



### Aufgabe 1: Entwurf von Assemblerprogrammen

- a) Nennen Sie die wichtigsten Vorteile der Assemblerprogrammierung gegenüber einer direkten Programmerstellung in Maschinensprache!

In folgenden Teilaufgaben werden zwei kurze, einfache Programmabschnitte entwickelt, die leicht abgewandelt beim Entwurf komplexerer Programme einsetzbar sind.

CLR C	Carry löschen -> CLR bit ; bit löschen
INC Rn	Register $n$ inkrementieren -> INC A ; Akku ink.
ACC.7	Bit 7 (höchstwertiges Bit) des Akkumulators; gibt bei Zweierkomplementdarstellung das Vorzeichen des Akkumulators an
MOV dest,source	source nach dest schreiben
SUBB A,Rn	Subtraktion mit „Borgen“, $A := A - Rn - C$
JB bit,Adresse	Sprung an Adresse, wenn bit gesetzt ist
JMP Adresse	unbedingter Sprung an Adresse
NOP	keine Operation
RET	Rücksprung aus einem Unterprogramm
RRC A	Akku nach rechts durch Carry schieben -> RLC A; nach links
RL A	Akku nach links schieben aber nicht durch Carry (vgl. RLC)
JNB bit,Adresse	Sprung an Adresse, wenn bit nicht gesetzt ist
ADD A,source	Inhalt von A um den Inhalt von source erhöhen

Tabelle 1.1: Befehlsauszug in 8051-Assemblersprache

- b) Schreiben Sie ein Unterprogramm `Add48`, das zum Inhalt des Registers R0 den Wert 48 (dezimal) addiert! Verwenden Sie zur Berechnung den Akkumulator A!
- c) Schreiben Sie ein Unterprogramm `Divider`, das eine 16 bit-Zahl, die in den Registern R1 und R0 steht (höherwertiger Anteil in R1), durch 2 dividiert!

#### **Hinweise:**

- Beachten Sie, dass eine Division durch 2 durch eine Schiebeoperation realisiert werden kann.
- Verwenden Sie für die Berechnungen den Akkumulator A.
- Das Ergebnis soll am Ende wieder in den Registern R1 und R0 stehen.
- Gehen Sie davon aus, dass das Carry-Bit zu Beginn des Unterprogramms gelöscht ist.

**Divider:**

**Ret**



### Aufgabe 2: Mikrocontrollerprogrammierung mit dem 80C51

Im ersten Aufgabenteil soll ein Unterprogramm für den 80C51-Mikrocontroller entwickelt werden, dessen Flussdiagramm in Abbildung 2.1 gegeben ist. Dem Unterprogramm werden in den Registern  $R0$ ,  $R1$ ,  $R2$  beliebige vorzeichenlose 8 bit-Dualzahlen übergeben.

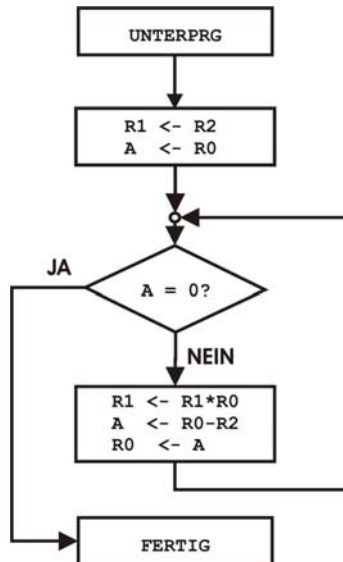


Abbildung 2.1: Flussdiagramm des Assemblerprogramms

- Schreiben Sie das entsprechende Assemblerprogramm für den in Abbildung 2.1 gegebenen Abschnitt! Vergessen Sie dabei nicht den Rücksprung ins Hauptprogramm und benutzen Sie die Bezeichnungen  $R0$ ,  $R1$ ,  $R2$ ,  $A$ ,  $FERTIG$  und  $UNTERPRG$  aus Abbildung 2.1.
- Welche Dualzahlen stehen nach Ablauf des Programms in  $R0$ ,  $R1$  und  $R2$ , wenn in den Registern zu Beginn des Programmsegments vorzeichenlose 8 bit-Dualzahlen mit folgenden Dezimalwerten stehen:  
 $R0: 6_d$ ,  $R1: 4_d$ ,  $R2: 3_d$
- Welche arithmetische Operation wird durch das Programmsegment ausgeführt, wenn zu Anfang  $R2$  den Wert  $1_d$  hat und das Ergebnis in  $R1$  in Abhängigkeit von  $R0$  ausgedrückt wird?

### Aufgabe 3: Baudratengenerierung mit dem 80C51

Unter Verwendung des integrierten Timers 1 bietet der Mikrocontroller 80C51 vielfältige Möglichkeiten der Baudratengenerierung für die serielle Datenübertragung.

Im Folgenden soll die Betriebsart 1 der seriellen Schnittstelle verwendet werden. Diese erlaubt die asynchrone Datenübertragung mit variabler Baudrate (8 bit UART), wobei als Zeitbasis die Überlaufrate von Timer 1 dient.



Zunächst soll eine Baudrate von 4800 bit/s generiert werden. Das Steuerbit SMOD habe den Wert 1. Um die Baudratengenerierung mit möglichst geringer Prozessorbelastung durchzuführen, soll Timer 1 im Autoreload-Modus betrieben werden. Der Autoreload-Wert für Timer 1 sei der Hexadezimalwert *EE*.

- a) Geben Sie die Belegung der Spezialfunktionsregister SCON und TMOD in Tabelle 3.1 an und kennzeichnen Sie irrelevante Bits durch 'x'!

	Bit 7							Bit 0
SCON								
TMOD								

Tabelle 3.1: Belegung der Spezialfunktionsregister

- b) Berechnen Sie die Taktfrequenz  $f_{osc}$ , die am Eingang von Timer 1 benötigt wird, damit eine Baudrate von exakt 4800 bit/s erzeugt wird!
- c) Wie groß ist der relative Fehler der erzeugten Baudrate, wenn der Prozessor mit 16 MHz getaktet wird?
- d) Geben Sie für SMOD = 1 die maximale Baudrate bei Betrieb des Timers im Autoreload-Modus an, wenn der Prozessor nun mit 12 MHz getaktet wird!

#### **Aufgabe 4: Timer/Counter des Mikrocontrollers 80C52**

Ein Mikrocontroller vom Typ 80C52 soll zur Überwachung eines Impulsgebers eingesetzt werden, der bis zu 20.000 Impulse pro Sekunde liefern kann. Dazu wird Timer 0 als Zähler verwendet, der die Impulse am T0-Pin jeweils während einer Zeit von 10 ms zählt. Timer 1 hat die Aufgabe, alle 10 ms einen Interrupt auszulösen. Die Taktfrequenz des Mikrocontrollers beträgt 12 MHz.

- a) Berechnen Sie den Reloadwert für Timer 1! Vernachlässigen Sie hierbei die Zeitverzögerung, die durch den Aufruf der Interruptroutine entsteht!
- b) Geben Sie in Tabelle 4.1 die notwendigen Einstellungen der Spezialfunktionsregister an, um die beiden Timer zu initialisieren und zu starten! Kennzeichnen Sie irrelevante Bits durch „X“!

Bit	7	6	5	4	3	2	1	0
IE								
TMOD								
TCON								

Tabelle 4.1: Belegung der Spezialfunktionsregister



In der Interruptroutine von Timer 1 wird der Zählerstand von Timer 0 (TL0) ausgelesen und in das Register R0 geschrieben. Wurde der Wert von 15.000 Impulsen pro Sekunde überschritten, wird der Portpin P1.0 auf Low gesetzt, ansonsten auf High. Außerdem muss Timer 1 neu geladen werden.

- c) Vervollständigen Sie das Flussdiagramm in Abbildung 4.1 für die Interruptroutine von Timer 1! Geben Sie dabei neben dem Kommentar jeweils auch den entsprechenden Assemblerbefehl an.

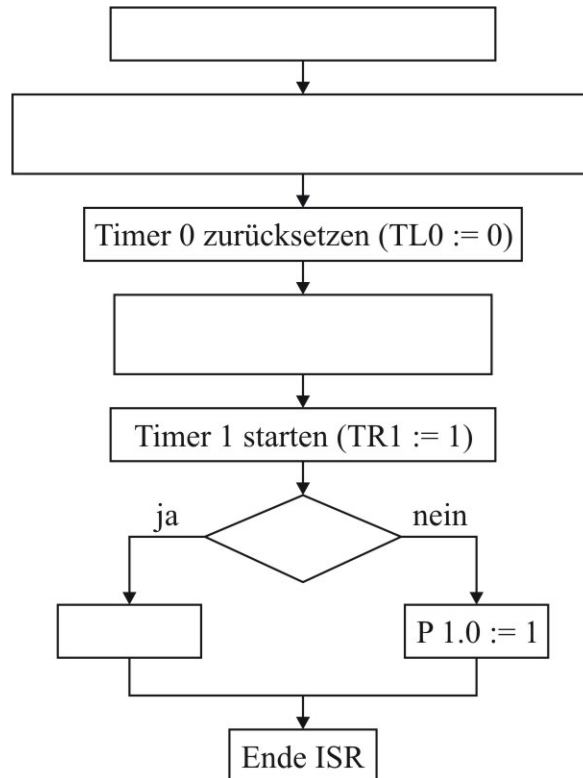


Abbildung 4.1: Flussdiagramm der Interruptroutine von Timer 1