Name: Daniel Khüny Punkte: 22/23P

Datum: 24.04.2013 Note: 1

1) Timer

Nutze den beigefügten Datenblattauszug.

```
TCCR0=0x0B // 0000 1011 → CTC-Mode  Prescaler = 64
TCNT0=0x10 // Startwert = 16
OCR0=0x7F  //127
TIMSK=0x02 // OCIE = 1
```

Taktfrequenz des Mikrocontrollers ist 8Mhz.

1.a) Registerbelegung – Multiple Choice 2,5/3P

	Richtig	Falsch	
Timer 0 ist ein 16 Bit Timer	☐	☒	✓
Der Timer besitzt intern einen Zähler	☒	☐	✓
In einem Durchlauf kann dieser Timer keine Zeiten >35ms realisieren	☐	☒	
Im CTC Mode zählt der Zähler immer bis 0xFF	☐	☒	✓
Über TCCR0 lässt sich der Timer stoppen	☒	☐	✓
Der Prozessortakt ist von OCR0 abhängig	☐	☒	✓

1.b) Berechnung der Zeit 3/3P

Nach welcher Zeit wird der Output Compare Match Interrupt ausgeführt? Erläutere die einzelnen Rechenschritte.

$f_T = \frac{f_{CLK}}{Presc \cdot (1+OCR0)} = \frac{8\,MHz}{64 \cdot (1+127)} = 976{,}56\,Hz$

$t_T = \frac{1}{f_T} = 1{,}024\,ms$

$t_{OCIE} = (127 - 16) \cdot t_T = 113{,}66\,ms$ ✓

1.c) Timeranpassung 4/4P

Was muss geändert werden, um möglichst nah an 2ms zu kommen? Erläutere die Rechenschritte und berechne die resultierende Abweichung.

$t_T = 2ms = f_T = \frac{1}{2ms} = 500\,Hz$

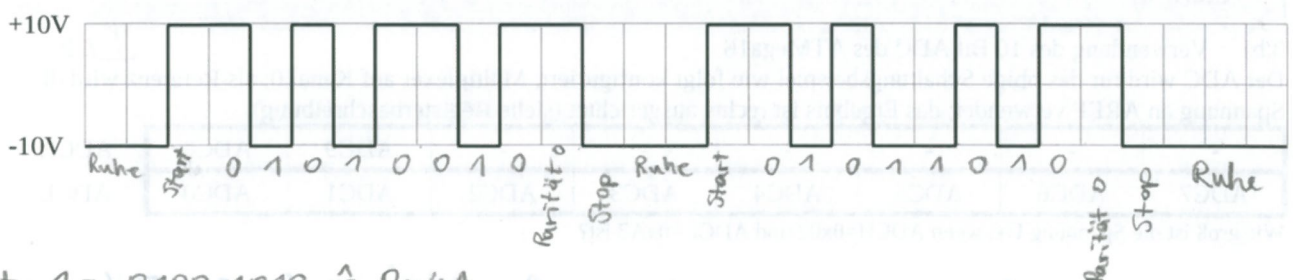
$OCR0 = \frac{f_{CLK}}{Presc \cdot f_T} - 1 = \frac{8\,MHz}{64 \cdot 500Hz} - 1 = 249 \triangleq 0xF9$ ✓

Abweichung = 0 % (bei idealen Bauteilen) ✓

2) Serielle Datenübertragung

2.a) Signalübertagung 5/5P

Über die RS232 Schnittstelle eines PCs werden Daten übertragen. Die Übertragungsparameter sind 8O1. Welche Daten wurden übertragen? Markiere Start, Stop und Paritätsbit. Kam es zu einem Übertragungsfehler?



Byte 1 = 0100 1010 ≙ 0x4A ✓

Byte 2 = 0101 1010 ≙ 0x5A ✓ jedoch ist es hier zu einem Übertragungsfehler gekommen da das Paritätsbit fälschlicherweise 0 ist

2.b) Einstellen des USART des ATMega16 3/3P

Die Taktfrequenz des Zielsystems ist 10 Mhz. Die gewünschte Baudrate ist 115200 Baud. Verwende den "Normal Mode" zur Generierung der Baudrate.

- Welcher Wert wird im USART Baud Rate Register (UBRR) eingestellt?
- Wie groß ist die Abweichung von der gewünschten Baudrate (in Prozent)?
- Wird die Übertragung mit dieser Abweichung funktionieren? Wo liegt die Grenze?
- Durch welche Maßnahme lässt sich die Abweichung minimieren?

Normal Mode ⇒ presc = 16 ✓

$\text{Baudrate} = \frac{f_{CLK}}{16\,(1+UBRR)}$ $UBRR = \frac{f_{CLK}}{16\,Baud} - 1 = \frac{10MHz}{16 \cdot 115200} - 1 = 4{,}43 \approx 4$

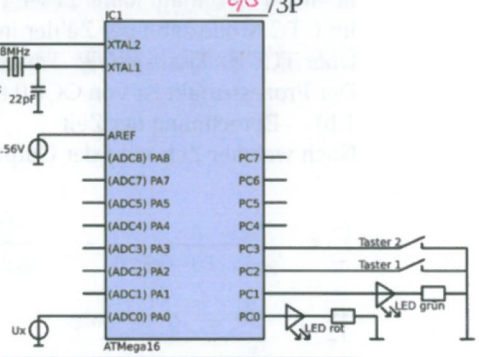
$\text{Baud} = \frac{10\,MHz}{16\,(1+4)} = 125\,000$ $\frac{125\,000}{115\,200} \cdot 100 = 108{,}5\,\%$ ⇒ F = + 8,5 %

Die Übertragung wird mit diesem Fehler nicht verlässlich funktionieren, da die Grenze für die Abweichung bei ± 2 % liegt. ✓

3) Anwendungsbeispiel

3.a) Multiple Choice 2,5/3P

	Richtig	Falsch
Beim Start leuchtet die rote LED	☒	☐
Wenn Taster1 gedrückt gehalten wird, ändert sich counter ständig	☐	☒ ✓
Mit den Tastern kann man hinauf und hinunter zählen	☒	☐ ✓
Die rote LED ist abhängig von counter	☒	☐ ✓
Durch häufiges Drücken von Taster1 kann man die grüne LED einschalten	☒	☐ ✓
counter kann nie größer als 100 werden	☐	☒ ✓




```
#include <avr/io.h>

int main(void) {
  unsigned char old_state=0, new_state;
  unsigned char counter=0;

  DDRC=0x03;
  PORTC=0xFE;

  while(1) {
    new_state=PINC;

    if ( (old_state&0x04) && ( (new_state&0x04)==0) ) {
      counter++;
    } else if ( (old_state&0x08) && ( (new_state&0x08)==0) ) {
      if (counter>0) counter--;
    }

    if (counter<5) {
      PORTC=(PORTC&0xFC)|0x02;
    } else if (counter>100) {
      PORTC=(PORTC&0xFC)|0x01;
    }

    old_state=new_state;
  }

  return 0;
}
```

3.b) Verwendung des 10 Bit ADC des ATMega16 2/2P

Der ADC wird für das obige Schaltungsbeispiel wie folgt konfiguriert: Multiplexer auf Kanal 0; als Referenz wird die Spannung an AREF verwendet; das Ergebnis ist rechts ausgerichtet (siehe Registerbeschreibung).

-	-	-	-	-	-	ADC9	ADC8	ADCH
ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL

Wie groß ist die Spannung Ux, wenn ADCH=0x02 und ADCL=0xA3 ist?

ADC = 0x 2A3 ≙ 675 10 Bit ADC $2^n = 1024$ $V_{Ref} = 2{,}56\,V$

$V_x = \frac{675}{1024} \cdot 2{,}56\,V = 1{,}69\,V$ ✓

zu 2.) b)

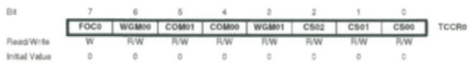
Die Abweichung kann auf folgende Arten minimiert werden:

- Änderung des Prescalers ✓
- Änderung von f_{CLK} ✓
- Änderung der gewünschten Baudrate ✓

Daniel Khiing

8-bit Timer/Counter Register Description

Timer/Counter Control Register – TCCR0



- **Bit 7 – FOC0: Force Output Compare**

The FOC0 bit is only active when the WGM00 bit specifies a non-PWM mode. However, for ensuring compatibility with future devices, this bit must be set to zero when TCCR0 is written when operating in PWM mode. When writing a logical one to the FOC0 bit, an immediate compare match is forced on the Waveform Generation unit. The OC0 output is changed according to its COM01:0 bits setting. Note that the FOC0 bit is implemented as a strobe. Therefore it is the value present in the COM01:0 bits that determines the effect of the forced compare.

A FOC0 strobe will not generate any interrupt, nor will it clear the timer in CTC mode using OCR0 as TOP.

The FOC0 bit is always read as zero.

- **Bit 3, 6 – WGM01:0: Waveform Generation Mode**

These bits control the counting sequence of the counter, the source for the maximum (TOP) counter value, and what type of Waveform Generation to be used. Modes of operation supported by the Timer/Counter unit are: Normal mode, Clear Timer on Compare Match (CTC) mode, and two types of Pulse Width Modulation (PWM) modes. See Table 38 and "Modes of Operation" on page 76.

Table 38. Waveform Generation Mode Bit Description[1]

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0	TOV0 Flag Set-on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

Note: 1. The CTC0 and PWM0 bit definition names are now obsolete. Use the WGM01:0 definitions. However, the functionality and location of these bits are compatible with previous versions of the timer.

- **Bit 5:4 – COM01:0: Compare Match Output Mode**

These bits control the Output Compare pin (OC0) behavior. If one or both of the COM01:0 bits are set, the OC0 output overrides the normal port functionality of the I/O pin it is connected to. However, note that the Data Direction Register (DDR) bit corresponding to the OC0 pin must be set in order to enable the output driver.

When OC0 is connected to the pin, the function of the COM01:0 bits depends on the WGM01:0 bit setting. Table 39 shows the COM01:0 bit functionality when the WGM01:0 bits are set to a normal or CTC mode (non-PWM).

Table 39. Compare Output Mode, non-PWM Mode

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

Table 40 shows the COM01:0 bit functionality when the WGM01:0 bits are set to fast PWM mode.

Table 40. Compare Output Mode, Fast PWM Mode[1]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match, set OC0 at BOTTOM, (non-inverting mode)
1	1	Set OC0 on compare match, clear OC0 at BOTTOM, (inverting mode)

Note: 1. A special case occurs when OCR0 equals TOP and COM01 is set. In this case, the compare match is ignored, but the set or clear is done at TOP. See "Fast PWM Mode" on page 77 for more details.

Table 41 shows the COM01:0 bit functionality when the WGM01:0 bits are set to phase correct PWM mode.

Table 41. Compare Output Mode, Phase Correct PWM Mode[1]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match when up-counting. Set OC0 on compare match when downcounting.
1	1	Set OC0 on compare match when up-counting. Clear OC0 on compare match when downcounting.

Note: 1. A special case occurs when OCR0 equals TOP and COM01 is set. In this case, the compare match is ignored, but the set or clear is done at TOP. See "Phase Correct PWM Mode" on page 79 for more details.

- **Bit 2:0 – CS02:0: Clock Select**

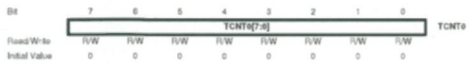
The three Clock Select bits select the clock source to be used by the Timer/Counter.

Table 42. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$clk_{I/O}$/(No prescaling)
0	1	0	$clk_{I/O}$/8 (From prescaler)
0	1	1	$clk_{I/O}$/64 (From prescaler)
1	0	0	$clk_{I/O}$/256 (From prescaler)
1	0	1	$clk_{I/O}$/1024 (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

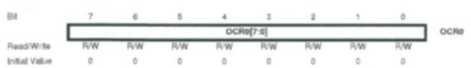
If external pin modes are used for the Timer/Counter0, transitions on the T0 pin will clock the counter even if the pin is configured as an output. This feature allows software control of the counting.

Timer/Counter Register – TCNT0



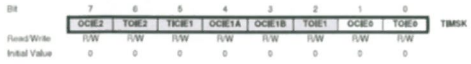
The Timer/Counter Register gives direct access, both for read and write operations, to the Timer/Counter unit 8-bit counter. Writing to the TCNT0 Register blocks (removes) the compare match on the following timer clock. Modifying the counter (TCNT0) while the counter is running, introduces a risk of missing a compare match between TCNT0 and the OCR0 Register.

Output Compare Register – OCR0



The Output Compare Register contains an 8-bit value that is continuously compared with the counter value (TCNT0). A match can be used to generate an output compare interrupt, or to generate a waveform output on the OC0 pin.

Timer/Counter Interrupt Mask Register – TIMSK



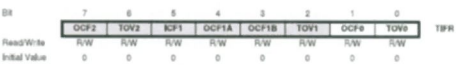
- **Bit 1 – OCIE0: Timer/Counter0 Output Compare Match Interrupt Enable**

When the OCIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Compare Match interrupt is enabled. The corresponding interrupt is executed if a compare match in Timer/Counter0 occurs, that is, when the OCF0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

- **Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

When the TOIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter0 occurs, that is, when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

Timer/Counter Interrupt Flag Register – TIFR



- **Bit 1 – OCF0: Output Compare Flag 0**

The OCF0 bit is set (one) when a compare match occurs between the Timer/Counter0 and the data in OCR0 – Output Compare Register0. OCF0 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, OCF0 is cleared by writing a logic one to the flag. When the I-bit in SREG, OCIE0 (Timer/Counter0 Compare Match Interrupt Enable), and OCF0 are set (one), the Timer/Counter0 Compare Match Interrupt is executed.

- **Bit 0 – TOV0: Timer/Counter0 Overflow Flag**

The bit TOV0 is set (one) when an overflow occurs in Timer/Counter0. TOV0 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV0 is cleared by writing a logic one to the flag. When the SREG I-bit, TOIE0 (Timer/Counter0 Overflow Interrupt Enable), and TOV0 are set (one), the Timer/Counter0 Overflow interrupt is executed. In phase correct PWM mode, this bit is set when Timer/Counter0 changes counting direction at $00.