

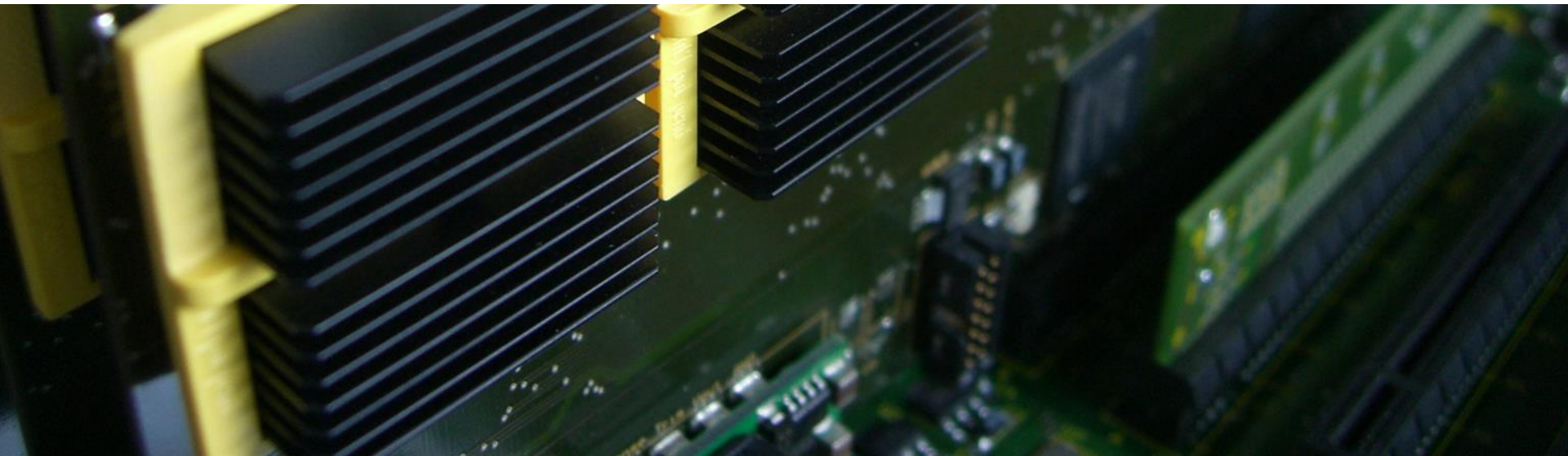
Eingebettete Prozessoren

SS 2014

Übung 12: ADC

Dipl.-Ing. Thomas Pöppelmann
Arbeitsgruppe Sichere Hardware
Horst Görtz Institut für IT-Sicherheit

17.07.2014

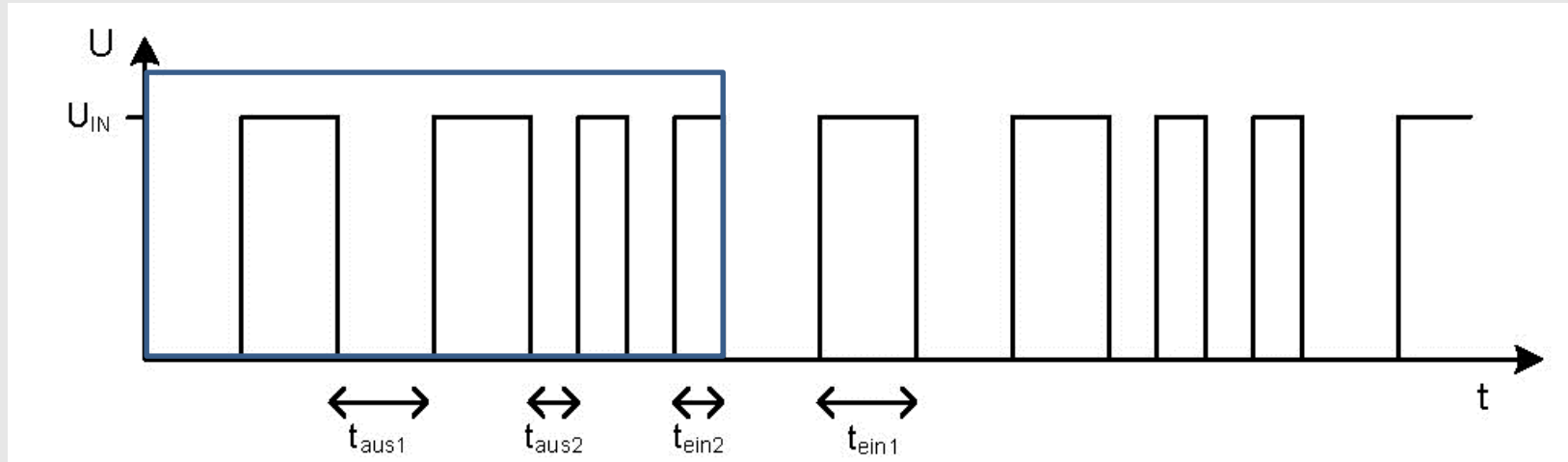


Agenda

1. **Besprechung Übung 11**
2. **Sicherungssysteme**
3. **ADC**

1. Besprechung Übung 11

Aufgabe 11)1): Duty Cycle



Nehmen Sie an, dass eine PWM-Einheit so konfiguriert wurde, dass über eine regelmäßige, Interrupt basierte Aktualisierung des OCR2-Referenzregisters das folgende periodische Signal am Pin OC2 generiert wird:

- Berechnen Sie den duty cycle und die Effektivspannung U_{eff} unter der Annahme, dass $t_{ein1} = t_{aus1} = 28 \mu s$ und $t_{ein2} = t_{aus2} = 14 \mu s$, sowie $U_{IN} = 5 V$ gegeben seien.

- Duty Cycle

- Duty Cycle: $d = \frac{t_{ein}}{t_{ein} + t_{aus}}$

- $d = (t_{ein2} + t_{ein1}) / (t_{ein1} + t_{aus1} + t_{ein2} + t_{aus2}) = (14 + 28) / (28 + 28 + 14 + 14) = 1/2$

- Eff. Wert:

- $U_{eff} = U_{IN} \frac{t_{ein}}{t_{ein} + t_{aus}}$

- Die effektive Spannung U_{eff} ist damit $U_{eff} = 2.5 \text{ V}$ ($5\text{V} \cdot 1/2$)

Aufgabe 11)2/3): Sleep Mode

Table 14. Active Clock Domains and Wake-up Sources in the Different Sleep Modes

Sleep Mode	Active Clock Domains					Oscillators		Wake-up Sources					
	clk _{CPU}	clk _{FLASH}	clk _{IO}	clk _{ADC}	clk _{ASY}	Main Clock Source Enabled	Timer Osc. Enabled	INT1 INT0	TWI Address Match	Timer 2	SPM/EEPROM Ready	ADC	Other I/O
Idle			X	X	X	X	X ⁽²⁾	X	X	X	X	X	X
ADC Noise Reduction				X	X	X	X ⁽²⁾	X ⁽³⁾	X	X	X	X	
Power Down								X ⁽³⁾	X				
Power Save					X ⁽²⁾		X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾			
Standby ⁽¹⁾						X		X ⁽³⁾	X				

Notes: 1. External Crystal or resonator selected as clock source
 2. If AS2 bit in ASSR is set
 3. Only level interrupt INT1 and INT0

- 2) Nehmen Sie an, der ATmega8 befindet sich im "Power Down" Sleep-Mode. Ist ein ADC Event in der Lage den Controller aufzuwecken?
 – Antwort: Nein
- 3) In welchem Sleep Mode kann der Controller durch ein "Other I/O" Event aufgeweckt werden?
 – Antwort: IDLE
- Hinweis: Ziel der Aufgabe - Datenblatt lesen

Aufgabe 11)4): PWM

Entwickeln Sie Makros um den Asuro mit einer beliebigen Geschwindigkeit z.B. Vor- und Rückwärts fahren zu lassen.

- Siehe Beispielprogramm im Blackboard

2. Sicherungssysteme

Sicherungssysteme

- Aktives System Monitoring umfasst
 - Überwachung der Temperatur des Prozessors oder des Gesamtsystems
 - Z.B. mittels eines Temperatursensors/ADC
 - Überwachung der Energieversorgung (Brown-Out Detection)
 - Nutzen eines ADCs bzw. Komparators, der die Eingangsspannung mit einer nicht zu unterschreitenden Referenzspannung vergleicht
 - Wichtig bei batteriebetriebenen oder kritischen Systemen
 - Manipulationsschutz
 - Erkennen von Manipulationen am System z.B. durch Sicherungsschalter
 - Schutz gegen Systemabstürze
 - Watch-Dog Timer

Watchdog Timer - Sicherungssysteme

- Watch-Dog Timer (WDT): Totmannschaltung, die Softwareabstürze und Deadlocks eines Programms in Hardware überwachen kann
- Programm muss die ordnungsgemäße Funktionsweise in regelmäßigen Abständen dem Prozessor bestätigen
- Realisiert als unabhängiger, zusätzlicher 10 Bit Zeitgeber
 - Einstellbarer Vorteiler
 - Bei Überlauf erfolgt automatischer Reset des Prozessors
- WDT muss regelmäßig zurückgesetzt werden
 - Kann zu Problemen führen, wenn der WDT nicht korrekt zurückgesetzt wird

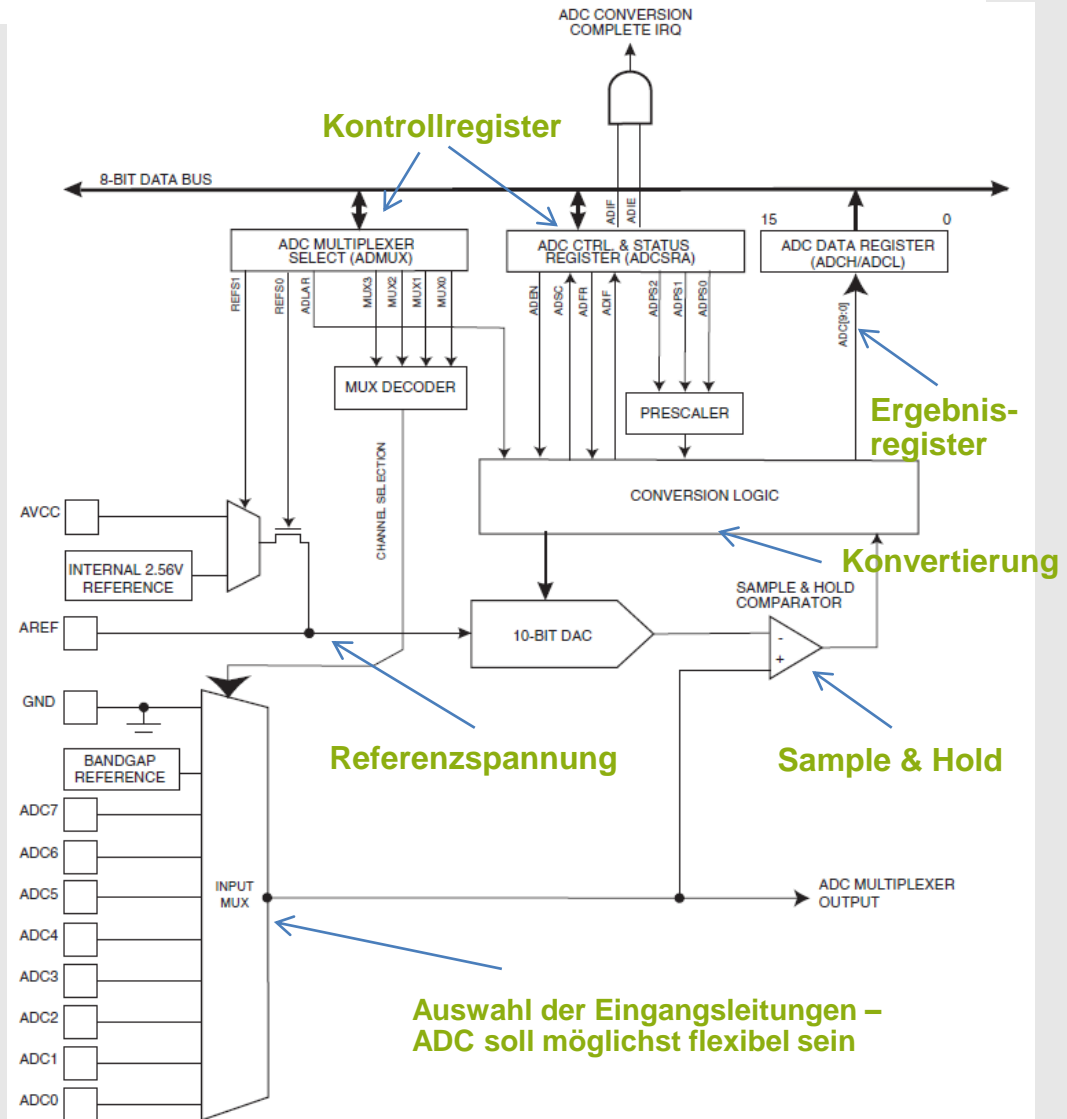
3. Konfiguration des ADC

ADC Wiederholung

- Der ADC quantisiert ein analoges Eingangssignal und gibt den Messwert als digitalen n-Bitvektor zurück.
 - Eingangsspannung wird in digitalen Wert umgewandelt
 - Einordnung der Messspannung mit zwischen GND und VCC
- Abtasttheorem muss beachtet werden
 - $f_A \geq 2(f_{\max} - f_{\min})$
 - Unterabtastung ist zu vermeiden (Fehler)
 - Überabtastung ebenfalls (Ressourcenverbrauch)
- Sample & Hold System verhindert Abweichungen durch Signaländerung während der Laufzeit des ADC
- Im AVR ist der ADC eine separate Hardwareeinheit die unabhängig von der CPU agieren kann

Der ADC im ATmega8

- 10-Bit Auflösung (Wertebereich 0 bis 1023)
 - Minimum: GND
 - Maximum: ~AREF (Referenzspannung)
 - $ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$
- Interne/Externe Referenzspannung
- Einmaliges Auslösen einer Messung oder wiederholte (frei laufende) Messung
- Signalisierung über Flags und Interrupts
- 6-8 Eingangskanäle (Hardware Multiplexer)
- Rauschen ist normal – besonders in den niederwertigen Bits
 - Abfrage $MESSUNG == x$ schlägt meistens fehl
 - Besser einen Bereich $y < MESSUNG \leq x$ testen
- Messung dauert (successive approximation)
 - Nach Initialisierung: 25 Zyklen (ADC)
 - Normal: 13 Zyklen (ADC)



ADC-Betriebsmodi

Zusammenfassung der Betriebsmodi des ADC:

- *Single conversion*: Eine Messung wird gestartet. *ADSC* setzen.
- *Free running*: Kontinuierliche Messung. *ADFR* setzen.

Interaktion/Signalisierung der CPU:

- Bit-clear: Ein Bit (*ADSC*) wird von der ADC-Hardware gelöscht, sobald eine Messung abgeschlossen ist
- Interrupt: Sobald eine Messung abgeschlossen ist, wird ein Interrupt ausgelöst
- Benutzung:
 - Im *Single conversion* Modus wird meistens über ein Flag signalisiert (*ADIF* is set/*ADSC* cleared)
 - Im *Free running* Modus wird meistens mit Interrupts gearbeitet (*ADIE* Bit und I-Bit in *SREG* gesetzt)
- Hinweis:
 - *ADIF*: ADC Interrupt Flag vs. *ADIE*: ADC Interrupt Enable

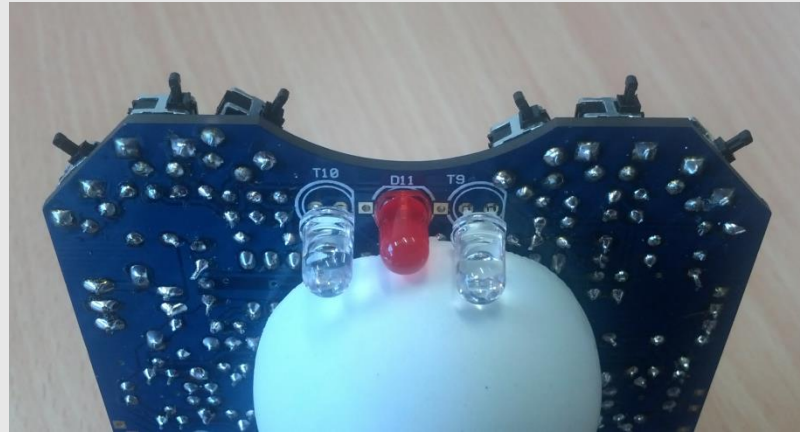
ADC Benutzen

- Auslösen einer Messung (Polling/ohne Interrupts)
 - Konfiguration im Vorfeld
 - ADC aktivieren: *ADEN* in *ADCSRA* setzen
 - Prescaler (ADC) auswählen (z.B. CLK/64): *ADPS2:0* in *ADCSRA* entsprechend setzen
 - Referenzspannung auswählen (z.B. AV_{CC} mit ext. Kapazität): *REFS1:0* in *ADMUX* setzen (Tabelle 74 im Datenblatt)
 - Messen eines Eingangspins
 - Eingangspin auswählen: *MUX3:0* in *ADMUX* setzen (Tabelle 75)
 - ADC starten: *ADCSC* in *ADCSRA* setzen
 - Warten bis Konvertierung beendet ist: *ADCSC* wird wieder von der Hardware nach Beendigung der Messung automatisch auf 0 gesetzt.
 - Auslesen des Ergebnisses (Reihenfolge beachten): erst *ADCL* und dann *ADCH*
 - Kann den ADC blockieren – keine neuen Messwerte
 - Wenn das Ergebnis linksbündig in *ADCH:ADCL* geschrieben wird muss nur *ADCH* ausgelesen werden

Übung

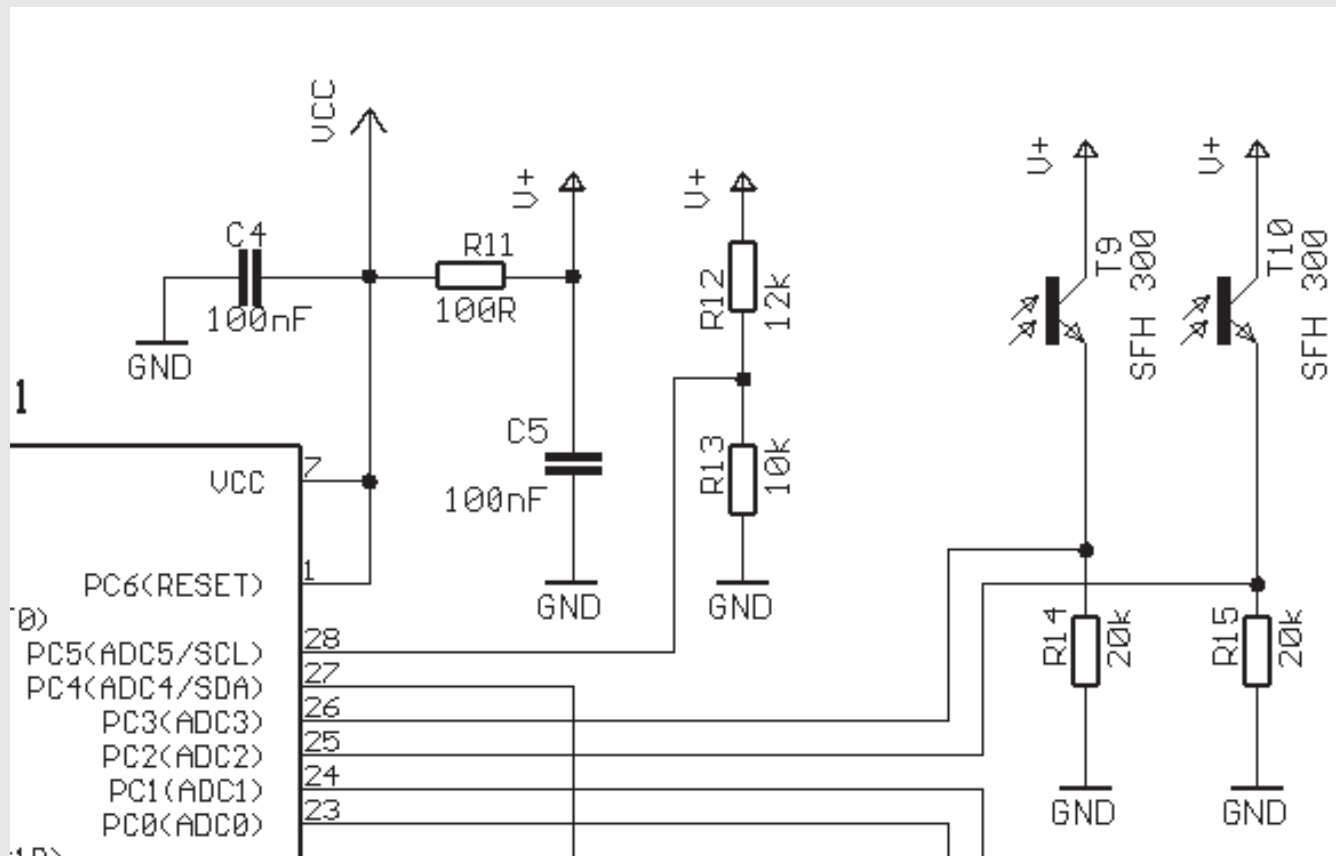
- 2 Gruppenaufgaben (Fototransistor + Kollisionstaster)
 - Zusammen 100%
- 2 Einzelaufgaben (Sonderpunkte)
 - Klausurrelevant
 - Gruppenaufgaben umfangreich, aus diesem Grund gibt es Sonderpunkte

Anwendung: Fototransistoren (Gruppenaufgabe 1)



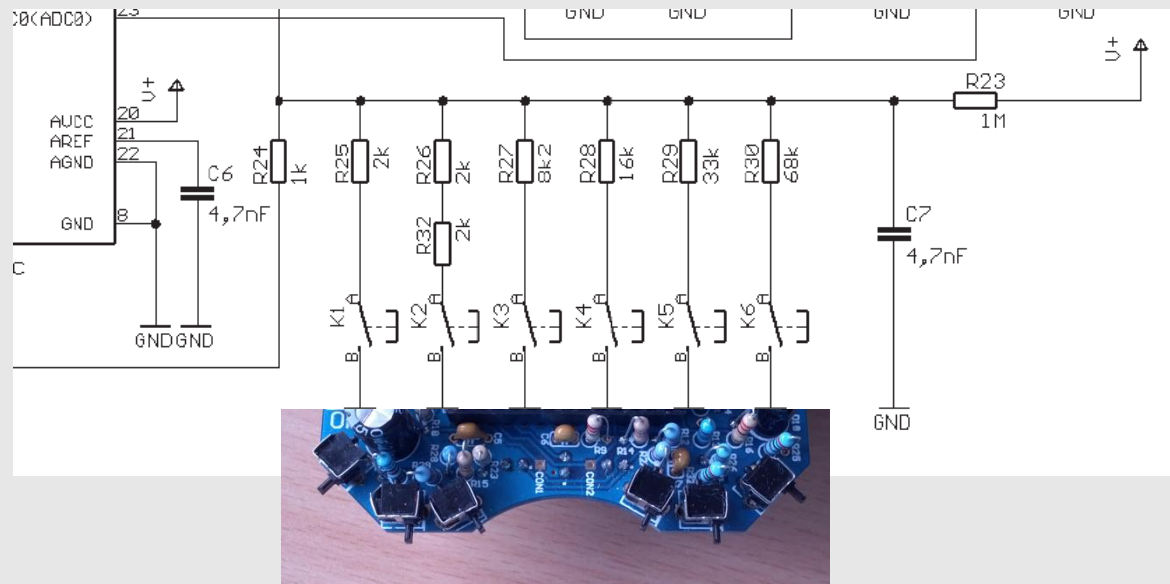
- Leuchtdiode erzeugt Licht welches von der Strecke reflektiert und von den Phototransistoren gemessen wird
- Kann zur Linienverfolgung benutzt werden
- Prinzip: Differenz aus linker und rechter Diode bilden: $\text{Diff} = \text{ADC}(\text{Links}) - \text{ADC}(\text{Rechts})$
 - Differenz unter bestimmten Scheitelwert \Rightarrow Gleich hell
 - Differenz größer als Scheitelwerts \Rightarrow Links heller
 - Differenz kleiner als negativer Scheitelwerts \Rightarrow Rechts heller
- Messungen können ungenau sein. Sehr stark abhängig vom Licht und dem Lichteinfallwinkel.

Anwendung: Fototransistoren (Gruppenaufgabe 1)



- T9 ist an *ADC3* angeschlossen
- T10 ist an *ADC2* angeschlossen

Anwendung: Kollisionstaster (Gruppenaufgabe 2)



- Bisher konnten wir nur ermitteln, ob ein Taster gedrückt wurde
- Per ADC Messung wird die am Pin anliegende Spannung gemessen
- Spannung ist abhängig vom zum Taster gehörenden Widerstand (Werte in Übung gegeben)
- RC Glied zur Glättung (siehe Übung)

Anwendung: Kollisionstaster

(Gruppenaufgabe 2)

- Falls das Messen der Kollisionstaster nicht funktioniert bitte einmal die angegebenen Werte ausprobieren:

CPI TMPH, 180

BRLO KEY0

CPI TMPH, 210

BRLO KEY1

CPI TMPH, 235

BRLO KEY2

CPI TMPH, 244

BRLO KEY3

CPI TMPH, 250

BRLO KEY4

CPI TMPH, 254

BRLO KEY5

RJMP NOKEY

Weitere Informationen

- Weitere Informationen zum ADC
 - Skript
 - Datenblatt
 - ADC Tutorial:
[http://www.mikrocontroller.net/articles/AVR-Tutorial: ADC](http://www.mikrocontroller.net/articles/AVR-Tutorial:ADC)
- Nach Abgabefrist der Übung 12 (24.07.2014, 23:59 Uhr) wird Lösungshinweis zur Klausurvorbereitung im Blackboard veröffentlicht
- Informationen zu Bonuspunktberechnung und Ausro-Führerschein in der Vorlesung oder im Blackboard

Motivation

- Eingebettete Prozessoren ist nur eine einsemestrige Vorlesung
- Einige Themen haben wir nicht oder nur sehr kurz behandelt
- Grundlagenkenntnisse aus der Vorlesung sollten ausreichen um sich selbständig in die Thematik weiter einzuarbeiten (bei Interesse)
- Folien dazu im Blackboard
 - Arduino
 - Ausro verbessern
 - Lehrveranstaltungen mit Empro Bezug
 - Hiwi mit Empro Bezug
 - Abschlussarbeiten

Dankeschön

Vielen Dank für die Aufmerksamkeit!

Viel Erfolg in der Klausur!