

Assembler ASEM-51

1. Anweisungen:

Assemblerbefehl: [symbol:] [instruction [arguments]] [;comment]

Definition: symbol instruction argument [;comment]

Steuerung: \$control [(argument)] [;comment]

Beispiele:

```
HERE:    MOV A,#0FFH    ;define label HERE and load A with FFH
YEAR     EQU 1999      ;define symbol for current year
$INCLUDE (80C517.MCU)  ;include SAB80C517 register definitions
```

2. Symbole:

zulässige Zeichen: A .. Z, 0 .. 9, _, ?

Länge: max. 31 relevante Zeichen

1. Zeichen: A .. Z, _, ?

3. Konstanten:

Zahlenkonstanten:

zulässige Zeichen: 0 .. 9, A .. F

1. Zeichen: 0 .. 9

<u>Typ</u>	<u>Zeichen</u>	<u>Suffix</u>
Binär	0 ... 1	B
Oktal	0 ... 7	Q oder O
Dezimal	0 ... 9	D oder ohne
Hex	0 ... F	H

Beispiele:

```
10011011B = 233Q = 155D = 155 = 9BH
10111100B = 274Q = 188D = 188 = 0BCH
```

Zeichenkonstanten:

einzelnes Zeichen: 'X'

Zeichenketten: 'Dies ist eine Zeichenkette!'

4. Ausdrücke:

zulässige Operanden:

Zahlen (siehe oben)

Benutzer-Symbole

Vordefinierte Symbole (z.B. SFR-Namen)

\$ (steht für den akt. Programmzähler)

zulässige Operatoren:

Unär: + - NOT HIGH LOW

Binär: + - * / MOD SHL SHR AND OR XOR .

<u>Prioritäten:</u>	()	hoch
	NOT HIGH LOW	↑
	.	
	* / MOD	
	+ -	
	AND	
	OR XOR	niedrig

5. Pseudo-Befehle (Assembler-Steuerungen)

Definiere Bytes: DB <arg1> [,<arg2> [,<arg3> ...]]

Definiere Worte: DW <arg1> [,<arg2> [,<arg3> ...]]

Reserviere Speicherplatz: DS <exp>

Reserviere Platz für Bits: DBIT <exp>

Beispiele:
DB 1,2,3+5,2*7,'A'
DB 'Eine Zeile',12
DW 0,0C800H,1999,4711
DS 200H

Definiere Modulname: NAME <symbol>

Definiere Adresse: ORG <exp>

Benutze Registerbank: USING <exp>

Programmende: END

5. Pseudo-Befehle (Fortsetzung)

Definiere Konstante: <symbol> EQU <expr>

Definiere Variable: <symbol> SET <expr>

Beispiele: COUNTREG EQU R5
 CHAPTER SET 1
 CHAPTER SET CHAPTER+1

Definiere symb. Codeadresse: <symbol> CODE <expr>

Definiere symb. Datenadresse: <symbol> DATA <expr>

Definiere symb. ind. Adresse: <symbol> IDATA <expr>

Definiere symb. Bitadresse: <symbol> BIT <expr>

Definiere symb. ext. Adresse: <symbol> XDATA <expr>

Beispiele: EPROM CODE 08000H
 STACK DATA 7
 V24BUF IDATA 080H
 REDLED BIT P1.5
 SAMPLER XDATA 0100H

Anfang Codesegment: CSEG [AT <expr>]

Anfang Datensegment: DSEG [AT <expr>]

Anfang ind. Datensegment: ISEG [AT <expr>]

Anfang Bitsegment: BSEG [AT <expr>]

Anfang ext. Datensegment: XSEG [AT <expr>]

Beispiele: CSEG AT 8000h ;starte ein neues Codesegment bei 8000H
 XSEG at 0 ;starte ein neues ext. Datensegm. bei 0
 DSEG ;wechsle zu bereits def. Datensegment

Segmenttypen: CODE, DATA, IDATA, XDATA, BIT, NUMBER

Beispiele: OFFSET EQU 16 ;Number
 START CODE 30H ;Code
 DOIT CODE 0100H ;Code
 REDLED BIT P1.3 ;Bit
 VARIAB4 DATA 20H ;Data
 PORT DATA 0C8H ;Data
 DELAY EQU 5 ;Number

 ==> START+OFFSET+3 ;Code
 START+DOIT ;Number
 PORT.RELAY ;Bit

6. Assembler-Steuerungen

Symbol	Typ	Standard	Kurz	Beschreibung
\$COND	G	\$COND	-	IF..ENDIF komplett in Listing
\$NOCOND	G		-	IF..ENDIF nur teilweise in Listing
\$CONDONLY	G		-	nur Assemblerzeilen bei IF
\$DATE(s)	P	"	\$DA	Datum in Seitenkopf einfügen
\$DEBUG	P	\$NODEBUG	\$DB	Debug-Info in OMF-51
\$NODEBUG	P		\$NODB	ohne Debug-Info
\$EJECT	G		\$EJ	Neue Seite im Listing
\$ERROR(s)	G	-		Generiere Assmblerr-Fehlermeldung
\$WARNING(s)	G	-		Nachricht auf Konsole ausgeben
\$GEN	G	\$GEN	\$GE	vollständige Makros in Listing
\$NOGEN	G		\$NOGE	nur Makro-Aufrufe in Listing
\$GENONLY	G		\$GO	list expansion lines only
\$INCLUDE(file)	G		\$IC	Einschließen eines ext. Quelltextes
\$LIST	G	\$LIST	\$LI	Listing ab hier einschalten
\$NOLIST	G		\$NOLI	Listing ab hier ausschalten
\$MOD51	P	\$MOD51	\$MO	vordefinierte SFRs
\$NOMOD51	P		\$NOMO	vordefinierte SFRs aus
\$NOBUILTIN	P	list SFR	-	vordefinierte SFRs nicht im Listing
\$NOTABS	P	use tabs	-	Listing ohne Tabs
\$PAGING	P	\$PAGING	\$PI	Seitenformatierung im Listing
\$NOPAGING	P		\$NOPI	keine Seitenformatierung im Listing
\$PAGELENGTH(n)	P	n=64	\$PL	Seitenlänge in Zeilen im Listing
\$PAGEWIDTH(n)	P	n=132	\$PW	Seitenbreite in Zeichen im Listing
\$PHILIPS	P	MCS-51	-	83C75x Unterstützung
\$SAVE	G		\$SA	sichert \$LIST/\$GEN/\$COND Status
\$RESTORE	G		\$RS	holt \$LIST/\$GEN/\$COND Status zurück
\$SYMBOLS	P	\$SYMBOLS	\$SB	erzeugt eine Symbol-Tabelle
\$NOSYMBOLS	P		\$NOSE	erzeugt keine Symbol-Tabelle
\$TITLE(s)	G	copyright	\$TT	Titel für Seitenkopf
\$XREF	P	\$NOXREF	\$XR	Erzeuge Querverweise
\$NOXREF	P		\$NOXR	Erzeuge keine Querverweise