

Aufgabenpaket 3

Aufgabe 3.1

Bestimmen Sie die MIPS-Rate des *ToyEmulators* aus Aufgabenpaket 1 auf verschiedenen Rechnern des Computerkabinetts. Schreiben Sie sich zu diesem Zwecke ein Toy-Programm mit Codelänge N und messen Sie die Laufzeit T Ihres Programmes in Sekunden. Hierbei sind konventionelle Zeitmeßgeräte wie Stopp- oder Armbanduhrn verwendbar.

Der Quotient $(N / 1000000) / T$ ist dann Ihre MIPS-Rate.

Aufgabe 3.2

Messen Sie die MIPS-Rate verschiedener PC's im Computerkabinett. Schreiben Sie sich dazu zwei Assembler-Programme in denen Sie die Systemzeit T_1 auslesen und auf dem Bildschirm ausgeben, dann ein Programmstück bekannter Codelänge N abarbeiten und schließlich erneut die Systemzeit T_2 ermitteln. Der Unterschied zwischen den Programmen soll in der Komplexität der verwendeten Assembler-Anweisungen bestehen. Verwenden Sie für ein Testprogramm nur `<inc>`, `<dec>` und ähnlich "leichte" Befehle und für das andere vorrangig `<mul>`, `<div>` etc. Um meßbare Laufzeiten zu erhalten, empfiehlt es sich, mit verschachtelten Schleifen zu arbeiten!

Der Quotient $(N / 1000000) / (T_2 - T_1)$ gibt dann an, wieviel Millionen Instruktionen pro Sekunde der Rechner bearbeiten kann.

Abzugeben sind kommentierte Ausdrücke der Quelltext und die Testprotokolle
Begründen Sie Unterschiede der MIPS-Raten in Aufgabe 2 und diskutieren Sie die Tauglichkeit dieser Art Rechner zu vergleichen.

Hinweise: Zum Auslesen der Systemzeit kann folgendes Code-Fragment benutzt werden:

```
mov ah,2ch  
int 21h ; Interrupt 21hex mit Option 2Chex auslösen
```

danach gilt folgende Registerbelegung:

CH = Stunden
CL = Minuten
DH = Sekunden
DL = Hundertstel Sekunden (für die Aufgabe nicht benötigt)

Alle Zahlen sind als Hexadzimalzahlen gespeichert. Zur Ausgabe entsinnen Sie sich bitte des Makros `SHOW_NUMBER_IN_AL` aus der Datei `MACRO.BIB` (Aufgabenpaket 2).

Nach Umwandlung in eine Subroutine (`PROC NEAR`) kann dieses Codestück in Verbindung mit dem Makro `SHOW_DL` zur Ausgabe der einzelnen Zeiteinheiten benutzt werden.

"Eine gründliche Einführung" in die Assembler-Programmierung finden Sie in der Informatikbibliothek (Wolfgang Link: "Assembler Programmierung - Eine gründliche Einführung").