

# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b>	<b>1</b>
<b>2</b>	<b>Die Entwicklungsumgebung</b>	<b>5</b>
2.1	Ein einfaches Beispiel .....	7
2.2	Assembler und Linker .....	8
2.3	Der Debugger .....	9
<b>3</b>	<b>Allgemeiner Aufbau eines Assembler-Programms</b>	<b>13</b>
3.1	Bezeichner .....	14
3.2	Befehle .....	16
3.3	Direktiven .....	17
3.4	Ein Standard-Rahmen für Assembler-Programme .....	18
3.5	Befehle und Direktiven des Beispielprogramms .....	19
3.6	Makro-Aufrufe .....	22
3.6.1	Ausgabe von Text .....	22
3.6.2	Einlesen von Text .....	25
3.6.3	Einlesen und Ausgeben von Zahlen .....	26
3.6.4	Zufallszahlen-Generator .....	29
3.7	Das Assembler-Listing .....	30
3.7.1	Der erweiterte Quell-Code .....	32
3.7.2	Die Symbol-Tabelle .....	34
3.7.3	Die Cross-Referenz-Liste .....	35
3.7.4	Steuerung der Listing-Ausgabe .....	36
<b>4</b>	<b>Konstanten und Variablen</b>	<b>39</b>
4.1	Ganze Zahlen .....	39
4.2	Zeichen und Zeichenketten .....	42
4.3	Vereinbarung von Variablen .....	43
4.4	Konstante Ausdrücke .....	47
4.4.1	Arithmetische Operatoren .....	49
4.4.2	Schiebeoperatoren .....	52

4.4.3	Logische Operatoren .....	53
4.4.4	Vergleichsoperatoren .....	54
4.5	Konstanten-Vereinbarung .....	55
<b>5</b>	<b>Arithmetische Befehle</b>	<b>59</b>
5.1	Die Register .....	59
5.1.1	Die allgemeinen Register .....	59
5.1.2	Die Index-Register .....	61
5.1.3	Die Zeiger-Register .....	61
5.1.4	Die Segment-Register .....	61
5.1.5	Der Befehlszeiger .....	62
5.1.6	Das Status-Register .....	62
5.2	Zuweisung, Addition und Subtraktion .....	65
5.3	Schiebebefehle .....	70
5.3.1	Die logischen Schiebebefehle .....	71
5.3.2	Die arithmetischen Schiebebefehle .....	73
5.3.3	Rotationsbefehle .....	74
5.4	Multiplikation .....	75
5.5	Division .....	80
5.6	Logische Befehle .....	83
5.7	Keller-Befehle .....	87
<b>6</b>	<b>Kontrollstrukturen im Assembler</b>	<b>89</b>
6.1	Die einseitige Verzweigung .....	90
6.2	Die zweiseitige Verzweigung .....	99
6.3	Bedingungsschleifen .....	100
6.4	Zählschleifen .....	107
<b>7</b>	<b>Adressierungsarten</b>	<b>111</b>
7.1	Die unmittelbare und die direkte Adressierung .....	111
7.2	Die Index-Adressierung .....	113
7.3	Die indirekte Adressierung .....	121
7.4	Die basis-indizierte Adressierung .....	123
7.5	Die index-indirekte Adressierung .....	128
7.6	Die Stringadressierung .....	131
7.6.1	Stringbefehle zum Datentransport .....	133
7.6.2	Stringbefehle zum Durchsuchen .....	139
7.6.3	Stringbefehle zum Vergleich von Speicherbereichen .....	143

---

<b>8</b>	<b>Makros und bedingte Assemblierung</b>	<b>145</b>
8.1	Makro-Definition und Makro-Aufruf .....	146
8.2	Blockwiederholungen .....	157
8.3	Bedingte Assemblierung .....	162
8.3.1	Allgemeine Bedingungsblöcke .....	164
8.3.2	Makro-spezifische Bedingungsblöcke .....	167
8.3.3	Geschachtelte Bedingungsblöcke .....	171
8.4	Geschachtelte Makros .....	172
8.5	Weitere Direktiven zur Steuerung der Listing-Ausgabe .....	176
<b>9</b>	<b>Unterprogramme</b>	<b>179</b>
9.1	Definition und Aufruf von Unterprogrammen .....	180
9.2	Parameter-Übergabe .....	182
9.3	Lokale Marken und lokale Variable .....	193
9.4	Ergebnisse aus Unterprogrammen .....	197
9.5	Spezielle Direktiven .....	200
9.6	Rekursive Unterprogramme .....	208
9.7	Aufruf von Interrupt-Routinen .....	216
9.8	Selbstgeschriebene Interrupt-Routinen .....	228
<b>10</b>	<b>Segmente und Segment-Anweisungen</b>	<b>233</b>
10.1	Die physikalische Speicheradresse .....	233
10.2	Die vereinfachten Segment-Anweisungen .....	239
10.3	Die Standard-Segment-Anweisungen .....	243
10.4	Die Segmente der vereinfachten Segment-Anweisungen .....	251
10.5	Kommandozeilen-Parameter .....	254
10.6	Der Bildschirm-Speicher .....	258
<b>11</b>	<b>Modularisierung von Programmen</b>	<b>261</b>
11.1	Sprachmittel des Assemblers zur Modularisierung .....	264
11.2	Der Binder .....	269
11.3	Die Bibliotheksverwaltung .....	275

<b>12</b>	<b>Assembler und Hochsprachen</b>	<b>277</b>
12.1	Die Schnittstelle zu C++ und C .....	280
12.1.1	Die Speichermodelle .....	280
12.1.2	C++ oder C ruft ein Assembler-Unterprogramm auf .....	281
12.1.3	Assembler ruft C++-Funktion auf .....	292
12.1.4	Verwendung gemeinsamer Daten .....	294
12.1.5	Ein Beispiel mit dem Speichermodell LARGE .....	297
12.2	Die Schnittstelle zu Pascal .....	299
12.3	Die Schnittstelle zu Turbo Pascal .....	299
12.3.1	Das Speichermodell von Turbo Pascal .....	300
12.3.2	Turbo Pascal ruft ein Assembler-Unterprogramm auf .....	302
12.3.3	Assembler ruft Turbo Pascal-Funktion auf .....	308
12.3.4	Verwendung gemeinsamer Daten .....	310
12.3.5	Das Speichermodell TPASCAL .....	311
<b>13</b>	<b>Zusammengesetzte Datentypen</b>	<b>313</b>
13.1	Strukturen .....	314
13.2	Variante Strukturen .....	319
13.3	Records .....	322
<b>14</b>	<b>Andere Prozessoren</b>	<b>325</b>
14.1	Einstellung des Prozessors .....	326
14.2	Die neuen und erweiterten Befehle des 80186 und 80286 .....	327
14.3	Der 32 Bit-Prozessor 80386 .....	331
14.3.1	Segment-Typen des 80386 .....	331
14.3.2	Erweiterte Verwendung der Register bei den Adressierungsarten .....	332
14.3.3	Neue Befehle des 80386 .....	333
14.3.4	Erweiterung vorhandener Befehle .....	337
<b>Anhang</b>		<b>339</b>
A	Die verwendeten Makros .....	339
B	ASCII-Tabelle .....	348
<b>Literaturverzeichnis</b>		<b>350</b>
<b>Index</b>		<b>351</b>