

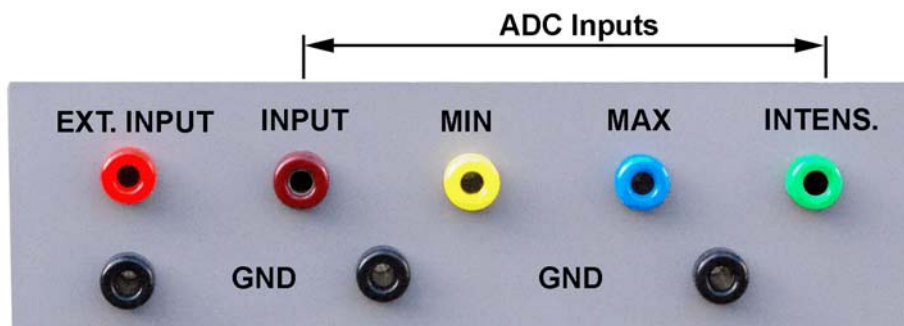
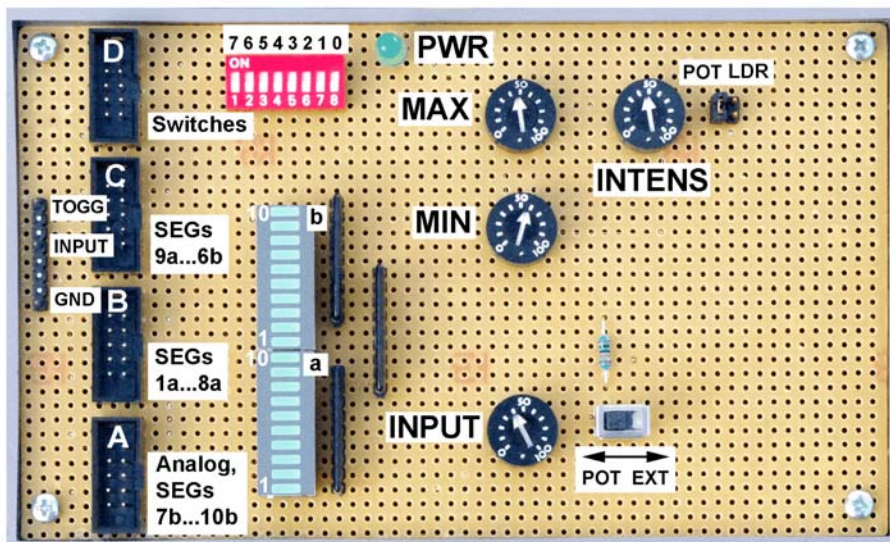
Grundlagen der Bargraphanzeige Der Balkenadapter 13

23. 10. 13

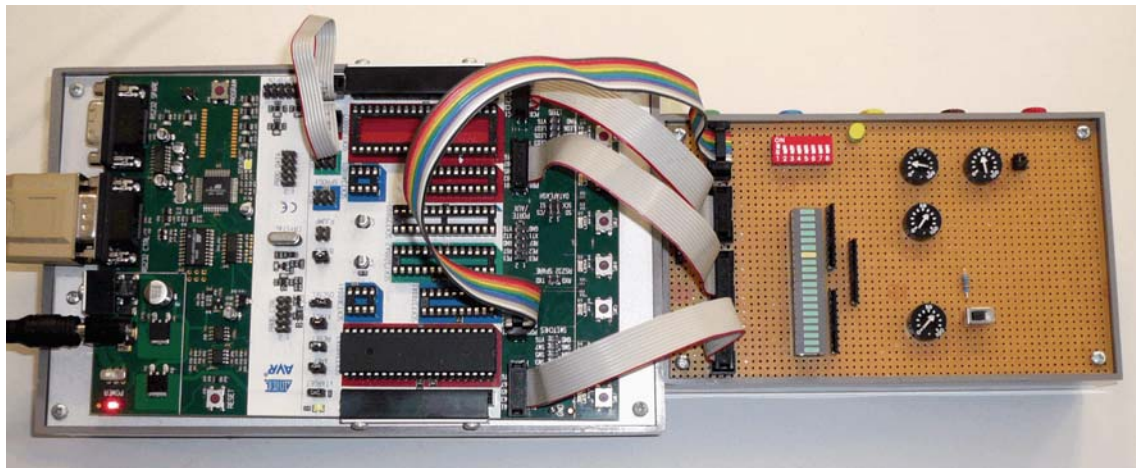
Zweck:

Erprobung der Leuchtbalkenanzeige (Bargraph) mit Mikrocontroller ATmega.

- 1 analoges Eingangssignal (INPUT),
- 2 analoge Grenzwerte (min, max),
- 1 analoges Eingangssignal zur Helligkeitseinstellung,
- 1 Digitalausgang (Meßpunkt) zur Zeitmessung mittels Oszilloskop,
- serielle Schnittstelle,
- 8 Konfigurationsschalter (gemeinsam mit der seriellen Schnittstelle und dem Digitalausgang an einem Port¹),
- maximal 20 Leuchtbalken (LEDs),
- alle Analogsignale mit Trimpotentiometern einzustellen,
- zusätzliche Buchsen für alle Analogsignale.



1: Schalter AUS (OFF), wenn ein Portsignal anderweitig genutzt werden soll.



Die Anzeige zwischen Minimum und Maximum

Es ist ein aktueller Eingangswert anzuzeigen. Er liegt zwischen einem minimalen und einem maximalen Wert (min, max).

- Wert = min ergibt Anzeige Null (gar nichts).
- Wert = max ergibt die größtmögliche Anzeige (alle Leuchtbalken (Full Scale Bars)).

Damit Dreisatzrechnung:

$$\text{max} - \text{min} : \text{FS_Bars} = \text{Wert} - \text{min} : \text{Anzeige}$$

$$\frac{\text{max} - \text{min}}{\text{FS_Bars}} = \frac{\text{Wert} - \text{min}}{\text{Anzeige}}$$

$$\text{Anzeige} = \frac{(\text{Wert} - \text{min}) \cdot \text{FS_Bars}}{\text{max} - \text{min}}$$

Die Wandlerauflösung (FS_ADC) und die Referenzspannung (Vref) spielen keine Rolle (Don't Cares).

Die Anzeige über den vollen Eingangsspannungshub des AD-Wandlers

Hier wird die gesamte Auflösung des AD-Wandlers ausgenutzt.

min = 0 V, max = Wandlerauflösung (Full Scale ADC).

Damit Dreisatzrechnung:

$$FS_ADC : FS_Bars = Wert : Anzeige$$

$$\frac{FS_ADC}{FS_Bars} = \frac{Wert}{Anzeige}$$

$$Anzeige = \frac{Wert \cdot FS_Bars}{FS_ADC}$$

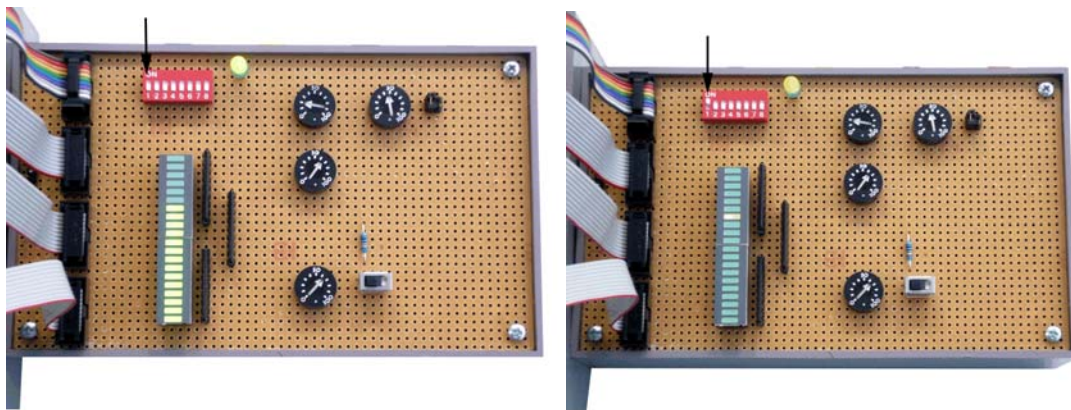
Die Referenzspannung (V_{ref}) spielt keine Rolle (Don't Care).

Sonderfälle der Anzeige:

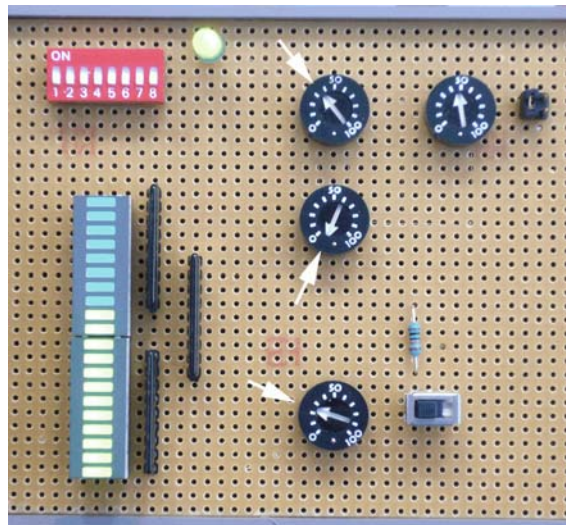
- Wenn min > max: Anzeige über die volle Wanderauflösung (FS_ ADC).
- Wenn Wert < min: Anzeige Wert null (gar nichts).
- Wenn Wert > max: Anzeige Größtwert (FS_Bars; alle Balken oder Leuchtpunkt ganz oben).

Art der Anzeige:

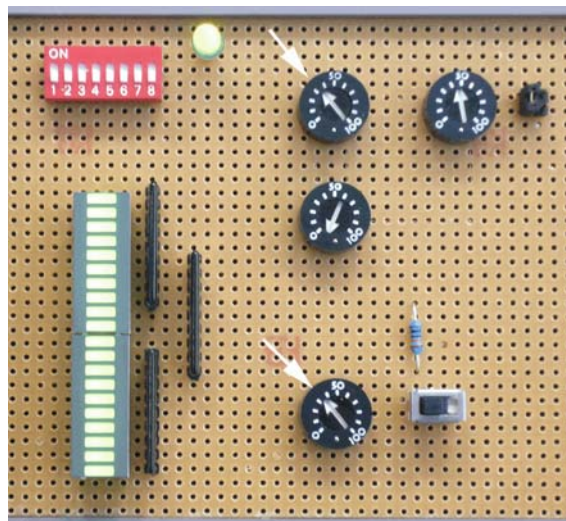
- als Balken (SW7 aus = High),
- als Leuchtpunkt (SW7 ein = Low).



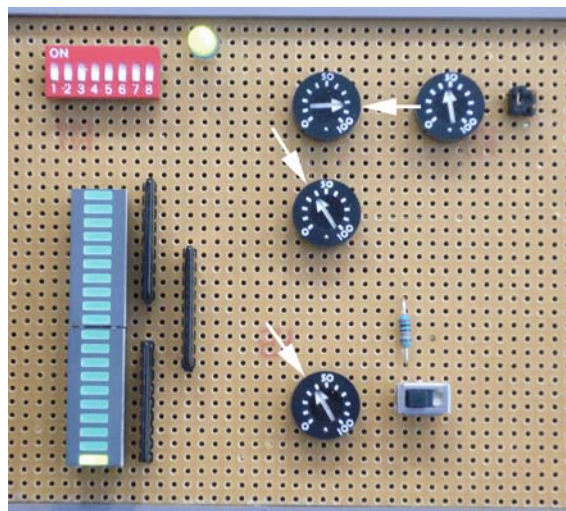
$\min < \max$. Eingangspegel dazwischen:



$\min < \max$. Eingangspegel etwa = max:



$\min < \max$. Eingangspegel etwa = min:



Programmschritte:

1. Initialisieren.
2. max und min einlesen.
3. Wert einlesen.
4. Zur Anzeige wandeln, ausgeben.
5. Zurück zu 2. (Endlosschleife.)

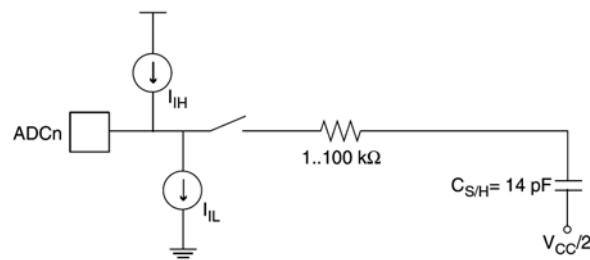
Der erste Versuch (einfach und schmucklos):

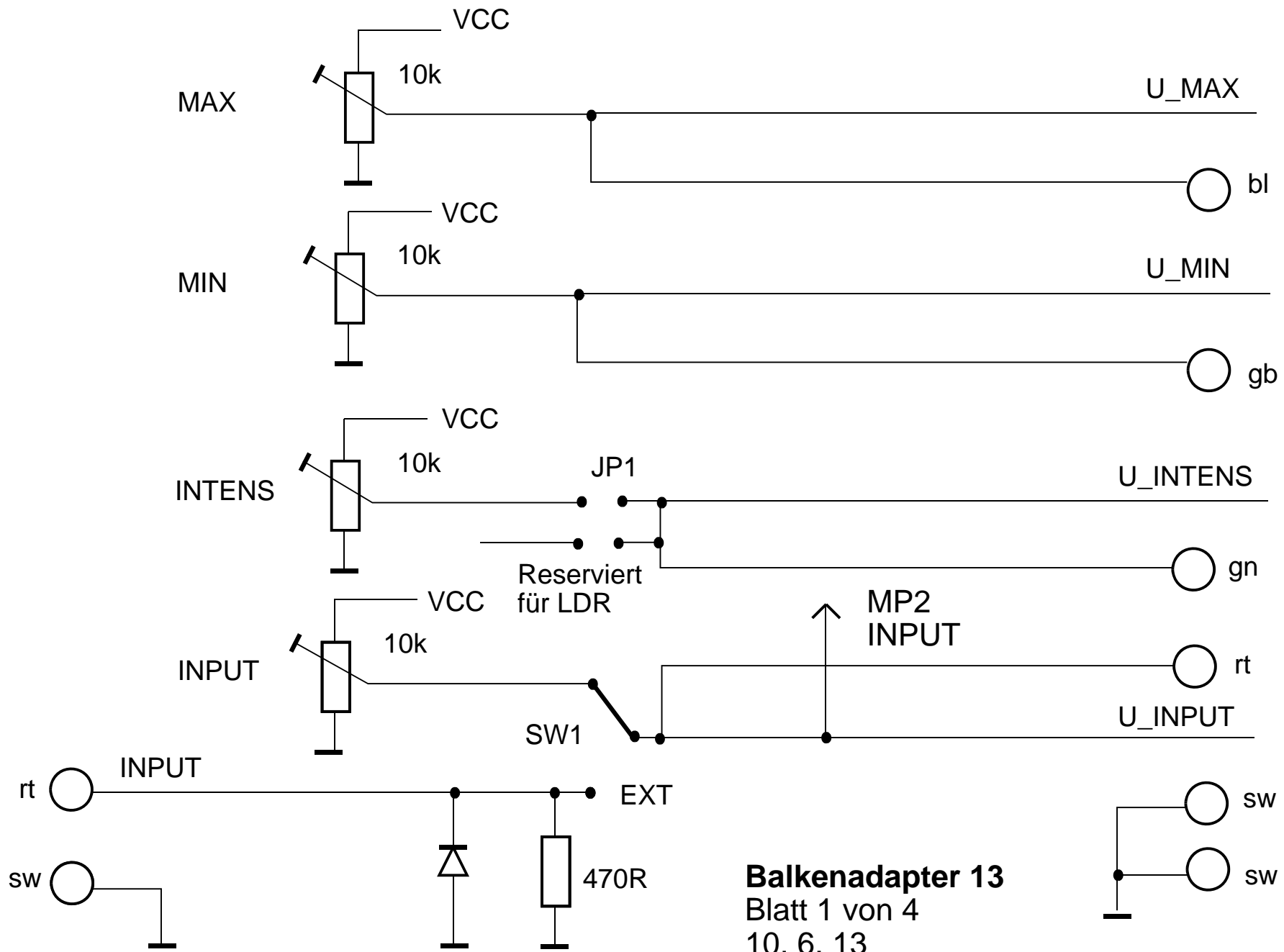
AVR GNU C ohne Optimierung. 4 MHz. Alle 20 Leuchtpunkte werden ausgenutzt. Ein Schleifendurchlauf dauert etwa 3 ms, also ca. 300 Abtastungen/s. Verdreifachung, wenn min und max Konstanten sind, also der obige Schritt 2 aus der Schleife herausgenommen wird.

Ein Grundproblem:

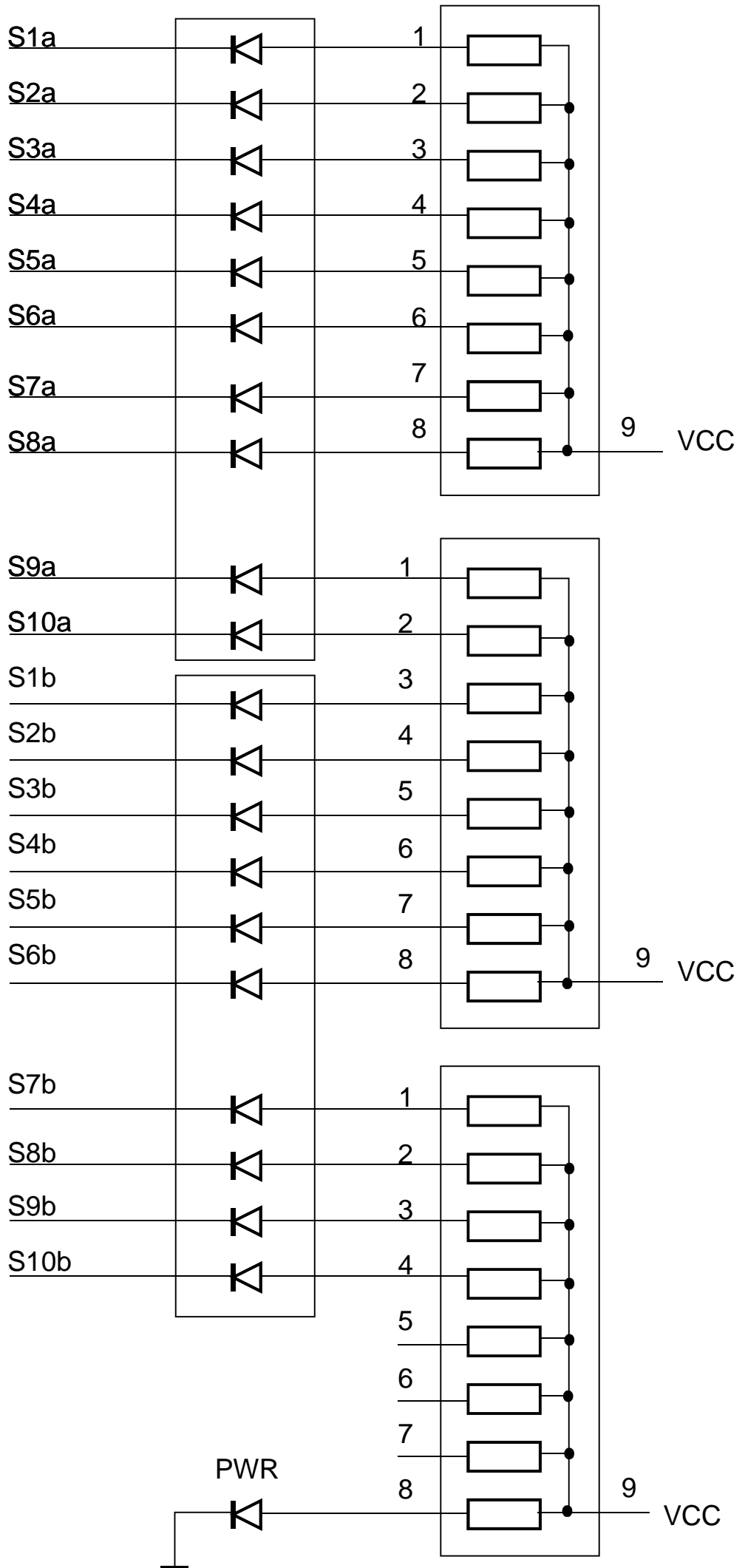
Der ADC will Quellen mit niedriger Impedanz sehen. 10k liegen schon an der oberen Grenze. Ist die Quellimpedanz höher, muß die Abtastzeit soweit erhöht werden, daß die Quelle in der Lage ist, den Kondensator des Abtast- und Haltegliedres unzuladen. Im Ernstfall dürfte es nicht ohne Puffer gehen.

Ein typischer ADC-Eingang (Atmel):

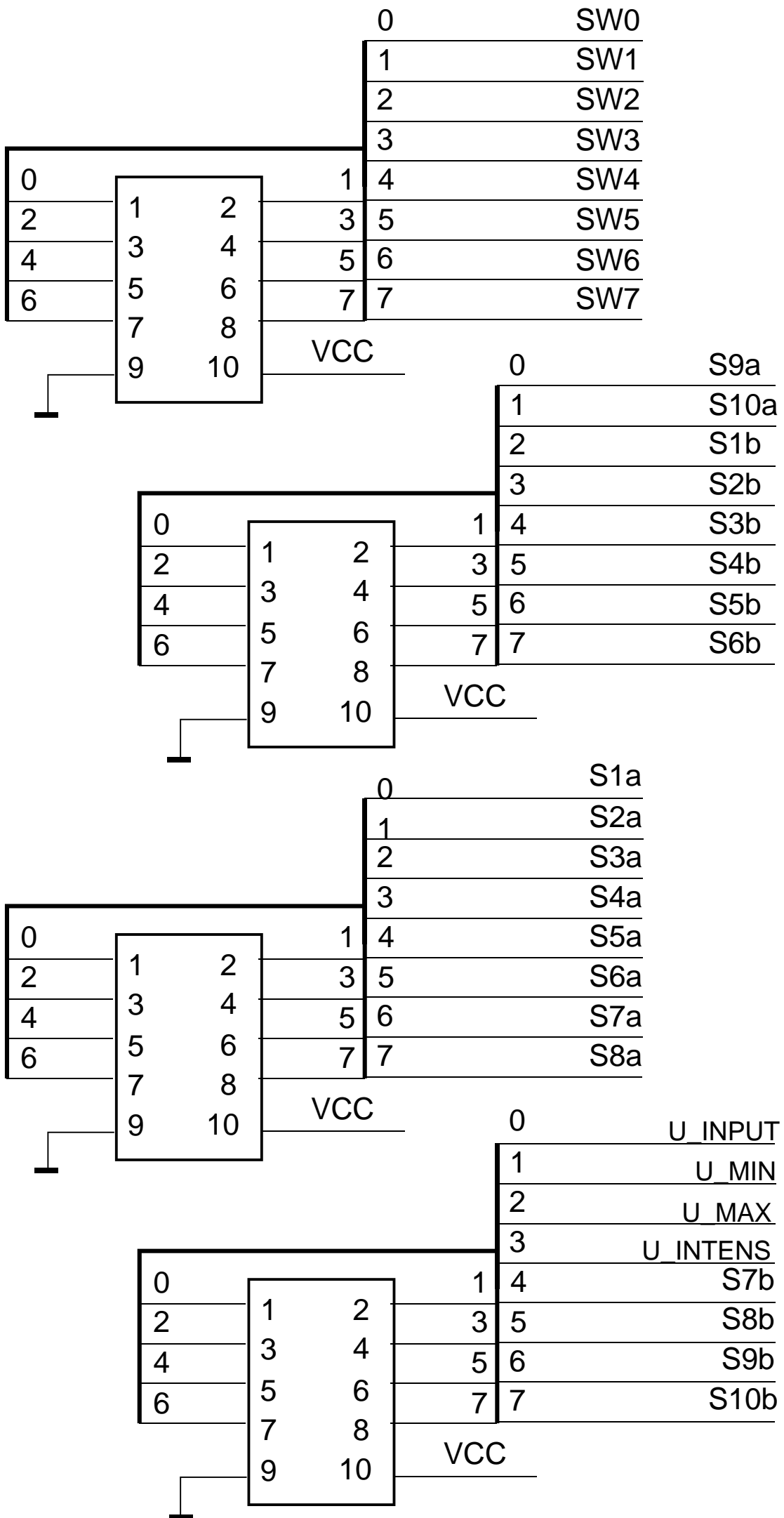


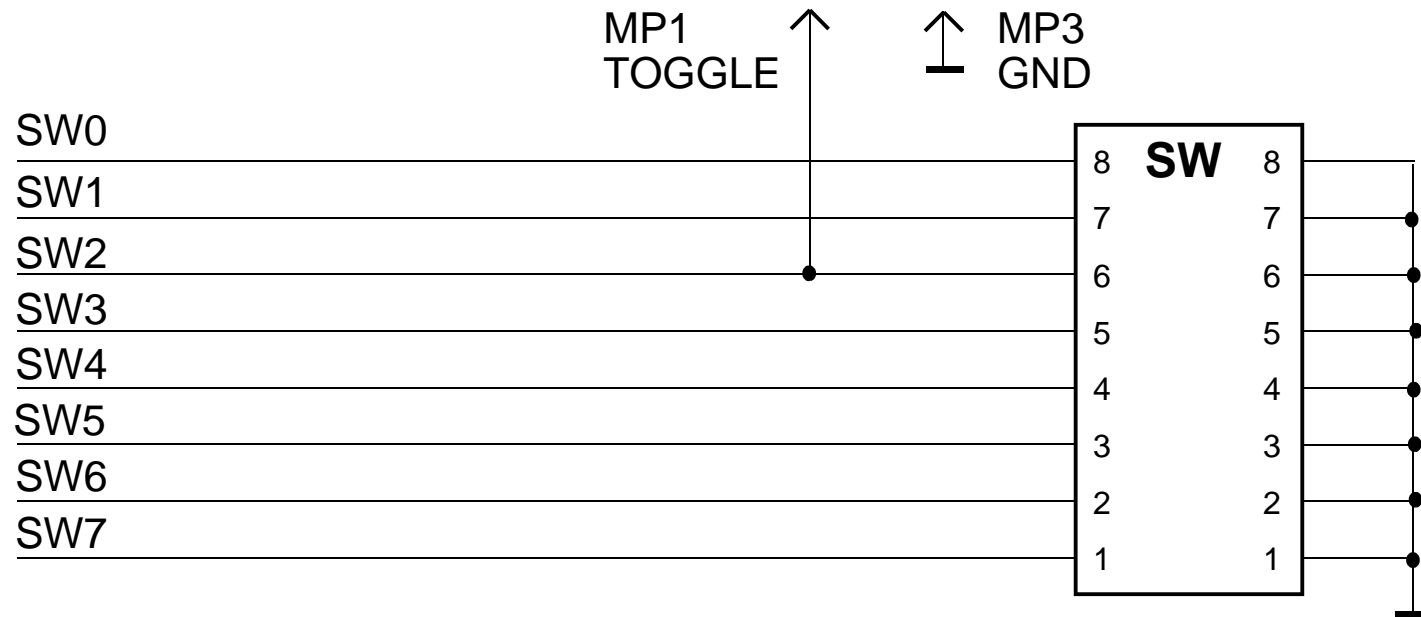


Balkenadapter 13
 Blatt 1 von 4
 10. 6. 13



Balkenadapter 13
 Blatt 2 von 4
 10. 6. 13





- Pull-up-Widerstände im Mikrocontroller aktivieren.
- SW1, 0 bei serieller Kommunikation stets aus.
- SW 2 bei Ansteuerung des Meßpunktes MP1 stets aus.
- Vorzugsweise SW7...3 ausnutzen.

Balkenadapter 13

Blatt 4 von 4

10. 6. 13