

Real-Time-Clock für PC 100

Implementierung einer REAL-TIME-CLOCK auf einem SIEMENS PC 100 Mikrocomputer

Für spezielle Anwendungszwecke wäre ein Zugriff auf eine interne Uhr vom BASIC-Programm aus eine sehr nützliche Hilfe. Der SIEMENS PC 100 bzw. AIM 65 bietet diese Möglichkeit leider nicht.

Da das Gerät aber über einen 16 Bit Zähler in Form des VIA 6522 verfügt, kann man über die hier beschriebene Software die Clock Funktion simulieren. Das Verfahren kann natürlich auch beim AIM 65 angewendet werden. Ein Zugriff auf die Uhr ist über den BASIC-Befehl PEEK jederzeit möglich. Die Uhr muß dabei parallel zum BASIC-Programm arbeiten. Der Prozessor müßte also die Möglichkeit von Multitasking bieten.

Das nun folgende Programm sollte für den Leser nur eine Anregung sein und ist sicherlich noch wesentlich zu verbessern.

Zählerinitialisierung

```
(K)*=0300
/12
0300 48 PHA
0301 A9 LDA #FF
0303 8D STA A002
0306 A9 LDA #C0
0308 8D STA A00B
030B 8D STA A00E
030E A9 LDA #FF
0310 8D STA A004
0313 8D STA A005
0316 58 CLI
0317 68 PLA
0318 60 RTS
```

Programmteil 1 von Adresse 0300 - 0318 initialisiert den Zähler mit der Anzahl der Taktschritte und startet den Zählvorgang. Anschließend

wird noch das Interrupt Disable Bit gelöscht, so daß der Prozessor jetzt auf die IRQ Unterbrechung reagiert, die der Zähler liefert sobald der Zählvorgang beendet ist. Nach Ablauf der Initialisierung kann der Prozessor wieder für eine andere Tätigkeit arbeiten, d.h. er kann das BASIC-Programm fortsetzen.

Hat der parallel zum Prozessor arbeitende Zähler die vorgegebene Anzahl von Taktzyklen abgezählt, gibt er eine IRQ Unterbrechung. Der Prozessor beendet das augenblickliche aktive BASIC-Programm und springt zu der Adresse auf die der IRQV4 Vektor (Adresse A400 / A401) verweist. IRQV4 muß also mit der Adresse des Interruptbehandlungsprogrammes geladen werden.

Interruptbehandlung

```
(K)*=0400
/32
0400 20 JSR 0300
0403 48 PHA
0404 8A TXA
0405 48 PHA
0406 98 TYA
0407 48 PHA
0408 A2 LDX #3B
040A A0 LDY #17
040C A9 LDA #0E
040E EE INC 03FF
0411 CD CMP 03FF
0414 10 BPL 043C
0416 A9 LDA #00
0418 8D STA 03FF
041B EE INC 03FE
041E EC CPX 03FE
0421 10 BPL 043C
0423 8D STA 03FE
0426 EE INC 03FD
0429 EC CPX 03FD
042C 10 BPL 043C
042E 8D STA 03FD
0431 EE INC 03FC
```

```

0434 CC CPY 03FC
0437 10 BPL 043C
0439 8D STA 03FC
043C 68 PLA
043D A8 TAY
043E 68 PLA
043F AATAX
0440 68 PLA
0441 40 RTI

```

Führt der Prozessor den Interrupt aus, so muß als erster Befehl sofort der Zählvorgang erneut gestartet werden. Erst jetzt kann man den auf diese Weise gewonnenen konstanten Zeittakt auswerten. Im Beispiel wird der Takt etwa auf Sekundenimpulse geteilt und in Minuten bzw. Stundentakte weiterverarbeitet. Dabei entsprechen ca. 14 Interrupttakte einer Sekunde. Die Zeitsignale stehen folgendermaßen im Speicher:

Adresse (Hex)	Zeitsignal
03FF	Interrupttakt ca. 71 mS
03FE	Sekunden
03FD	Minuten
03FC	Stunden

Mit den angegebenen Werten arbeitet die Uhr nicht ganz exakt, sie geht etwas zu schnell. Durch Veränderung der Anzahl der Taktimpulse bzw. des Teilverhältnis (Befehl in Adresse 040C LDA xx) kann man die Uhr aber exakt eichen.

Nach Beendigung der Interruptbehandlung kehrt der Prozessor wieder zum unterbrochenen BASIC-Programm zurück.

Nachdem wir die beiden Assemblerprogramme eingeschrieben haben und den IRQV4 Vektor geladen haben, können wir jetzt das BASIC-System starten. Um die Assemblerprogramme nicht zu zerstören muß man entsprechend Speicherplatz reservieren.

BASIC-Programm

(M)=A400 00 04 7B E0

(5)

MEMORY SIZE? 750

WIDTH?

220 BYTES FREE

SIEMENS PC100 BASIC

10 POKE 04,00

```

20 POKE 05,03
30 FOR I=0 TO 3
40 POKE 1020+I,00
50 NEXT I
60 X=USR(0)
70 PRINT PEEK(1020);
80 PRINT PEEK(1021);
90 PRINT PEEK(1022)
100 S1=PEEK(1022)
110 IF S1=PEEK(1022)
    THEN 110
120 GOTO 70

```

In Zeile 10 und 20 des BASIC-Programmes wird die Anfangsadresse des Zählerstartprogrammes in die USR - Adresse geschrieben. Danach werden die Zeitspeicher mit 0 initialisiert (man kann natürlich auch die aktuelle Zeit eingeben).

Die Uhr muß nur einmal gestartet werden und läuft dann parallel zu allen anderen Aktivitäten. Gestoppt werden kann die Uhr nur über die RESET-Taste. Zeile 60 des BASIC-Programmes startet die Uhr. Wird das BASIC-Programm gestoppt, so läuft die Uhr weiter. Man kann dies leicht überprüfen, indem man die BREAK-Taste (F1) betätigt und einige Zeit wartet um danach GOTO 70 einzugeben. Man sieht, daß die Uhr wiedergelaufen ist.

Folgende Befehle ergeben die Zeit:

```

PEEK( 1020 ) → Stunde
PEEK( 1021 ) → Minute
PEEK( 1022 ) → Sekunde
PEEK( 1023 ) → Takt

```

Da die Programme nur Beispiele sind, die die Vorgehensweise zeigen sollen, kann man die beiden Assemblerprogramme platzsparend zu nur einem Block zusammenfassen.

Wegen der ständigen Unterbrechungen des Prozessors können unangenehme Überraschungen auftreten wenn der Thermodrucker oder das Kassetteninterface aktiv sind. Die Uhr sollte vorher besser abgeschaltet werden.

Trotz des Parallelbetriebes werden BASIC-Programme fast nicht merkbar langsamer.

Dietmar Wäber ■