

# Logik, Sprünge und Unterprogramme

Der Befehlssatz des Prozessors enthält die Befehle AND, OR, XOR, TEST und NOT der booleschen Logik. D

Das Format für diese Befehle ist folgendes:

| Anweisung | Format im Programmcode  |
|-----------|-------------------------|
| AND       | AND operand1, operand2  |
| OR        | OR operand1, operand2   |
| XOR       | XOR operand1, operand2  |
| TEST      | TEST operand1, operand2 |
| NOT       | NOT operand1            |

Der erste Operand kann entweder in einem Register oder im Speicher sein. Der zweite Operand kann entweder in einem Register/Speicher oder ein unmittelbarer (konstanter) Wert sein. Speicher-zu-Speicher-Operationen sind nicht möglich.

Die Operatoren werden bitweise ausgeführt je nach Ergebnis werden die Flags CF, OF, PF, SF oder ZF gesetzt.

## Die AND-Instruktion

Die AND-Anweisung vergleicht zwei Operanden indem sie eine bitweise AND-Operation durchführt. Die bitweise UND-Verknüpfung ergibt 1, wenn die übereinstimmenden Bits beider Operanden 1 sind, andernfalls ergibt sie 0.

Das Ergebnis der Operation wird im ersten Operand gespeichert.

Beispiel:

```
Operand1:  1010 0101
Operand2:  1000 0011
-----
AND -> Operand1: 1000 0001
```

Beispiel:

```
section .data
tabelle TIMES 10 DW 97

section .text
global _start      ;must be declared for linker (ld)
_start:            ;tell linker entry point

MOV BL, 01111010B
```

```
AND BL, 11110000B

MOV [tabelle], BL

;tabelle ausgeben
mov  edx,20      ;message length
mov  ecx,tabelle ;message to write
mov  ebx,1       ;file descriptor (stdout)
mov  eax,4       ;system call number (sys_write)
int  0x80        ;call kernel

mov  eax,1       ;system call number (sys_exit)
int  0x80        ;call kernel
```

An folgendem Beispiel kann man sich einige neue Möglichkeiten erschließen:

From:  
<https://wiki.qg-moessingen.de/> - QG Wiki

Permanent link:  
<https://wiki.qg-moessingen.de/faecher:informatik:oberstufe:techinf:assembler:logik:start?rev=1632164981>

Last update: 20.09.2021 21:09

