

Einsatz eines 8031-Experimentiersystems im Unterricht

Andreas HEINBACH, HTL Wien-I

TGM-DSK-169: ADAUTO.ASM, TGM-LIT-020

An der Abendschule der HTL Wien1, Abteilung Elektrotechnik findet ein zweisemestriger Lehrgang für Berufstätige für Informatik und Mikroelektronik statt. Teil des Unterrichts ist ein vierstündiges Laboratorium, das geblockt in zwei mal sechs Abenden zu je fünf Stunden gehalten wird. Erstmals im Sommersemester 1991 wurde es den Teilnehmern freigestellt, an zwei der zwölf angebotenen teilzunehmen. Das heißt, das Thema 8031-Prozessoren wurde an sechs Abenden mit ein und derselben Gruppe intensiv behandelt.

Durch dieses System sollte einerseits den Teilnehmern berufsspezifische Weiterbildung ermöglicht werden, andererseits konnte der Vortragende erwarten, mit seinem Wissensangebot auf großes Interesse zu stoßen. Als Problem erwies sich jedoch - wie wahrscheinlich in jedem Abendlehrgang - die Aufnahmefähigkeit, die nach einem achtstündigen Arbeitstag natürlich nicht mehr uneingeschränkt vorhanden ist. Auch auf divergierende Grundkenntnisse im Elektronikbereich mußte Rücksicht genommen werden.

Für die Durchführung der Übungen stand ein PC mit zwei Laufwerken zur Verfügung. Aus Komfort- und Datensicherheitsgründen wären 3½" HD-Laufwerke von Vorteil. Die Übungen fanden im Elektrotechnik Labor statt, Elektronik-Kleinmaterial war vorhanden. Eine Experimentierplatine MCB515 der Firma KEIL mit einem Monitorprogramm in einem Eprom wurde zusammen mit einer einfachen Peripherieplatine mit zweistelligem LED-Display, Optokoppler, Potentiometer an einem Analogeingang und einem Taster zum Austesten der Programme verwendet.

Die Mikrocontrollerprogramme konnten sowohl am PC simuliert werden, als auch mit der selben Benutzeroberfläche mit der Hardware ausgetestet werden. Wir verwendeten dafür die Programme DScope51 und MonScope51, ebenfalls von der Firma KEIL. Die Programme stellen zwei Einzelschrittmodi, Breakpoints und Watchpoints mit programmierbaren Prozeduren zur Verfügung. Die Bedienung der Programme ist auch für Ungeübte relativ rasch erlernbar, jedoch wäre eine ständige Anzeige des RAM-Bereiches (bzw. nur der Special Function Register) sicherlich vorteilhaft.

Wir versuchten, pro Laborabend ein Thema, bzw. eine Applikation zu erarbeiten. Am Anfang stand eine kurze theoretische Einführung, danach wurde eine Aufgabe formuliert, die bis zum Ende des Abends gelöst sein sollte.

Als Beispiel für so eine Aufgabe sei ein einfaches Spannungsmeßgerät mit automatischer Meßbereichsumschaltung zu programmieren. Am Analogeingang liegt eine Spannung zwischen 0V und 5V. Der Gesamtmeßbereich soll aufgrund der tatsächlich anliegenden Spannung folgendermaßen aufgeteilt werden:

```
0....1.25 V Meßbereich0
1.25..2.5 V Meßbereich1
2.5...3.75 V Meßbereich2
3.75..5 V Meßbereich3
```

Im Programm wird bei der ersten Messung aufgrund der beiden höchstwertigen Bits des Meßergebnisses der Meßbereich für eine zweite Messung eingestellt und sodann zuerst Bereich und dann - nach einer Pause - der zweite Meßwert über eine Tabelle auskodiert und auf Port1 gelegt. Port1 liegt an beiden 7-Segment-Anzeigen, mit Port3.3 und Port3.4 können diese einzeln geschaltet werden. Durch die Meßbereichsumschaltung wird die Auflösung der Messung von 8 auf 10bit (5mV) erhöht, wobei nicht vergessen werden darf, daß die Genauigkeit des Vergleichs nur 10mV beträgt.

ADAUTO.ASM

```
$ NOMOD51
$ INCLUDE (c:REG515.inc)
org 0000h
reset:  mov Dptr,#Tabelle ; Zeichentabelle 7 Segment
start:  mov R0,#00 ; 0 V bis 5 V
call wandle
call bereich
call wandle
call ausgabeB
mov a,r6
call ausgabeN
call paus1
jmp start
wandle: ; einlesen Kanal 0
; R0 Wert für DAPR ( Bereich ), Ergebnis in r6
loop:  clr mx0 ; wähle Kanal 0
clr mx1
clr mx2
mov DAPR,R0 ; starte Wandlung
jb BSY,$ ; warte bis fertig umgewandelt
mov r6,ADDAT
ret
bereich: ;Setzt r0 in Abhängigkeit des Wandlereerg
;nr des messber ==> r7, Wert f. DAPR ==> r0
mov a,r6
jb acc.7,s1
jb acc.6,s2
mov r0,#40h
mov r7,#00
jmp aus
s2:  mov r0,#84h
mov r7,#01
jmp aus
s1:  jb acc.6,s3
mov r0,#0c8h
mov r7,#02
jmp aus
s3:  mov r0,#0ch
mov r7,#04
aus:  ret
ausgabeN: ; Ausgabe Akku als HexZahl
mov b,#10h
div ab
cjne a,#00,zehner
mov r1,#0 ; 1. Stelle 0, Anzeige dunkel
jmp einer
zehner: movc A,@A+Dptr
mov R1,A
einer: movc A,@A+Dptr
mov R2,A
mov r3,#07fh
Anzeige: mov P1,R1 ; 1. Stelle
clr P3.4
setb P3.4
mov P1,R2 ; 2. Stelle
clr P3.3
setb P3.3
djnz r5,Anzeige
djnz r3,Anzeige ; Anzeige-/ Samplezeit
ret
ausgabeB: mov a,r7
movc a,@a+dptra
mov p1,a
clr p3.3
call paus1
setb p3.3
ret
Pause: R3,#10h
Pause1: djnz R3,Pause1
ret
paus1: mov r3,#0ffh
mov r4,#04
l3:  djnz r5,l3
djnz r3,l3
djnz r4,l3
ret
; Segment GFEDCBA
Tabelle: DB 10111111b; 0
DB 10000110b; 1
DB 11011011b; 2
DB 11001111b; 3
DB 11100110b; 4
DB 11101101b; 5
DB 11111101b; 6
DB 10000111b; 7
DB 11111111b; 8
DB 11100111b; 9
DB 11101111b; A
DB 11111100b; B
DB 10111001b; C
DB 11011110b; D
DB 11111001b; E
DB 11110001b; F
end
```