

Der Analog-Digital-Wandler des ATmega32

Der Analog-Digital-Wandler des Atmega32 arbeitet nach dem Verfahren der sukzessiven Approximation mit einer Auflösung von 10 Bit. Über einen Multiplexer können 8 Spannungsquellen angeschlossen werden (Sonderfunktion von PortA).

Die Register:

Das ADC Kontroll- und Status-Register ADCSRA

Mit diesem Register wird die Taktfrequenz für die Analog-Digital-Wandlung eingestellt, der Wandler aktiviert und die Wandlung gestartet. Die nicht beschriebenen Bits betreffen Trigger- und Interrupt-Betrieb werden hier nicht behandelt(Siehe hierzu das Datenblatt).

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit 7 ADEN: ADC Enable Mit einer 1 wird der Analog-Digital-Wandler aktiviert.

Bit 6 ADSC: Start Conversion Jede Einzelwandlung muss mit ADSC=1 gestartet werden. Ist die Wandlung abgeschlossen, stellt der Analog-Digital-Wandler ADSC auf 0 zurück.

Bit 2,1 und 0

Diese Bits legen die Taktfrequenz fest, mit der die Wandlung erfolgt. Dabei wird die Controller-Taktfrequenz (hier 8 Mhz) durch den angegebenen Teilungsfaktor geteilt. Die Wandler-Taktfrequenz sollte zwischen 50kHz und 200kHz liegen.

Bei einer Controller-Taktfrequenz von 8 MHz wählt man einen Teilungsfaktor von 64. Die resultierende Wandler-Taktfrequenz ist dann $f=8\text{MHz}/64 = 125\text{kHz}$.

Table 85. ADC Prescaler Selections

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Das ADC Multiplexer Selection Register - ADMUX

Dem Digital-Analog-Wandler ist ein Multiplexer vorgeschaltet, der einen der 8 Eingangskanäle (PortA PA0-PA7) zuschaltet. Dieser Kanal wird mit dem ADMUX-Register(MUX4..0) ausgewählt.

Darüber hinaus stellt man mit ADMUX die Referenzspannung ein. Diese bestimmt den Messbereich. Bei REFS1=0 und REFS0=0 wird die externe 5V Spannung als Referenzspannung benutzt.

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Table 83. Voltage Reference Selections for ADC

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

MUX4..0	Single Ended Input
00000	ADC0
00001	ADC1
00010	ADC2
00011	ADC3
00100	ADC4
00101	ADC5
00110	ADC6
00111	ADC7

Das ADC-Datenregister

Das ADC-Datenregister enthält nach einer abgeschlossenen Wandlung den digitalen Messwert mit einer Auflösung von 10 Bit.

Dieses 16-Bit-Register kann in C mit **ADCW** angesprochen werden.

15	14	13	12	11	10	9	8	
-	-	-	-	-	-	ADC9	ADC8	ADCH
7	6	5	4	3	2	1	0	ADCL
ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	

C-Funktionen zur Spannungsmessung:

```
void ADC_init(void){
//Diese Funktion aktiviert den ADC

sbi(ADCSRA,ADEN);//ADC aktivieren und
sbi(ADCSRA,ADPS2);//Teilungsfaktor auf 64 stellen:
sbi(ADCSRA,ADPS1);
}

uint16_t ADC_read(uint8_t kanal){
//Diese Funktion misst eine Spannung an einem ADC-Kanal
//und gibt den entsprechenden digitalen Wert 0 - 1023 (10 Bit)
//zurück.
//-----
ADMUX=kanal; //Kanal des Multiplexers wählen:
sbi(ADCSRA,ADSC);//Messung starten
while(bit_is_set(ADCSRA,ADSC));// Auf Ergebnis warten:
return ADCW;
}
```

Die Spannung u_0 an Kanal 0 (ADC0) kann dann (bei AREF=5V) berechnet werden mit:

```
u0 = 5.0*ADC_read(0)/1023;// mit double u0
```