



Informatikwerkstatt, Foliensatz 5

Boden- und Abstandssensor

G. Kemnitz

Institut für Informatik, Technische Universität Clausthal
24. November 2014



Inhalt des Foliensatzes

Anschluss der Sensoren

Bodensensor

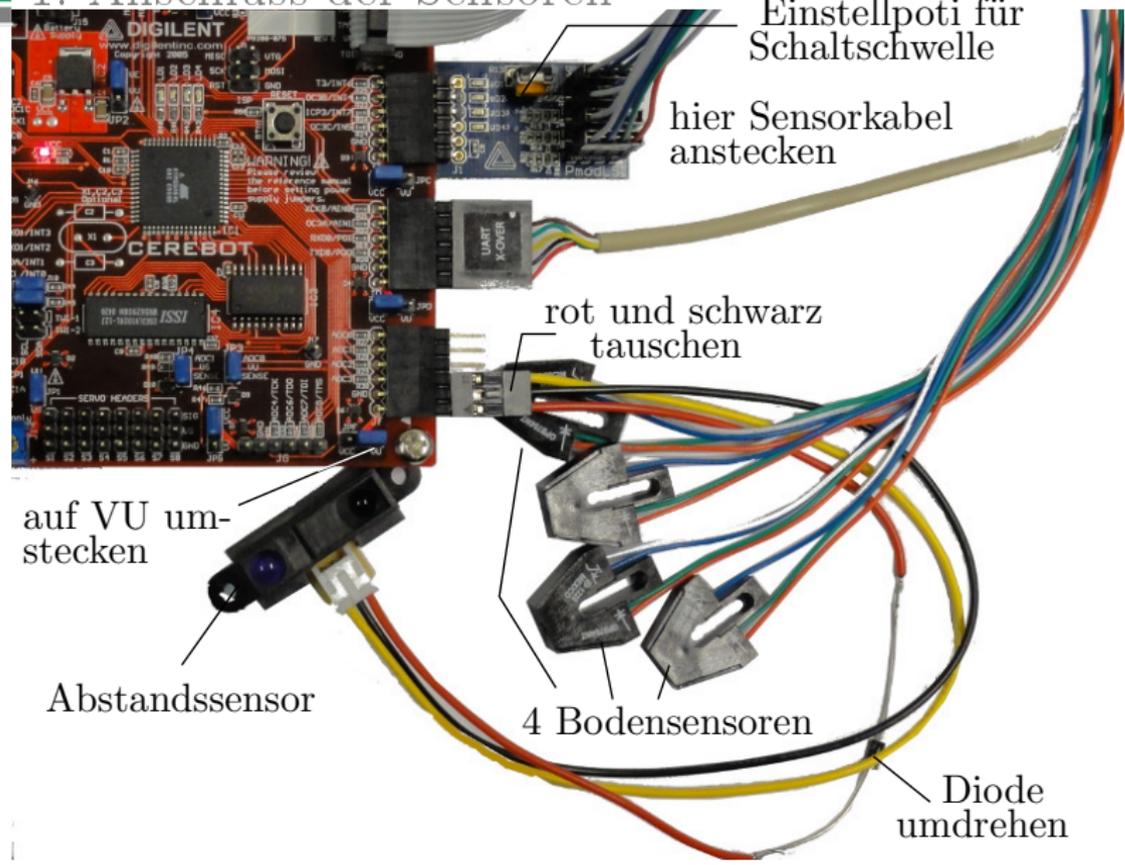
Abstandssensor



Anschluss der Sensoren



1. Anschluss der Sensoren



Einstellpoti für Schaltschwelle

hier Sensorkabel anstecken

rot und schwarz tauschen

auf VU umstecken

Abstandssensor

4 Bodensensoren

Diode umdrehen



*Aufgabe 5.1: Versuchsaufbau vorbereiten

Bodensensoren:

- Stecker der 4 Bodensensoren in das PmodLS1 stecken (weiß zum Stecker). Das PmodLS1 an Stecker JC.
- Spannung anschließen. Schaltschwelle so einstellen, dass die LEDs auf dem Modul bei Annäherung der Reflexionslichtschranken an eine weiße Fläche anschalten.

Abstandssensor:

- Steckerreihenfolge, falls noch nicht erfolgt, von schwarz-rot-gelb in in rot-schwarz-gelb ändern.
- Kontrolle, dass die Diode in der Spannungsversorgungsleitung schon gedreht ist (roter Schrumpfschlauch). Wenn nicht, umlöten.
- Jumper JPF auf VU (5V) umstecken¹.

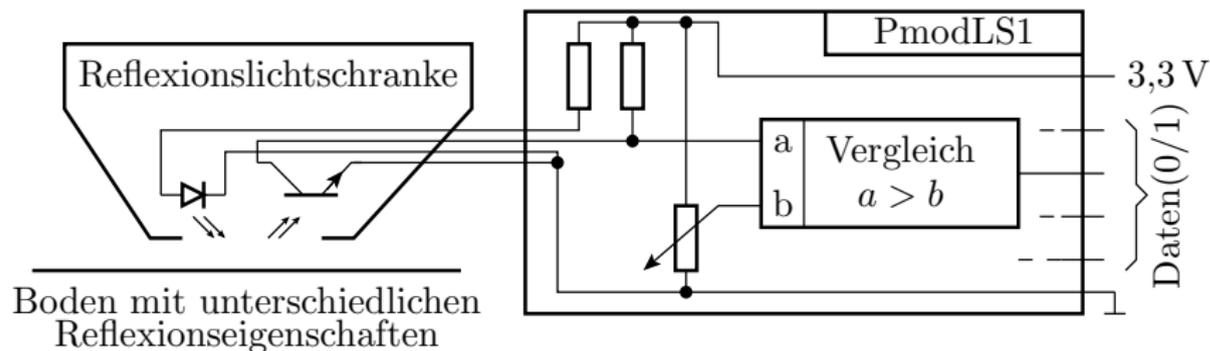
¹Vor Anstecken eines anderen Moduls Jumper auf 3,3V zurückstecken!



Bodensensor

Bodensensor

Der Bodensensor besteht aus vier Reflexionslichtschranken. Jede Reflexionslichtschranke besteht aus einer Infrarot-Leuchtdiode und einem Infrarot-Fototransistor. Je näher oder besser reflektierend der Boden ist, desto stärker ist das Ausgangssignal des Fototransistors. Auf dem PMod erfolgt eine Wandlung in 1 für große und 0 für kleine Werte. Die Schaltschelle wird mit dem Potentiometer eingestellt und die Nullen und Einsen auf LEDs angezeigt.





Testprogramm

```
36 int main(void){
37     initLCD();
38     DDRE &=~0xF0; //PE4..7 Eingänge
39     // LCD löschen + Textausgabe
40     writeString((uint8_t*)"\x1B[jBodensens=");
41     while(1)
42     {
43         //Cursor Zeile 0, Spalte 10
44         writeString((uint8_t*)"\x1B[0;10H");
45         writeBit(PINE&(1<<6)); // Sensor 1
46         writeBit(PINE&(1<<4)); // Sensor 2
47         writeBit(PINE&(1<<7)); // Sensor 3
48         writeBit(PINE&(1<<5)); // Sensor 4
49     }
50 }
```



2. Bodensensor

```
12 void initLCD(){
13     UBRR1H = 0; UBRR1L = 49; // 9600 Baud
14     UCSR1B = (1<<TXEN1); // Empfang ein
15     UCSR1C = 0b00000110; // 8n1
16 }
17
18 void writeChar(uint8_t z){
19     while (!(UCSR1A&(1<<UDRE1))){}
20     UDR1 = z;
21 }
```



2. Bodensensor

```
23 void writeBit(uint8_t b){
24     if (b) writeChar('1');
25     else  writeChar('0');
26 }
27
28 void writeString(uint8_t *ptr){
29     while (!(*ptr==0)){
30         writeChar(*ptr);
31         ptr++;
32     }
33 }
```



*Aufgabe 5.2: Linienverfolgung

- Überlegen Sie sich, wie die Bodensensoren zweckmäßig an das Fahrzeug angebaut werden müssen, um Linien auf dem Boden zu verfolgen.
- Testen Sie für unterschiedliche Linien (auf Papier, Fliesenfugen auf dem Flur, ...) ob der Sensor diese Linien erkennt und verfolgen kann.
- Entwickeln Sie eine Idee für einen Algorithmus für die Linienverfolgung und skizzieren Sie ihn als Programmablauf.
- Programmieren und testen Sie ihren Algorithmus.

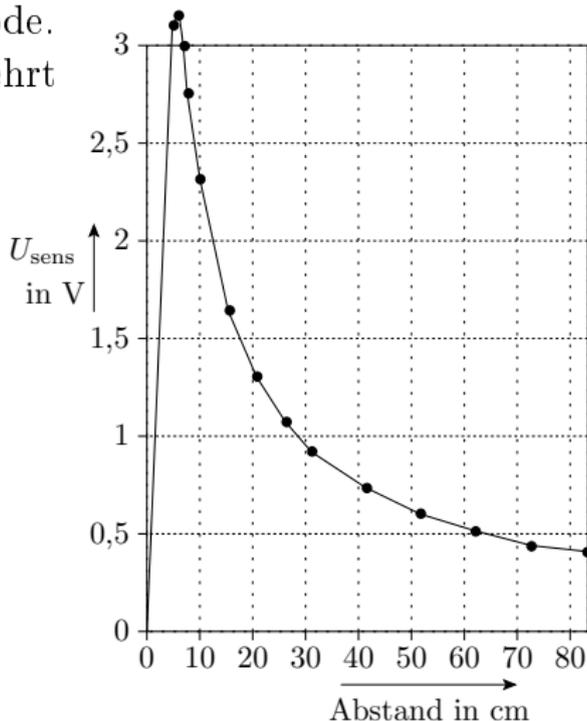
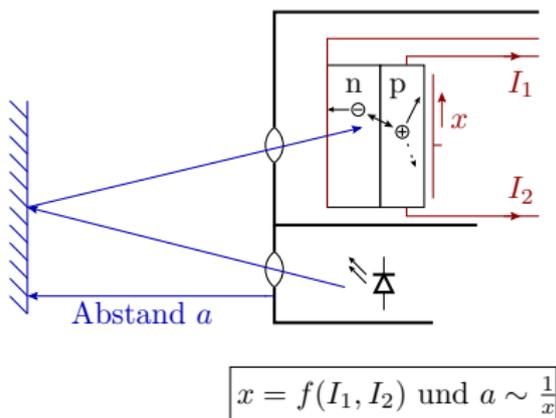


Abstandssensor

Abstandssensor

Der Abstandssensor arbeitet nach dem Triangulationsprinzip mit einer positionsempfindlichen Diode.

- Messspannung etwa umgekehrt proportional zum Abstand.
- Wandlungsgeschwindigkeit ca. 20 Werte pro Sekunde.





Testprogramm

- Das Testprogramm misst die Spannung am Sensorausgang und gibt sie auf dem LC-Display in Volt aus.
- Der ADC (Analog-Digital-Wandler) im Prozessor bestimmt den Messwert m mit 10 Bit Auflösung relativ zu einer Referenzspannung von $3,3V^2$:

$$m = 1024 \cdot \frac{U_{\text{sens}}}{3,3V}$$

- Zur Verbesserung der Genauigkeit werden 322 Wandlerergebnisse addiert. Das ergibt den 10^5 -fachen Spannungswert in Volt mit der $\sqrt{322}$ fachen Varianz einer Einzelmessung, d.h. mit etwa einer Dekade höheren nutzbaren Auflösung.
- LCD-Ausgabe mit einer Vor- und drei Nachkommastellen.

²Auf der Baugruppe ist am Referenzspannungseingang die Versorgungsspannung von $3,3V$ angeschlossen.



3. Abstandssensor

ADC-Initialisierung:

```
11 void initADC(){
12     ADMUX = (3<<MUX0); // Kanal 3 auswählen
13     // Einschalten mit Wandlungstakteiler 64
14     ADCSRA = (1<<ADEN)|(0b110<<ADPS0);
15     DDRE &= ~0x80; // Sensoreingang als Eingang
16     PORTE &=~0x80; // Ausgabewert 0 (hochohmig)
17 }
```

- Der Sensor ist an ADC3 (PF3) (Kanal 3 auswählen).
- Der Wandlertakt als CPU-Takt durch Teilerwert

$$f_{\text{ADC}} = \frac{f_{\text{CPU}}}{64} \approx 117 \text{ kHz}$$

soll im Bereich von 500 bis 200 kHz liegen. Eine ADC-Wandlung dauert 13 Wandlertakte.

- Zur Vermeidung von Spannungsverfälschungen ist PF3 als Eingang mit Ausgabewert 0 (Pull-up aus) einzustellen.



3. Abstandssensor

```
19 uint32_t getADC(){
20     uint16_t i;
21     uint32_t wert=0;
22     // U=3.3V*W/1024, 322 Messungen => Wert in
23     for (i=0;i<322;i++){ // in 10µV-Schritten
24         ADCSRA |= (1<<ADSC); //Wandlung starten
25         // Warten bis "fertig"-Bit gesetzt ist
26         while(!(ADCSRA & (1<<ADIF))){}
27         // "fertig"-Bit durch Schreiben einer
28         ADCSRA |= (1<<ADIF); // 1 löschen
29         wert += ADC;
30     }
31     return wert;
32 }
```

- Wandlungsstart durch Setzen von ADSC in ADCSRA.
- Bei Wandlungsabschluss setzt der Prozessor ADIF=1.
- ADIF wird durch Schreiben einer Eins gelöscht.



3. Abstandssensor

```
56 int main(void) {
57     uint32_t w, b;
58     initADC();
59     initLCD();
60     writeString((uint8_t*)"x1B[jU_Abst=");
61     while(1) {
62         //Cursor Zeile 0, Spalte 8
63         writeString((uint8_t*)"x1B[0;8H");
64         w = (uint32_t)getADC();
65         for (b=100000;b>10;b/=10) {
66             writeChar(w/b+'0');
67             w -= b*(w/b);
68             if (b==100000) writeChar(',');
69         }
70         writeChar('V');
71     }
72 }
```



*Aufgabe 5.3: Umrechnung der Messspannung in cm

- Bestimmen Sie für ihren Sensor den Zusammenhang zwischen Hindernisabstand a und der Sensorspannung U_{sens} .
- Untersuchen Sie für unterschiedliche Arten von Hindernissen, wie genau die Messungen reproduzierbar sind und entscheiden Sie, wie viele Nachkommastellen des Abstandswertes Sie in welchem Entfernungsbereich nutzen wollen.
- Testen Sie, ob der Zusammenhang zwischen dem Abstand a und der Spannung U_{sens} durch eine Funktion

$$a = \frac{c_1}{U_{\text{sens}} - c_2}$$

(c_1, c_2 – empirisch zu bestimmende Konstanten) angenähert werden kann. Wenn nicht, suchen Sie nach einer anderen brauchbaren Näherung, z.B. stückenweise durch Geraden.

- Schreiben Sie ein Testprogramm, das auf dem LC-Display den Abstand in cm mit einer sinnvollen Stellenanzahl zeigt.