Rechnerorganisation: Weiter Macrobeispiele (i86, MASM Assembler)

Robert Tolksdorf
Freie Universität Berlin

Macros für Verzweigungen

- Assemblercode mit Macros:

```
.IF cx == 20
mov dx, 20
.ELSE
mov dx, 30
.ENDIF
```

- Erzeugter Assemblercode:

```
.IF cx == 20
0017 83 F9 14 * cmp cx, 014h
001A 75 05    * jne @C0001
001C BA 0014    mov dx, 20
.ELSE
001F EB 03    * jmp @C0003
0021          *@C0001:
0021 BA 001E    mov dx, 30
.ENDIF
0024          *@C0003:
```


Makros für Schleifen

-
```
.REPEAT
*@C0001:
  inc ax
.UNTIL ax==6
* cmp ax, 006h
* jne @C0001
```

-
```
.REPEAT
*@C0003:
  mov ax, 1
.UNTILCXZ
* loop @C0003
```

-
```
.REPEAT
*@C0004:
.UNTILCXZ [bx].field != 6
* cmp [bx].field, 006h
* loope @C0004
```


Schleifen mit Abbruch

-
```
.WHILE 1    ; Loop forever
mov ah, 08h ; Get key without echo
int 21h
.BREAK .IF al == 13 ; If ENTER, break out of the loop
.CONTINUE .IF (al < '0') || (al > '9')
; If not a digit, continue looping
mov dl, al ; Save the character for processing
mov ah, 02h ; Output the character
int 21h
.ENDW
```

Schleifen mit Abbruch

```
.WHILE 1
0017        *@C0001:
0017 B4 08    mov ah, 08h
0019 CD 21    int 21h
.BREAK .IF    al == 13
001B 3C 0D *  cmp al, 0Dh
001D 74 10 *  je @C0002
.CONTINUE .IF (al < '0') || (al > '9')
001F 3C 30 *  cmp al, '0'
0021 72 F4 *  jb @C0001
0023 3C 39 *  cmp al, '9'
0025 77 F0 *  ja @C0001
0027 8A D0    mov dl, al
0029 B4 02    mov ah, 02h
002B CD 21    int 21h
.ENDW
002D EB E8 *  jmp @C0001
002F *@C0002:
```


Prozeduraufrufe

```
myproc PROC NEAR PASCAL USES di si,
arg1:WORD, arg2:WORD, arg3:WORD
LOCAL local1:WORD, local2:WORD
```

Vorspann:

```
push bp ; Step 1:
mov bp, sp ; point BP to stack top
sub sp, 4 ; Step 2: space for 2 local words
push di ; Step 3:
push si ; save registers listed in USES
```

Nachspann:

```
pop si ; Undo Step 3
pop di
mov sp, bp ; Undo Step 2
pop bp ; Undo Step 1
ret 6 ; Clean stack of pushed arguments
```

Aufruf:

```
invoke myproc, 1, 2, 3
```