen der Taster auf und misst die Zeit zwischen 2 Übertragungen ans Schieberegister. Aus der Differenz "mit" und "ohne" erhält man einen Wert für das Einlesen der Taster. Tasten drücken muss man dabei natürlich nicht.
- 7.1 1k Byte entspricht hier 1024 Byte
- 7.2 geht auch was bei der Modulo-Rechnung?
- 7.3 müsste aber schon bei einem kleineren Wert stabil werden.