



**Prüfung:** μ-Computertechnik - Diplom  
**Termin:** Donnerstag, 10.07.2008; 11:00-13:00  
**Prüfer:** Prof. Walter  
**Hilfsmittel:** beliebig, keine Kommunikationsmittel

|                     |       |
|---------------------|-------|
| <b>Name:</b>        | _____ |
| <b>Vorname:</b>     | _____ |
| <b>Studiengang:</b> | _____ |
| <b>Labor:</b>       | _____ |
| <b>USB-Stick:</b>   | _____ |

Bitte überprüfen Sie, ob alle Protokolle des Labors in den Webseiten sind, inkl. Protokoll der Präsentation  
bitte keine rote Farbe verwenden

(nicht ausfüllen)!

| <b>Aufgabe</b> | <b>mögl. Punkte</b> | <b>erreichte Punkte</b> |
|----------------|---------------------|-------------------------|
| <b>1</b>       | <b>13</b>           |                         |
| <b>2</b>       | <b>17</b>           |                         |
| <b>3</b>       | <b>10</b>           |                         |
| <b>4</b>       | <b>10</b>           |                         |
|                |                     |                         |
| <b>Gesamt</b>  | <b>50</b>           |                         |
|                |                     |                         |
|                | <b>Note</b>         |                         |

**Bearbeiten Sie die Aufgaben nur, falls Sie keine gesundheitlichen Beschwerden haben.**

**Viel Erfolg!**

**Bemerkungen:** Leeren Sie bei Prüfungsbeginn den Stick. Bitte erstellen Sie die Lösungen auf der eigenen Festplatte und kopieren diese anschließend auf den Stick.

**Schreiben Sie in jeden Programmkopf ihren Namen! Bei nicht vorhandenem Namen wird die Lösung NICHT gewertet.**



**1. 8051/80535 Assembler „KARTSIM.ASM“ 13 Punkte  $\Sigma$ \_\_\_\_\_**

Das Ausgangssignal von vier Lasersensoren mit TTL-Signalaufbereitung bei einem Fahrzeug soll simuliert werden. Ein 80535-Controller EURO-535 führt diese Simulation durch. Folgende Kenndaten für das E-Kart sind gegeben.

$v_{max}=55\text{km/h}$  Maximalgeschwindigkeit

$d_{RAD}=39,5\text{ cm}$  Raddurchmesser

16= Anzahl der reflektierenden / nichtreflektierenden Flächen (Periodendauer)

- Zeichnen Sie das zu erwartende Signal für eine Umdrehung in das Diagramm ein. Beschriften Sie die Achsen. Beginnen Sie mit dem Low-Signal (schwarze Fläche)
- Berechnen Sie die kleinste auftretende Periodendauer
- Schreiben Sie die PDL für „KARTSIM.ASM“
- Erstellen Sie das Programm „KARTSIM.ASM“, welches mit Hilfe von Timer 2 an Port 1.1 und Port 1.2 das zu erwartende Signal bei Höchstgeschwindigkeit des Karts ausgibt. (Duty-Cycle 50%)
- Überprüfen Sie das Signal im Simulator mittels des Logikanalysators.

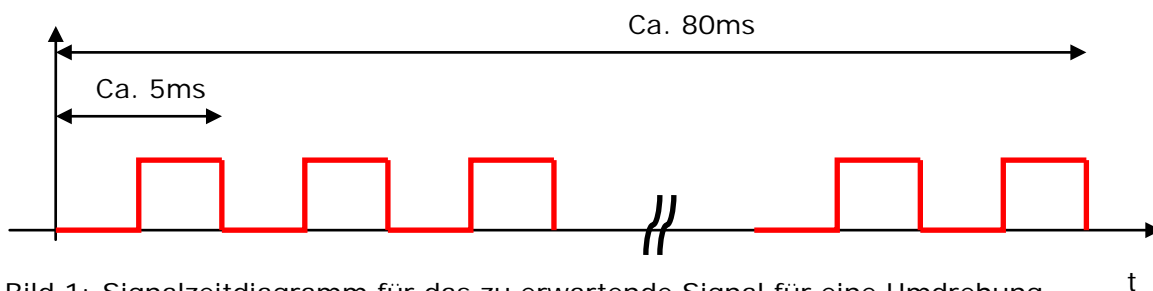


Bild 1: Signalzeitdiagramm für das zu erwartende Signal für eine Umdrehung



Bild 2: Reflexionsfläche für den Lasersensor



2. 8051/80535 Programm „KARTMESS.ASM“

17 Punkte Σ \_\_\_\_

Mit einem weiteren 80535-Mikrocontroller EURO\_535 werden die beiden Signale gemessen. Hierzu wird Port 1.1 des Mikrocontrollers zur Simulation der Signale auf die beiden Eingänge des Mikrocontrollers zur Messung für die linken Räder des E-Karts und Port 1.2 auf die beiden Eingänge für die rechten Räder des E-Karts geschaltet.

| Ausgang 80535<br>SIM | Eingang 80535<br>Mess | Bemerkung       | Kanal-<br>kennzeichnung |
|----------------------|-----------------------|-----------------|-------------------------|
| P1.1                 | P1.0                  | A13 RAD HR IEX3 | 0                       |
| P1.1                 | P1.1                  | C13 RAD VR IEX4 | 1                       |
| P1.2                 | P1.2                  | A14 RAD VL IEX5 | 2                       |
| P1.2                 | P1.3                  | C14 RAD HL IEX6 | 3                       |

- a) Zeichnen Sie einen Schaltplan für die beiden Mikrocontroller und verbinden Sie die angegebenen Signale.
- b) Schreiben Sie die PDL und das Programm KARTMESS.ASM mit folgenden Eigenschaften.

Bei steigender Flanke der Sensorsignale wird der jeweilige Wert von Timer 2 im externen Speicher abgelegt. Die Werte mit Kennzeichnung des Kanals finden sich anschließend beispielsweise in folgender Speicherform:

| Speicherzelle | Adresse | Bemerkung    |
|---------------|---------|--------------|
| 1             | 2000H   | Kanal 1      |
| CCL1          | 2001    |              |
| CCH1          | 2002    |              |
| 3             | 2003    | Kanal 3      |
| CCL3          | 2004    |              |
| CCH3          | 2005    |              |
| ~             | ~       |              |
| ~             | ~       |              |
|               | xxxxH   | Letzter Wert |
|               |         |              |
|               |         |              |

Insgesamt sollen 1365 Messwerte im externen Speicher abgelegt werden.



**3. 8051/80535 Programm „KARTANAL.ASM“ 10 Punkte Σ\_\_\_\_\_**

Für eine einfache Analyse werden die Werte über die serielle Schnittstelle an ein Terminalprogramm an den PC weitergegeben. Vor dort können die Werte per Copy and Paste in Excel eingefügt werden.

- a) Schreiben Sie die PDL und das Programm KARTANAL.ASM

**4. 8051/80535 Programm „KARTDIFF.ASM“ 10 Punkte Σ\_\_\_\_\_**

Schreiben Sie die PDL und ein Programm „KARTDIFF.ASM“, welches die Zeitdifferenzen für den Kanal 0 ab Adresse 3000H bis xxxxH ablegt.



Aufgabe 1

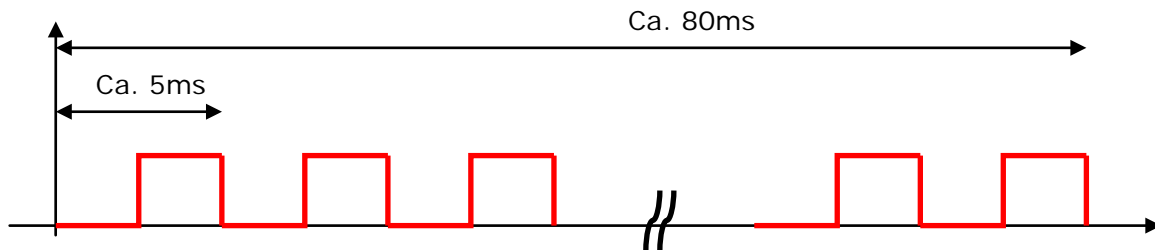


Bild 1: Signalzeitdiagramm für das zu erwartende Signal für eine Umdrehung

$$v_{max} = 55000m / 3600s = 15,277m/s$$

$$s = \pi * d = 3,14 * 0,395m = 1,241m$$

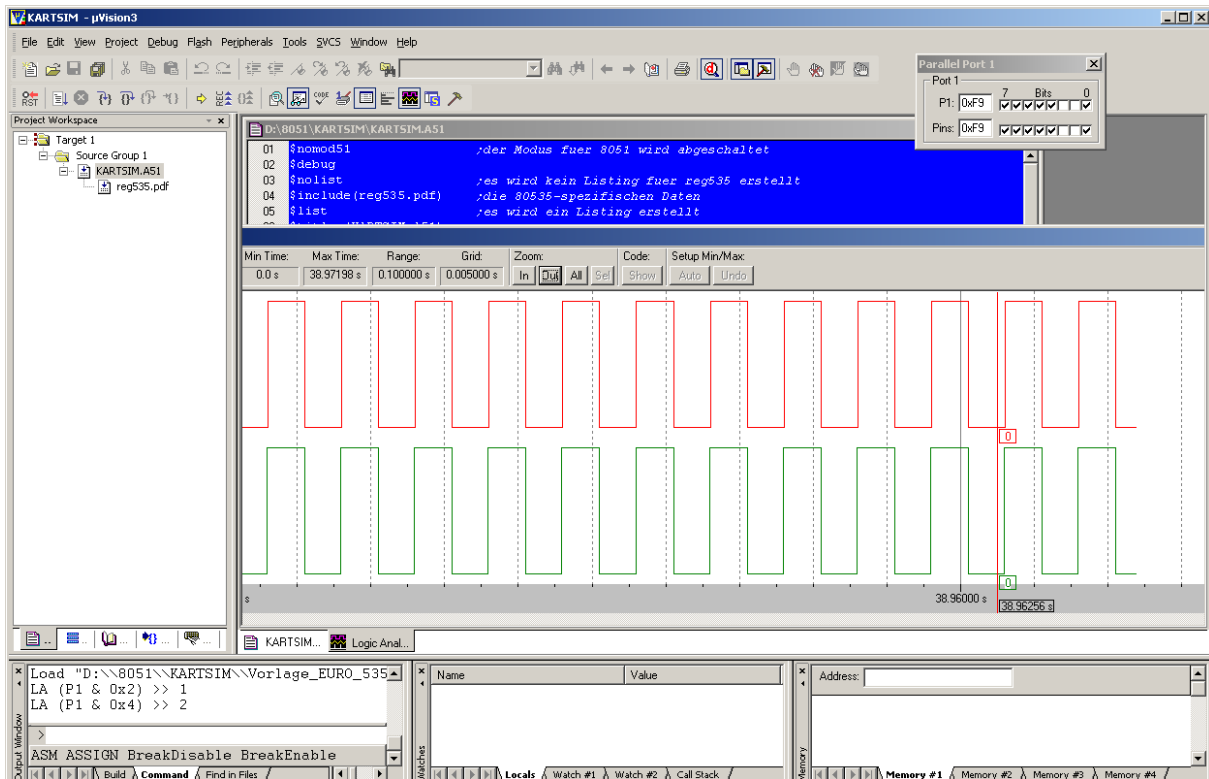
$$f_{max} = 12,31Hz * 16 \text{ Perioden / Umdrehung} = 197 \text{ Hz}$$

Tmin > ca. 5ms

```
$nomod51           ;der Modus fuer 8051 wird abgeschaltet
$debug
$no!ist           ;es wird kein Listing fuer reg535 erstellt
#include(reg535.pdf) ;die 80535-spezifischen Daten
$list            ;es wird ein Listing erstellt
$title (KARTSIM.A51)
;-----
;Programmbeschreibung
;-----
;Programm:
;PDL
;Ein Rechtecksignal mit der Periodendauer T>=5ms soll an Port 1.1 und Port1.2
;ausgegeben werden
;Timer 2 Initialisierung: CCEN --> Ausgabe
;Reload 65536-5000=60536=EC78H
;Compare: 65536-2500=63036 =F63CH
;Wiederladen bei Ueberlauf
;Timer 2 starten
;
;Erstellt am: Mittwoch, 30. Juli 2008 16:11:23
;Programmiert:
;
;Verwendete Einspruenge: keine
;
;Verwendete Unterprogramme: keine
;
;
;Verwendete Register und Variable:
;Registerbank(0)
;R2
;
;Kommentar:
;
;Aenderungen:
;Geaendert am: Mittwoch, 30. Juli 2008 16:11:23
;
;
;-----
;Initialisierungsteil fuer allgemeine Konstanten
```



```
;-----  
CSEG AT 0H ;Legt absolute Codesegmentadresse auf 0h  
jmp INIT  
;  
;Interrupt-Vektoren  
;-----  
;ORG  
;  
;-----  
;Initialisierungsteil fuer On-Chip Peripherie  
;  
ORG 100H ;Programmstart bei 100H  
INIT:  
;TIMER2 Initialisierung  
mov CRCH,#0ECH ;Reload Register 5ms  
mov CRCL,#78H  
mov CCH1,#0F6H ;Port 1.1 Compare- steigende Flanke  
mov CCL1,#3CH  
mov CCH2,#0F6H ;Port 1.2 Compare - steigende Flanke  
mov CCL1,#3CH  
mov CCEN,#00101000b ;Port 1.1 und Port 1.2 compare enabled  
setb T2R1 ;Autoreload bei Timer2 overflow  
setb T2I0 ;fosc/12 los gehts  
;  
;-----  
;Programmschleife  
;-----  
  
ABFRAGE:  
jmp ABFRAGE  
end
```





Aufgabe 2 „KARTMESS.ASM

```
$nomod51                ;der Modus fuer 8051 wird abgeschaltet
$debug
$no1ist                 ;es wird kein Listing fuer reg535 erstellt
#include(reg535.pdf)     ;die 80535-spezifischen Daten
$list                  ;es wird ein Listing erstellt
$title (KARTMESS.A51)
;-----
;Programmbeschreibung
;-----
;Programm:
;PDL:
;Timer 2 wird initialisiert
;16 Bit Zaehler, 1 MHz
;Interrupt Initialisierung
;Freigabe EX3 bis EX6
;ISR --> Speicherung mit Erhoehung und Zaehlen der Werte, R2, R3
;3*1365=4095=FFFH
;P1.0 - Kanal 0 - IEX3 EXTI3 EQU 053H
;P1.1 - Kanal 1 - IEX4 EXTI4 EQU 05BH
;P1.2 - Kanal 2 - IEX5 EXTI5 EQU 063H
;P1.3 - Kanal 3 - IEX6 EXTI6 EQU 06BH
;IEXn wird durch Hardware - Interruptvektor zurueckgesetzt
;
;
;Erstellt am: 1. Juni 2008 04:41:23
;Programmiert: Juergen Walter
;
;Verwendete Einspruenge: keine
;
;Verwendete Unterprogramme: keine
;
;
;Verwendete Register und Variable:
;Registerbank(0)
;R2
;
;Kommentar:
;
;Aenderungen:
;Geaendert am: 1. Juni 2008 04:41:23
;
;
;-----
;Initialisierungsteil fuer allgemeine Konstanten
;-----

CSEG AT 0H                ;Legt absolute Codesegmentadresse auf 0h
jmp INIT
;
;-----
;Interrupt-Vektoren
;-----
ORG 053H
call ISR_EX3_P1_0
reti
ORG 05BH
call ISR_EX4_P1_1
reti
ORG 063H
call ISR_EX5_P1_2
reti
ORG 06BH
call ISR_EX6_P1_3
```



```
reti
;-----
;Initialisierungsteil fuer On-Chip Peripherie
;-----
ORG 100H                ;Programmstart bei 100H
INIT:
;TIMER 2
setb T2I0                ;Timer zaehlt mit 1 MHz
setb I3FR                ;EX3 steigende Flanke
                        ;T2CON 010x 0001

mov CCEN,#01010101b    ;CCEN
;Interrupt
;mov IEN1,00111100    ;Interrupts freigeben
setb EX3                ;Freigabe Interrupt P1.0 EX3
setb EX4                ;Freigabe Interrupt P1.1 EX4
setb EX5                ;Freigabe Interrupt P1.2 EX5
setb EX6                ;Freigabe Interrupt P1.3 EX6
setb EAL                ;Alle Interrupts freigeben
;DPTR initialisieren
mov DPTR,#2000H        ;Anfangsadresse
mov R3,#0FH            ;Zaehlregister
mov R2,#0FFH
;-----
;Programmschleife
;-----
```

```
ABFRAGE:
jmp ABFRAGE
```

```
ISR_EX3_P1_0:
call IST_ENDE          ;Messung Ende?
mov A,#0                ;Kanal 0
movx @DPTR,A
inc DPTR                ;
mov A,CRCL
movx @DPTR,A          ;CC0 in ext. Datenspeicher
inc DPTR
mov A,CRCH
movx @DPTR,A
inc DPTR
ret
```

```
ISR_EX4_P1_1:
call IST_ENDE          ;Messung Ende?
mov A,#1                ;Kanal 1
movx @DPTR,A
inc DPTR                ;
mov A,CCL1
movx @DPTR,A          ;CC0 in ext. Datenspeicher
inc DPTR
mov A,CCH1
movx @DPTR,A
inc DPTR
ret
```

```
ISR_EX5_P1_2:
call IST_ENDE          ;Messung Ende?
mov A,#2                ;Kanal 2
movx @DPTR,A
inc DPTR                ;
mov A,CCL2
movx @DPTR,A          ;CC0 in ext. Datenspeicher
inc DPTR
mov A,CCH2
movx @DPTR,A
inc DPTR
```





```
ret

ISR_EX6_P1_3:
call IST_ENDE           ;Messung Ende?
mov A,#3                ;Kanal 3
movx @DPTR,A
inc DPTR
mov A,CCL3              ;
movx @DPTR,A           ;CC0 in ext. Datenspeicher
inc DPTR
mov A,CCH3
movx @DPTR,A
inc DPTR
ret

IST_ENDE:
djnz R2,WEITER_0       ;Messung Ende?
djnz R3,WEITER_0
clr EAL
clr P3.2               ;L1 geht an
WEITER_0:
ret

end
```



