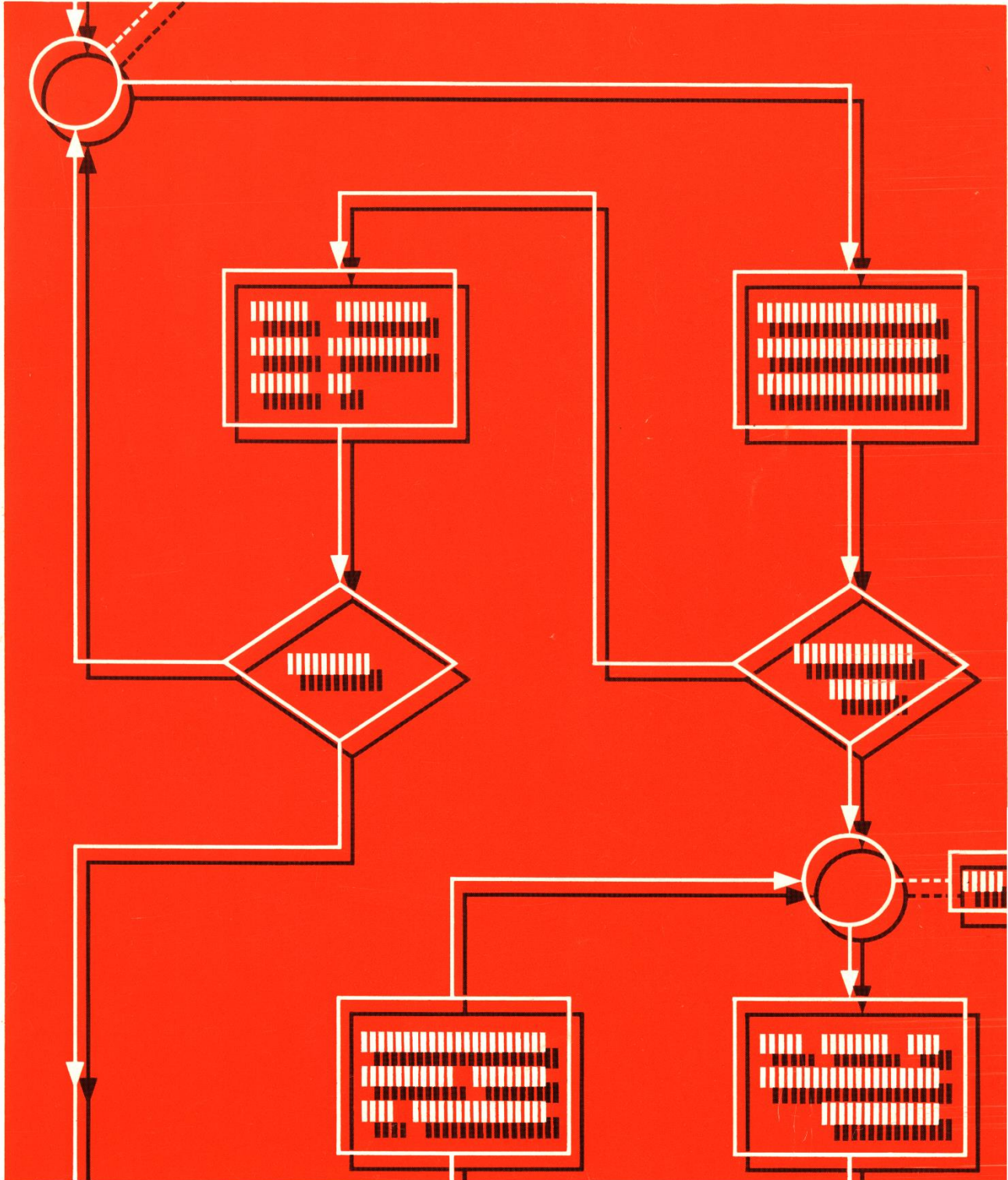


Mikroprozessor-Labor

Inhalt, Stichwörter und Betriebsprogramm



**Technisches Lehrinstitut
Dr.-Ing. P. Christiani Konstanz**



Inhalt

Hardware

Was kann ein Mikroprozessor	H 1
Der Zusammenbau des Entwicklungssystems	H 5
Das Handwerkszeug	H 5
Das Entwicklungssystem	H 5
Bestücken der Leiterplatte Tasten und Anzeige	H 7
Montage der Leiterplatte für Federleiste, 64polig	H 15
ICs auf der Baugruppe CPU	H 19
Das Netzgerät	H 21
Inbetriebnahme des Entwicklungssystems	H 22
Grundlagen der Digitaltechnik	H 27
Die Verknüpfung binärer Signale	H 27
Die Theoreme von de Morgan	H 41
Belastbarkeit digitaler Schaltglieder	H 44
Das Basisflipflop	H 47
Das JK-Flipflop	H 54
Das D-Flipflop	H 57
Monostabile Kippglieder	H 59
Der Aufbau des Mikroprozessors	H 61
Akkumulator, Extension-Register und Status-Register	H 62
Quarzuhr mit Ziffermanzeige	H 71
Der Aufbau des Mikroprozessors	H 73
Die Pointer-Register	H 73
Die Register als binäre Zahlenspeicher	H 83
Leiterplatte für Relais und LEDs	H 95
Der Zusammenbau der Leiterplatte	H 96
Der Schaltplan der Leiterplatte	H 100
Funktionsprüfung	H 103
Speicher und Adressen im Mikroprozessor-System	H 105
Die Daten im Mikroprozessor-System	H 105
Die Festspeicher	H 107
Schreib-Lese-Speicher	H 109
Das System der Speicher-Adressen	H 111
Anordnung von Speichern und Adressen in unserem Mikroprozessor-System	H 115
Schreib-Lese-Speicher und Festspeicher	H 117
Verschiedene Speicher-Prinzipien	H 117
Der Aufbau von Schreib-Lese-Speichern	H 122
Unterschiedliche Speicherzellen im Speicherfeld	H 130
Der Drucker	H 133
Der Zusammenbau des Druckers	H 134
Die Inbetriebnahme des Druckers	H 151
Der Schaltplan unseres Mikroprozessor-Systems	H 159
Tasten und Anzeige	H 159
Das Mikroprozessor-System als Schreibmaschine	H 169
Festspeicher	H 183
Masken-programmierte Festspeicher (ROM)	H 185
Anwender-programmierbare Festspeicher (PROM, REPRON)	H 189
Der Schaltplan unseres Mikroprozessor-Systems	H 193
Bitmuster-Anzeige, EIN-AUS-Anschluß und DRUCKER-Anschluß	H 193
Der Schaltplan unseres Mikroprozessor-Systems	H 203
Das Programm-Entwicklungssystem	H 203
Der Mikrocomputer	H 206
Die CPU unseres Mikroprozessor-Systems	H 213
Die Hardware-Anschlüsse der CPU	H 213
Adressen auf dem Datenbus	H 215
Status-Flags auf dem Datenbus	H 218
Zusammenschaltung mehrerer Mikroprozessoren	H 219

Software

Der Programmablaufplan	S 1
Anweisungen	S 2
Verzweigung und Schleifen	S 3
Unterprogramme	S 5
Merker (Flags)	S 7
Programmunterbrechungen (Interrupts)	S 9
Der Programmablauf beim Mikroprozessor	S 11
Die Nachbildung von Hardware durch Software	S 13
Die Verknüpfung von zwei definierten Signalwerten	S 13
Die Verknüpfung beliebiger Signalwerte	S 18
Ein Universal-Verknüpfungsprogramm	S 20
Die Darstellung von Zahlen durch Zeichen	S 25
Das Sedezimalsystem	S 25
Rechnen mit sedezimal dargestellten Zahlen	S 33
Addieren im Sedezimalsystem	S 33
Subtrahieren mit Sedezimalsystem	S 37
Sedezimale Addition und Subtraktion mit dem Mikroprozessor	S 41
Die Darstellung von Zahlen durch Zeichen	S 45
Das Dualsystem	S 45
Rechnen mit dual dargestellten Zahlen	S 49
Addieren im Dualsystem	S 49
Subtrahieren im Dualsystem	S 50
Das Zusammenwirken von CPU und Speicher im Mikroprozessor-System	S 57
Laden und Speichern	S 57
Ein Programm für Daten im Speicher	S 58
Der Befehlssatz unseres Mikroprozessors	S 65
Sprungbefehle	S 65
PC-relative Sprungbefehle	S 66
Die Berechnung des Displacements	S 69
Pointer-indizierte Sprungbefehle	S 75
Elektronische Orgel	S 83
Tongenerator mit Additions-Befehl	S 83
Das Orgel-Programm	S 86
Der Befehlssatz unseres Mikroprozessors	S 91
Bedingte Sprungbefehle	S 91
PC-relativer Sprung bei positivem Akkumulator-Inhalt	S 93
PC-relativer Sprung bei Akkumulator-Inhalt Null	S 98
PC-relativer Sprung bei Akkumulator-Inhalt ungleich Null	S 107
Pointer-indizierte Befehle für bedingte Sprünge	S 111
Mikrozyklen als Maß für die Befehlsausführungszeit	S 117
Der Verzögerungsbefehl	S 120
Die Programmierung langer Verzögerungen	S 124
Tongeneratoren	S 130
Der Befehlssatz unseres Mikroprozessors	S 141
Der Lade- und der Speicher-Befehl	S 141
Befehle mit autoindizierter Adressierung	S 152
Das Verschieben eines Programmblocks	S 158
Die Verknüpfungsbefehle	S 167
UND-Verknüpfungsbefehle	S 171
ODER-Verknüpfungsbefehle	S 177
Exklusiv-ODER-Befehle	S 178
Programmunterbrechungen (Interrupt)	S 187
Die Bearbeitung eines Interrupts	S 187
Interrupt-Einrichtung im Mikroprogramm	S 190
Der Befehlssatz unseres Mikroprozessors	S 193
Die Interrupt-Befehle	S 194
Ein einfaches Würfelspiel	S 199
Komplexe Interrupt-Systeme	S 205

Der Befehlssatz unseres Mikroprozessors	S 209
Wie ein Mikrocomputer rechnet	S 209
Die Additionsbefehle und das Übertrags-Bit	S 211
Die Befehle für dezimale Addition	S 218
Subtrahieren mit dem Mikroprozessor	S 221
Die Subtraktion dezimal dargestellter Zahlen	S 230
Das Status-Bit OV	S 235
Der Befehlssatz unseres Mikroprozessors	S 239
Die Schiebe- und Rotationsbefehle	S 239
Der Befehl SIO für den seriellen Eingang und Ausgang	S 241
Software-Hilfen	S 250
Schlußwort	S 254

Übungen

Hinweise zum Anfertigen und Einsenden der Lösungen ..	Ü 1
Prüfungsaufgaben und Lösungen der im Text gestellten Aufgaben	Ü 3

Tabellen

Anschlüsse des ISP-8A/600-Mikroprozessors (SC/MPII) ..	T 1
SC/MP-Architektur	T 2
Schaltplan des Druckers	T 3
7-Bit-Code ASC II	T 4
SC/MP-Befehlsliste	T 5
Alphabetische Befehlsliste	T 8
Programm: Verschieben eines Datenblocks	T 9
Betriebsprogramm: Eingabe	T 10
Adressen-Plan, Scratchpad, aufrufbare Unterprogramme ..	T 11

Kartei

.....	K 1
-------	-----

Stichwörter

A

Abfragen der Peripherie S 104
 Abfrageprogramm H 166
 Abschlußzeugnis Ü 2
 AC H 63
 Acht-Bit-Mikroprozessor H 86, H 111
 Acht-Bit-Register H 87, H 105
 Acht-Bit-Wort H 87, H 106
 Active high H 126, H 214
 Active low H 126, H 207, H 214
 Addieren|im Dualsystem S 49
 – im Dezimalsystem S 33
 Addition von sedezimal dargestellten Zahlen H 88
 – , dezimal S 218
 – , sedezimal mit Mikroprozessor S 41
 Additions|befehle S 211
 – programm, Gebrauchsanweisung S 42
 Adreß|-Bereich H 115
 – -Byte H 123
 Adreßbus H 114, H 159
 Adresse H 112, H 115, S 101
 – , effektive S 58
 Adressen|auf Datenbus H 215
 – -Belegung H 161
 – -Decodierer H 187
 – im Mikroprozessor-System H 105
 Adressierung, autoindiziert S 152
 Äquivalenz-Verknüpfung H 39
 Akkumulator H 63, H 93
 Alphanumerische Zeichen H 157
 American Standard Code for Information Interchange H 157
 Ampel S 128
 Anschlüsse der SC/MP-CPU H 213
 Antivalent H 36
 Anweisung S 2
 Anwender|-Programm H 183
 – -programmierbare Festspeicher H 189
 Anzeige H 159
 Architektur H 61, H 213
 – , SC/MP T 2
 ASCII H 157
 – -Byte H 177
 – -String H 157
 Assembler S 162, S 251
 – -Läufe S 252
 Assemblierer S 162, S 251
 Auffangflipflop H 57
 Ausdruck, Register H 152
 Ausgang, seriell S 241
 Aussortieren S 104
 Austausch-Befehl H 74
 Autoindizierte Adressierung S 152
 Autoindizierter Befehl S 152

B

Basisflipflop H 47
 – aus NAND-Gliedern H 50
 Bauelement, digitales H 2
 Baugruppe CPU H 19
 Bausteine, periphere H 116

Bedingter Sprung S 87, S 91, S 115
 Befehl H 4
 – , Additions- S 211
 – , Austausch- H 74
 – , autoindizierter S 152
 – , Ein-Byte- H 88
 – , Exklusiv-ODER- S 178
 – , für seriellen Eingang und Ausgang S 241
 – , Interrupt- S 194
 – , Kopier- H 74, S 58
 – , Lade- S 141
 – , Makro- S 253
 – , ODER- S 177
 – , Rotations- S 239
 – , Schiebe- S 239
 – , Speicher- S 146
 – , UND- S 171
 – , Verknüpfungs- S 167
 Befehls|liste T 5
 – nummer H 73
 – satz H 63, S 13, S 51
 Belastbarkeit|der CPU-Anschlüsse H 204
 – digitaler Schaltglieder H 44
 Betriebsanweisung Mikroprozessor-System S 24
 Betriebsprogramm S 71, H 108, H 183
 Betriebssystem H 108
 Bidirektional H 128, H 159, H 207
 Binär H 83
 – -Zeichen H 45, H 83, H 85, H 122
 Binäre Signale, Verknüpfung H 27
 – Signalwerte H 44
 Binary digit H 85
 Bipolare Technik H 130, H 132
 Bipolares ROM H 186
 Bit H 85
 – -parallel S 241
 – -seriell S 241
 – , höchstwertiges H 87
 – , niedrigstwertiges H 87
 – muster H 85
 – muster-Anzeige H 193
 – organisierter Speicher H 124
 Blinkschaltung H 92
 Break Point S 174
 Brennen H 190
 Buchstaben H 133
 Buffer H 204
 Bus H 114, H 122
 – -Treiber H 206
 Byte H 87
 – , Adreß- H 123

C

Central Processing Unit H 2
 Chip H 126, H 187
 Chip enable H 126, H 188
 – fläche H 191
 CMOS H 130
 Code|, Mnemonischer H 64, S 101
 – , 1- aus 16 H 123
 Codierung S 25
 Compiler S 254
 Compute S 209

CONT H 214
 Count Down S 100
 Counter S 99
 CPU H 2, H 61
 – -Anschlüsse, Belastbarkeit H 204
 Cross-Assembler S 252
 CY/L H 86, S 213

D

D-Auffangflipflop H 57
 D-Flag H 218
 D-Flipflop H 57, H 161, H 194, H 204
 Darstellung von Zahlen S 25
 Daten H 106, H 112
 – im Mikroprozessor-System H 105
 – , Eingabe S 193
 Datenbus H 114, H 159
 Daten|verarbeitungs-System H 106
 – wort H 186
 De Morgansche Theoreme H 41
 Debug S 251
 – -Routinen S 251
 Decodierer H 124, H 161, H 209
 – , Adressen- H 187
 Dekrementieren S 99
 Delay H 92, S 120
 Dezimale Addition S 218
 – Subtraktion S 230
 Dezimalsystem S 25
 Differenz S 38
 Diffusion H 185
 Digitales Bauelement H 2
 – Schaltwerk H 3
 Digitaltechnik, Grundlagen H 27
 Dioden-Matrix H 185
 Disjunktion H 32
 Displacement S 61, S 68, S 101, S 111, S 143
 – , Berechnung S 69
 – im Extension-Register S 88
 – 80 S 146, S 149, S 173
 Dividieren S 209
 Don't-care H 127
 Drucker H 133
 – -Anschluß H 193
 – -Metallpapier H 145
 – , Inbetriebnahme H 151
 – , Zusammenbau H 134
 Dual-in-line-Gehäuse H 2
 Dualsystem S 45
 Durchkontaktierte Leiterplatte H 7
 Dynamischer Speicher H 131

E

ECL-Technik H 132
 Effektive Adresse S 58, S 68
 Ein|-Aus-Anschluß H 193
 – -Byte-Befehl H 88
 – -Chip-Mikroprozessor H 3, H 106
 H 203
 – -Lauf-Assemblierer S 252

Einer|-Komplement-Addition S 225
 – komplement S 224
 Eingabe von Daten S 193
 Eingang, seriell S 241
 Eins-zu-eins-Sprachen S 253
 Einweg-Bus-Treiber H 206
 Einzelschritt H 66, H 154, S 63
 Empfänger H 44
 – -Schaltglied H 45
 Entscheidung S 91
 Entwicklungssystem H 5
 EPROM H 184
 Erasable Programmable Read-Only-Memory H 184
 Ergänzungszahl S 53
 EX H 63
 Exklusiv|-ODER-Befehle S 178
 – -Verknüpfung H 37, H 167, S 179
 Extension-Register H 63

F

Fachwortkartei K 6
 Fehlerhafte Eingaben, Korrektur S 16
 Fernschreiber S 162
 Festspeicher H 107, H 183
 – A H 115
 – B H 115
 – , programmierbar H 107
 Flag S 7, S 8, H 100
 – D H 218
 – H H 219
 – I H 218
 – R H 218
 – -Signal S 83
 Flipflop H 48, H 51, H 122
 – , antivalenter Ausgang H 53
 – , Flankensteuerung H 55
 – , JK H 54
 – , RS H 52
 – , Schaltzeichen H 52
 – , Taktzustandssteuerung H 56
 Floppy Disk H 119, H 132
 Flowchart S 1
 Flußdiagramm S 1
 Frequenz S 85
 Funktionstabelle H 29
 Fußgänger-Ampel S 196

G

Gebrauchsanweisung | für Additionsprogramm S 42
 – für Subtraktionsprogramm S 43
 Gleichwertiges Schaltzeichen H 46
 Grenzstelle S 4
 Größer-kleiner-Vergleich S 95

H

H-Bereich H 44, H 85, H 87
 – -Flag H 219

Halbleiter-Speicher H 121
 HALT H 219
 – -Stelle S 174
 Hardware H 1
 Hardwired Logic H 4
 Hauptprogramm S 187
 Hexadezimalsystem S 26
 HIGH H 27
 HIGH-Teil H 75
 Higher Order Byte H 90
 Hintergrundprogramm S 194
 HOB H 90
 Höchstwertiges Bit H 87, S 93

I

I-Flag H 218
 IE H 86, S 190
 Inbetriebnahme des Entwicklungssystems H 22
 Information H 27
 Inkrementieren S 99
 Intelligente Schaltung S 91
 Intelligenz S 91
 Interface H 205
 International Organization for Standardization H 157
 Interpretierer S 253
 Interrupt S 9, S 187
 – im Mikroprogramm S 190
 – Request S 188
 – Aufruf S 190
 – Befehle S 194
 – Hierarchie S 10
 – Routine S 10, S 188
 – Schachtelung S 11, S 192, S 205
 – Systeme, komplexe S 205
 Invertieren S 21
 Irregulär H 52
 ISO H 157
 ISP-8A/600 T 1
 I²L H 130

J

JK-Flipflop H 54
 – , Schaltzeichen H 55

K

KByte H 120, H 132
 Kernspeicher H 118, H 131
 Kilobyte H 120
 Kippglied, Monostabil H 59
 Klammeraffe S 155
 Klartext H 133
 Kompilierer S 253
 Komplement-Bildung S 55
 Komplementierung S 53
 Konjunktion H 30
 Kopier-Befehl S 58, H 74

Kopieren H 67
 Korrekturen fehlerhafter Eingaben S 16

L

L-Bereich H 44, H 85, H 87
 Label S 96, S 116, S 162
 Ladebefehl S 141
 – , autoindiziert S 152
 – , PC-relativ S 144
 – , Pointer-indiziert S 144
 Laden S 57, H 63
 Latch H 57, H 194
 – -Flipflop H 57
 Latchen H 194, H 217
 Least Significant Bit H 87
 LED H 96
 Leerzeichen H 170
 Leiterplatte | für Federleiste, 64polig H 15
 – für Relais und LEDs H 95
 – – , Schaltplan H 101
 – für Tasten und Anzeige H 7
 – , durchkontaktierte H 7
 Lese-Zyklus H 168, H 207
 Line-Feed H 157, H 170
 LOB H 90
 Löscheingang H 52
 Löschen von REPROM H 191
 Logik, negative H 27
 – , positive H 27
 Logisch S 91
 LOW H 27
 – -Teil H 75
 Lower Order Byte H 90
 LSB H 87

M

Makrobefehl S 253
 Maschinensprache S 174, S 250
 Maske H 184
 Masken|-Byte S 175
 – programmierter Festspeicher H 185
 – programmiertes ROM H 189
 Maskieren H 167, S 175
 Materialkartei K 1
 Merker S 7
 Metallpapier H 145
 Mikrocomputer H 3, H 206
 Mikroprogramm H 113, S 120, H 168
 Mikroprozessor H 4
 – , Acht-Bit- H 86, H 111
 – , Ein-Chip- H 3, H 106, H 203
 – , ISP-8A/600 T 1
 – , SC/MP H 213
 – , Sechzehn-Bit- H 86, H 111
 – , Vier-Bit- H 86, H 186
 Mikrozyklus S 117
 Minimal-System H 203
 Minuend S 38
 Mnemonischer Code H 64, S 101
 Monitor H 108
 Monostabiles Kippglied H 59
 MOS-ROM H 187
 Most Significant Bit H 87

MSB H 87, S 93
Multiplizieren H 175, S 209
Multiprocessing H 220

N

NADs H 218
NAND|-Glied H 2
– -Verknüpfung H 2, S 19, H 34
NBREQ H 215
Negation H 34
– mit Exklusiv-ODER S 184
Negative Logik H 27
NENIN H 214
NENOUT H 214
Netzgerät H 21
NHOLD H 215
NICHT-Glied H 36
Niedrigstwertiges Bit H 87
NOR-Verknüpfung S 23, H 32
NRST H 214
NUR-Lese-Speicher H 184

O

Objektprogramm S 250
ODER-Verknüpfung H 31, S 167
ODER-Verknüpfungsbefehle S 177
Oktal H 115
– , System S 32
One Shot H 60
Opcode S 61
Operation S 4
Operationscode H 64, S 61, S 101, S 111
– , Variation S 58
Orgel S 86
Output Disable H 218
OV-Bit H 86, S 235
Overflow S 237

P

Parallel|-Serie-Umwandlung S 243
– -Übertragung S 241
PC H 74
– -relativ S 68
– -relativer Sprung S 70
Periodendauer S 119
Periphere Bausteine S 116
Peripherie H 86, H 195, S 193, H 204
Prüfung auf Ungleichheit S 183
Prüfungsaufgaben Ü 1
Pointer H 73
– -indizierter, bedingter Sprung S 111
– -Register H 73
– -Register PC H 113
– -Register P0 S 77
Polling H 166
Port H 195, H 203
Porto Ü 1
Positive Logik H 27

Programm H 2, S 11, H 133
– , Abfrage- H 166
– -Ablaufplan S 1, S 11
– -Anwender- H 183
– -Block verschieben S 158
– -Baustein S 6
– -Counter H 74, H 113
– -Daten H 113
– -Entwicklungssystem H 5, H 95, H 108
– H 183, H 203
– -Halt S 174
– , lauffähiges H 93
– -Schleife S 5
– -Sprünge H 79
– -Unterbrechung S 9, S 187
– -Verzweigung S 5
– -Zähler H 74, H 113
Programmable Read Only Memory H 107, H 184
Programmierbarer Festspeicher H 107
Programmieren H 189
Programmier|gerät H 184
– -sprache, höhere S 253
PROM H 107, H 184, H 189
– -Bipolar H 189
– -Programmiergerät H 180
Prozedur S 6
– anweisung S 6
Prozeßsteuerung H 95
Prüfung auf Gleichheit S 183
Puffer H 204
P0-indiziert S 111

Q

Quarz-Frequenz H 72
Quarzoszillator H 168, H 212
Quarzuhr H 71
Quellprogramm S 251
Quindezimalsystem S 29

R

R-Flag H 218
RAM H 109, H 121
Random Acces Memory H 109
Read Only Memory H 107, H 184
Rechenzeit S 210
Rechnen S 209
– mit Dual dargestellten Zahlen S 49
Register H 62, H 133
– , Acht-Bit H 105
– als binäre Zahlenspeicher H 83
– -Ausdruck H 152
– -Inhalt H 86
– , Sechzehn-Bit H 105
Relais-Steuerung H 196
Reprogrammable Read Only Memory H 107, H 184
REPROM H 108, H 184, H 190
– , löschen H 191
RESET H 214
Resident-Assembler S 252
Retten H 75, S 188

ROM H 107, H 184
– , Bipolares H 186
– , maskenprogrammiert H 189
– , MOS- H 187
Rotationsbefehle S 239
Routine S 6, H 115
RS-Flipflop H 52
Rucksack S 226
Rückführung H 47
Rückwärts-Sprung S 73, S 82

S

SA H 86
SB H 86
SC/MP|-Anschlüsse T 1
– -Architektur T 2
– -II T 1
Schaltglieder, Belastung H 44
Schaltnetz H 47, H 123
Schaltplan H 193
Schaltung, intelligente S 91
Schaltwerk H 47
– , digitales H 3
Schaltzeichen, gleichwertiges H 46
Schiebe-Befehle S 239
Schleife S 3
– , Warte- S 90, S 103, S 116
Schnittstelle H 205
Schreib-Lese-Speicher H 107, H 109, H 117
– , Aufbau- H 122
Schreib-Zyklus H 168, H 207
Schreibmaschine H 169
Scratchpad H 112, H 115
Sechsfach-D-Flipflop H 164
Sechzehn-Bit-Mikroprozessor H 86, H 111
– -Register H 105
Sedezimal|dargestellte Zahlen, Addition H 88
– system S 25
Sedezimale|Addition mit Mikroprozessor S 41
– Subtraktion S 94
Segmentanschlüsse H 162
Sender H 44
– -Schaltglied H 45
SENSE|A H 86
– B H 86
Serieller|Ausgang S 241
– Eingang S 241
Setzeingang H 52
Sieben-Segment-Anzeige S 148, H 161
Signal H 2, H 45
– , binär, Verknüpfung H 27
– Parameter H 27, H 44
– wert H 44
SIN S 241
Single Step H 66
Software S 1, H 4
– -Hilfen S 250
– -Zähler S 100
Source-Programm S 251
SOUT S 241
Speicher H 115
– -Adressen H 111
– -Bausteine H 117
– -Befehl S 146
– -Bereich H 133

– ,bitorganisiert H 124
 – ,dynamischer H 131
 – -Element H 111
 – ,Fest- H 107
 – glied H 51
 – ,Halbleiter H 121
 – im Mikroprozessor-System H 105
 – inhalt, Ausdruck H 156
 – matrix H 187
 – platz H 111
 – ,Schreib-Lese- H 117
 – ,Stapel- S 153
 – ,statisch H 131
 – ,wortorganisiert H 111, H 124
 – zelle H 111
 Sprung, bedingter S 87, S 115
 – bedingung S 93
 – befehle S 65
 – befehl, bedingt S 91
 – befehl, PC-relativ S 66
 – befehl, Pointer-indiziert S 75
 – , PC-relativ S 74, S 93, S 98, S 107
 – , Pointer-indiziert, bedingt S 111
 – , rückwärts S 73, S 82
 – , unbedingt S 80
 – , vorwärts S 73, S 82
 – weite S 111
 Stack S 153
 – pointer S 155
 Stapelspeicher S 153
 Start-Adresse S 70
 Statische Speicher H 131
 Status|-Bit OV S 235
 – Flags auf Datenbus H 218
 Status-Register H 66, H 198
 Stellenwertsystem S 27, S 46
 Steuerbus H 114, H 208
 String H 157, S 247
 – , ASCII H 157
 Subroutine S 6
 Subtrahend S 38
 Subtrahieren S 221
 – durch Komplement-Addition S 52
 – im Dualsystem S 50
 – im Sedezimalsystem S 37
 – , dezimal S 230
 – , sedezimal S 94
 – mit Mikroprozessor S 41
 Subtraktions|befehl S 51
 – programm, Gebrauchsanweisung S 43

T

Takt|-Wechselspannung S 119
 – eingang H 55
 – frequenz S 117
 – generator H 203, H 212
 Taschenrechner H 3
 Tasten H 159
 – 0802 und 0803 H 167
 – -Byte S 87
 – , Ziffern- H 105
 – feld H 160, H 164
 Theoreme von de Morgan H 41
 Tongenerator S 83, H 103, S 130
 Treiberschaltungen H 203

Tri-State H 215
 TTL H 3, H 44, H 130

U

Übertrag S 33, S 49, S 213
 Übertrags-Bit S 211
 Ultraviolettes Licht H 191
 Unbedingter Sprung S 80
 UND-Verknüpfung S 20, H 28, S 167
 UND-Verknüpfungs-Befehle S 171
 Universal-Verknüpfungs-Programm S 20
 Unterbrechungs-Anforderung S 188
 Unterprogramm S 5, S 109
 – -Schachtelung S 198
 – -Technik H 81
 UV-Licht H 191

V

Variable H 29
 Vergleich, größer-kleiner S 95
 Verkehrsampel S 128
 Verknüpfung, Exklusiv-ODER S 167
 Verknüpfung H 2
 – , Äquivalenz H 39
 – , binärer Signale H 27
 – , Exklusiv-ODER H 37
 – , NAND H 34
 – , NOR H 32
 – , ODER H 31, S 167
 – , UND H 28, S 167
 Verknüpfungsbefehle S 167
 – , Exklusiv-ODER S 178
 – , ODER- S 177
 – , UND S 171
 Verschieben eines Datenblocks T 9
 Verschieben von Programm-Block S 158
 Verweilzeit H 59
 Verzögerung H 92
 – , Feineinstellung S 85
 – , Länge H 92, S 124
 Verzögerungs|befehl S 120
 – Zeit S 121
 – – , kürzeste S 122
 – – , längste S 122
 Verzweigung S 3
 Vier-Bit-Mikroprozessor H 86, H 186
 Vierfach-NAND-IC H 1
 Vorwärts-Sprung S 73, S 82
 Vorzeichen-Bit S 73

W

Wahrheitstabelle H 29
 Warteschleife S 90, S 103, S 116
 Wirtschaftlichkeit H 4
 Wort H 87
 – , Acht-Bit- H 106
 Wortorganisierter Speicher H 111, H 124
 Würfelspiel S 199, S 207

Z

Zählen S 25, S 83
 Zähler S 99, S 116
 – mit Maximalwert S 181
 Zahl S 25
 – , Darstellung S 45
 Zahlendarstellung S 33
 Zehnerkomplement S 52
 Zeichen|folge S 46
 – , alphanumerisch H 156
 – , binär H 45, H 83, H 85, H 122
 – kombination S 46
 – vorrat S 26
 Zeiger H 73
 Zeilen|drucker H 134, H 170
 – vorschub H 170
 Zentraleinheit H 2
 Ziel-Adresse S 68
 Ziffer S 25
 Ziffern-Tasten H 105
 Zugriff S 57
 Zugriffszeit H 120, H 187
 Zwei-Byte-Befehl H 88
 Zweierkomplement S 56, S 232
 Zweig S 4
 Zweiweg-Bus-Treiber H 162

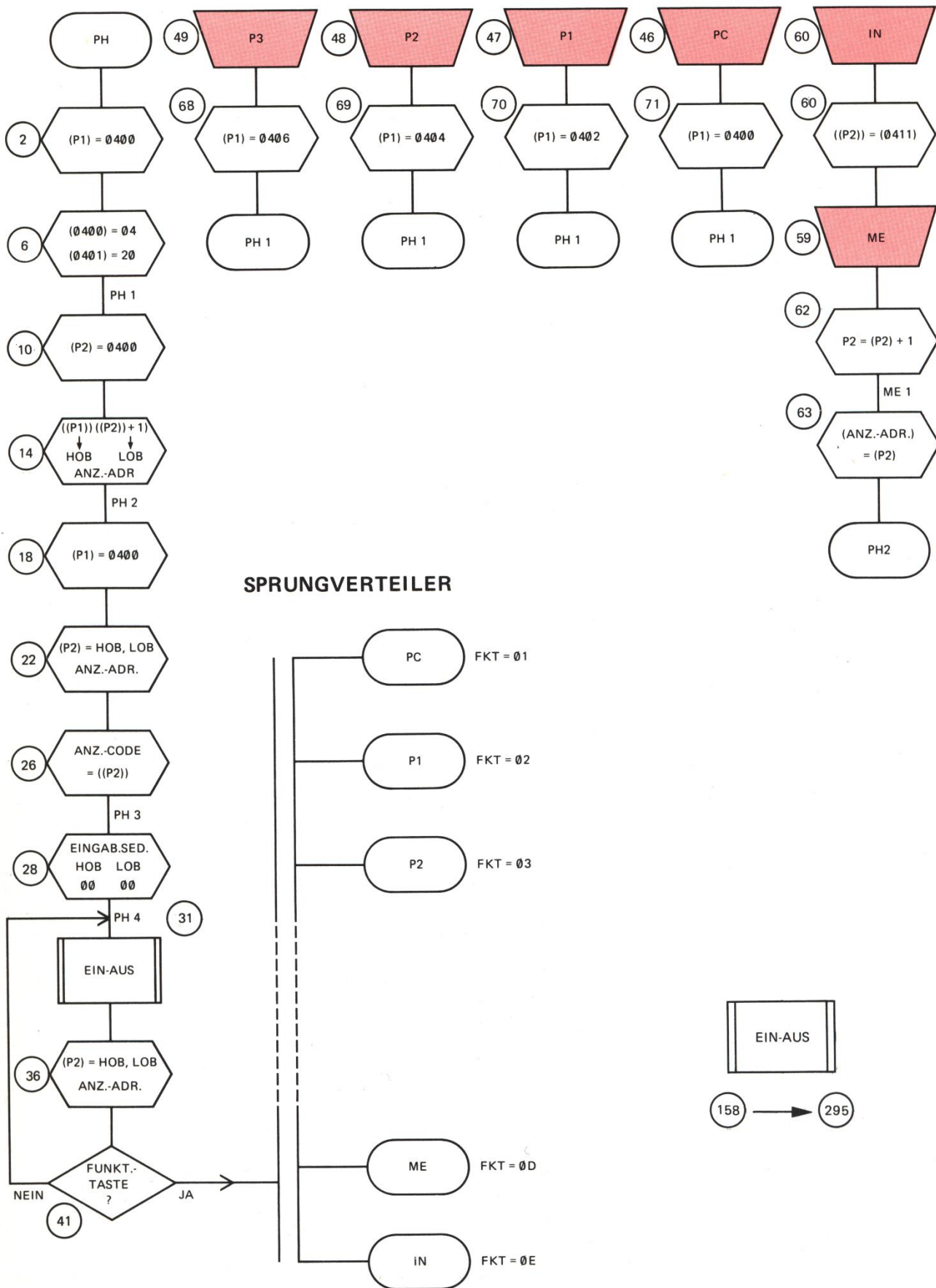
1

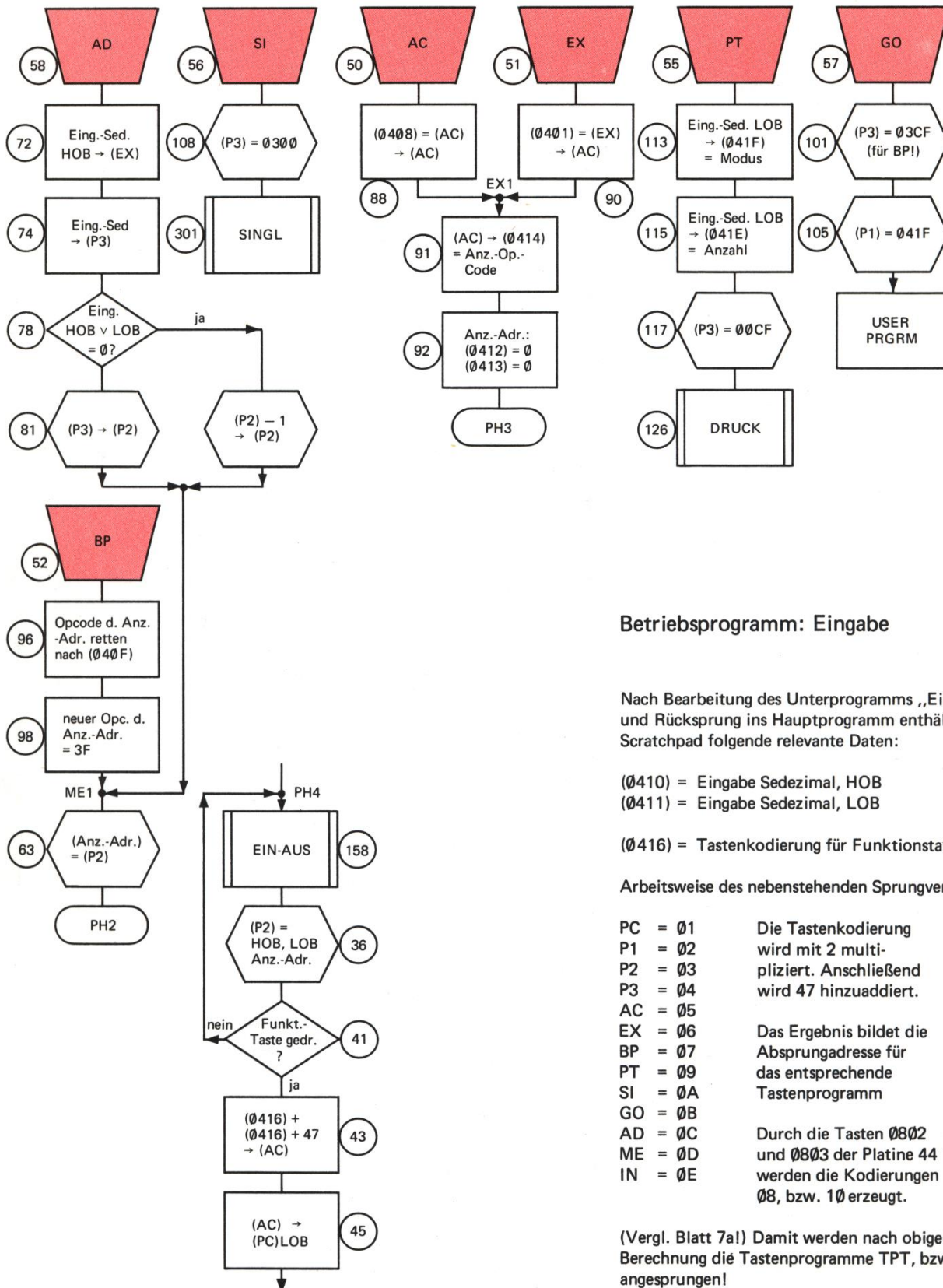
1-Aus-16-Code H 123



Blatt 0

[illegible]





Betriebsprogramm: Eingabe

Nach Bearbeitung des Unterprogramms „Ein-Aus“ und Rücksprung ins Hauptprogramm enthält das Scratchpad folgende relevante Daten:

(0410) = Eingabe Sedezimal, HOB
(0411) = Eingabe Sedezimal, LOB

(0416) = Tastenkodierung für Funktionstaste

Arbeitsweise des nebenstehenden Sprungverteilers:

PC = 01	Die Tastenkodierung
P1 = 02	wird mit 2 multi-
P2 = 03	pliziert. Anschließend
P3 = 04	wird 47 hinzuaddiert.
AC = 05	
EX = 06	Das Ergebnis bildet die
BP = 07	Ab sprungadresse für
PT = 09	das entsprechende
SI = 0A	Tastenprogramm
GO = 0B	
AD = 0C	Durch die Tasten 0802
ME = 0D	und 0803 der Platine 44 126
IN = 0E	werden die Kodierungen
	08, bzw. 10 erzeugt.

(Vergl. Blatt 7a!) Damit werden nach obiger Berechnung die Tastenprogramme TPT, bzw. ME angesprochen!

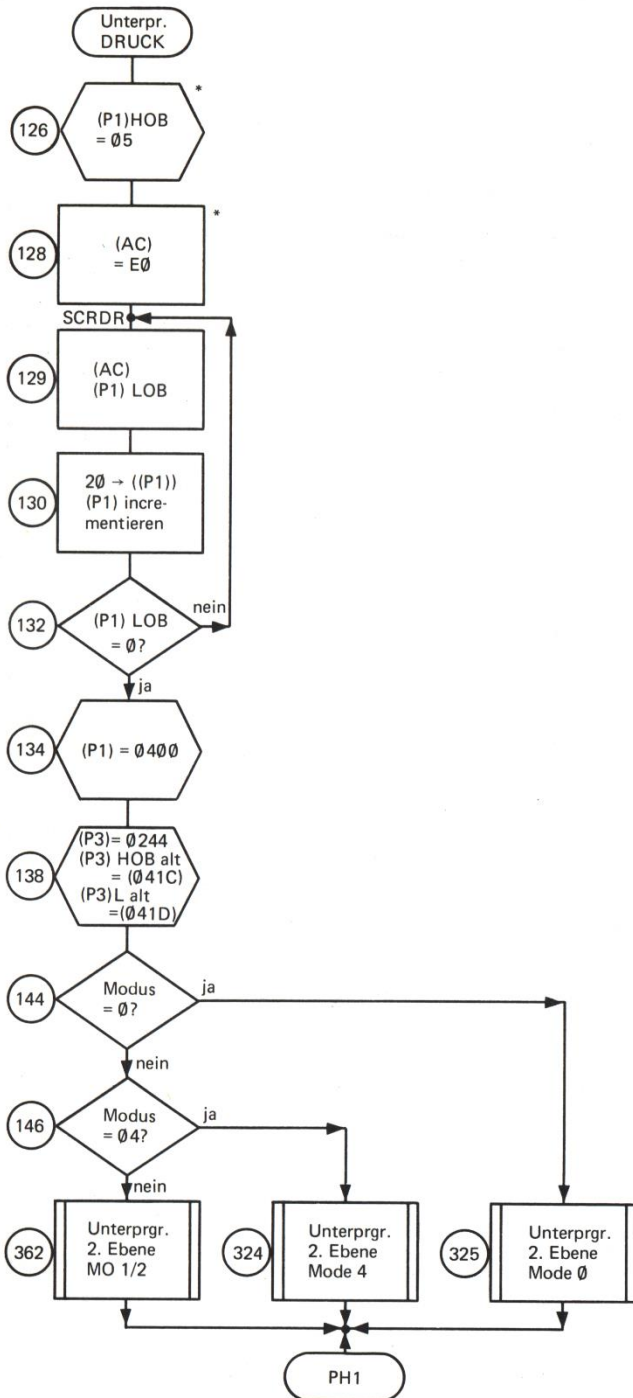


Programm: Betriebsprogramm, –
Eingabe

Blatt 2

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen	
46	TPC	JMP	PC ↓	004A	90	2C	FKT = 01 (Taste PC)	Sprung-Verteiler
47	TP1	JMP	P1 ↓	C	90	28	FKT = 02 (Taste P1)	
48	TP2	JMP	P2 ↓	E	90	24	FKT = 03 (Taste P2)	
49	TP3	JMP	P3 ↓	0050	90	20	FKT = 04 (Taste P3)	
50	TAC	JMP	AC ↓	2	90	3D	FKT = 05 (Taste AC)	
51	TEX	JMP	EX ↓	4	90	3F	FKT = 06 (Taste EX)	
52	TBP	JMP	BP ↓	6	90	49	FKT = 07 (Taste BP)	
53				8	00		(Anm.: Taste 0802, -Platine 44 126, -bewirkt Fortsetzung bei 005A = PT!)	
54				9	00			
55	TPT	JMP	PT ↓	A	90	60	FKT = 09 (Taste PT)	
56	TSI	JMP	SI ↓	C	90	57	FKT = 0A (Taste SI)	
57	TGO	JMP	GO ↓	E	90	4B	FKT = 0B (Taste GO)	
58	TAD	JMP	AD ↓	0060	90	18	FKZ = 0C (Taste AD)	
59	TME	JMP	ME ↓	2	90	04	FKT = 0D (Taste ME)	
60	IN	LD	11 (P1)	4	C1	11	FKT = 0E (Taste IN) →	
61		ST	00 (P2)	6	CA	00	((P2)) = (0411)	
62	ME	LD	@ 01 (P2)	8	C6	01	(P2) + 1 → (P2)	
63	ME1	XPAH 2		A	36		(Anzeige-Adresse) = (P2)	
64		ST	12 (P1)	B	C9	12		
65		XPAL 2		D	32			
66		ST	13 (P1)	E	C9	13		
67		JMP	PH2 ↑	0070	90	AB	Rücksprung nach 18	
68	P3	LD	@ 02 (P1)	2	C5	02	(P1) = 0406	
69	P2	LD	@ 02 (P1)	4	C5	02	(P1) = 0404	
70	P1	LD	@ 02 (P1)	6	C5	02	(P1) = 0402	
71	PC	JMP	PH1 ↑	8	90	95	(P1) = 0400 (Rückspr. nach 10)	
72	AD	LD	10 (P1)	A	C1	10	Eingabe Sed. HOB = (EX)	
73		XAE		C	01		(P3) = Eingabe Sed. HOB, LOB	
74		LDE		D	40			
75		XPAH 3		E	37			
76		LD	11 (P1)	F	C1	11		
77		XPAL 3		0081	33			
78		LD	11 (P1)	2	C1	11	Eing. Sed. HOB V LOB = 0 ?	
79		ORE		4	58		ja: Sprung nach 86	
80		JZ	INCRE ↓	5	98	06		
81		XPAH 3		7	37			
82		XPAH 2		8	36			
83		XPAL 3		9	33			
84		XPAL 2		A	32		(P3) → (P2) = Anzeige-Adresse	
85		JMP	ME1 ↑	B	90	DD	Rücksprung nach 63	
86	INCRE	LD	@ FF (P2)	D	C6	FF	P2 = (P2) – 1	
87		JMP	ME1 ↑	F	90	D9	Rücksprung nach 63	
88	AC	LD	08 (P1)	0091	C1	08	(0408) laden, –	
89		JMP	EX1 ↓	3	90	02	Sprung nach 91	
90	EX	LD	0A (P1)	5	C1	0A	(040A) laden	
91	EX1	ST	14 (P1)	7	C9	14	speichern in (0414)	
							(Anm.: Taste 0803, – Platine 44 126, –bewirkt Sprung nach 0068 ≙ ME!)	

[illegible]



Betriebsprogramm: Unterprogramm „Drucken“ (Unterprogramm in der 1. Unterprogramm-Ebene)

Relevante Parameter, — unmittelbar nach dem Ansprung des Unterprogramms:

(P2) = Anzeige-Adresse (22)

(P3) = Rücksprung-Adresse = 00CA (121)

(041F) = Modus (Druckart-Kode) (114)

(041E) = Anzahl der Befehle, Bytes, Zeichen (116)

* Die Anfangsadresse des Drucker-Scratchpads ist 05E0

129 → 132 Das Drucker-Scratchpad 05E0 bis 05FF wird mit „Space“ (Blanks) geladen

138 Die alte Rücksprung-Adresse (S.O.!) wird „gerettet“:

(P3) HOB alt = (041C)

(P3) LOB alt = (041D)

Programm: Betriebsprogramm, –
 UP „Drucken“

Blatt 4

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen
123				00CD	53		Signum der Software-Firma (SOB)
124				E	4F		
125				F	42		
126	DRUCK	LDI	05	00D0	C4	05	(P1) = 05E0 (Scratchpad Drucker)
127		XPAH 1		2	35		
128		LDI	E0	3	C4	E0	
129	SCRDR	XPAL 1		5	31		(05E0) = 20 (Space, - Blank) P1 = (P1) + 1
130		LDI	20	6	C4	20	(P1) LOB = 00 ? nein: → Rücksprung nach 129
131		ST	@ 01 (P1)	8	CD	01	
132		XPAL 1		A	31		
133		JNZ	SCRDR ↑	B	9C	F8	
134		LDI	04	D	C4	04	(P1) = 0400
135		XPAH 1		F	35		
136		LDI	00	00E0	C4	00	
137		XPAL 1		2	31		(P3) = 0244 (041C) = (P3) HOB alt = 00 (041D) = (P3) LOB alt = CA [(P3) alt = Rückspr.-Adr. zum Betriebsprogr. Eingabe!]
138		LDI	02	3	C4	02	
139		XPAH 3		5	37		
140		ST	1C (P1)	6	C9	1C	(Modus) = 0 ? ja: Sprung nach 154
141		LDI	44	8	C4	44	
142		XPAL 3		A	33		
143		ST	1D (P1)	B	C9	1D	(Modus) = 04 ? ja: Sprung nach 151
144		LD	1F (P1)	D	C1	1F	
145		JZ	REAUS ↓	F	98	0A	
146		SR		00F1	1C		
147		SR		2	1C		
148		NOP		3	08		
149		JNZ	ASCII	4	9C	01	Sprung nach MO1/2 362
150		XPPC 3	↓	6	3F		(P3) = 0207
151	ASCII	LDI	07	7	C4	07	
152		XPAL 3		9	33		
153		XPPC 3	↓	A	3F		Sprung nach MODE 4 324
154	REAUS	LDI	09	B	C4	09	(P3) = 0209
155		XPAL 3		D	33		
156		XPPC 3	↓	E	3F		
157				F	00		Sprung nach MODE 0 325
							Zur „Handhabung des ASCII-Codes“ siehe Blätter 4b bis 4f!

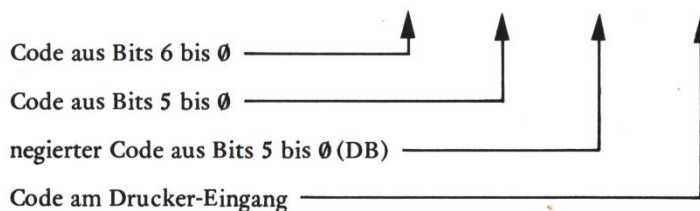


Programm: Betriebsprogramm

Blatt 4 b

Handhabung des ASCII-Codes, — Buchstaben

Z	7	6	5	4	3	2	1	0	A	B	C	D
@	1	1	0	0	0	0	0	0	40	00	3F	C0
A	0	1	0	0	0	0	0	1	41	01	3E	C1
B	1	1	0	0	0	0	1	0	42	02	3D	C2
C	0	1	0	0	0	0	1	1	43	03	3C	C3
D	1	1	0	0	0	1	0	0	44	04	3B	C4
E	0	1	0	0	0	1	0	1	45	05	3A	C5
F	1	1	0	0	0	1	1	0	46	06	39	C6
G	0	1	0	0	0	1	1	1	47	07	38	C7
H	1	1	0	0	1	0	0	0	48	08	37	C8
I	0	1	0	0	1	0	0	1	49	09	36	C9
J	1	1	0	0	1	0	1	0	4A	0A	35	CA
K	0	1	0	0	1	0	1	1	4B	0B	34	CB
L	1	1	0	0	1	1	0	0	4C	0C	33	CC
M	0	1	0	0	1	1	0	1	4D	0D	32	CD
N	1	1	0	0	1	1	1	0	4E	0E	31	CE
O	0	1	0	0	1	1	1	1	4F	0F	30	CF
P	1	1	0	1	0	0	0	0	50	10	2F	D0
Q	0	1	0	1	0	0	0	1	51	11	2E	D1
R	1	1	0	1	0	0	1	0	52	12	2D	D2
S	0	1	0	1	0	0	1	1	53	13	2C	D3
T	1	1	0	1	0	1	0	0	54	14	2B	D4
U	0	1	0	1	0	1	0	1	55	15	2A	D5
V	1	1	0	1	0	1	1	0	56	16	29	D6
W	0	1	0	1	0	1	1	1	57	17	28	D7
X	1	1	0	1	1	0	0	0	58	18	27	D8
Y	0	1	0	1	1	0	0	1	59	19	26	D9
Z	1	1	0	1	1	0	1	0	5A	1A	25	DA
[	0	1	0	1	1	0	1	1	5B	1B	24	DB
~	1	1	0	1	1	1	0	0	5C	1C	23	DC
]	0	1	0	1	1	1	0	1	5D	1D	22	DD
{	1	1	0	1	1	1	1	0	5E	1E	21	DE
}	0	1	0	1	1	1	1	1	5F	1F	20	DF

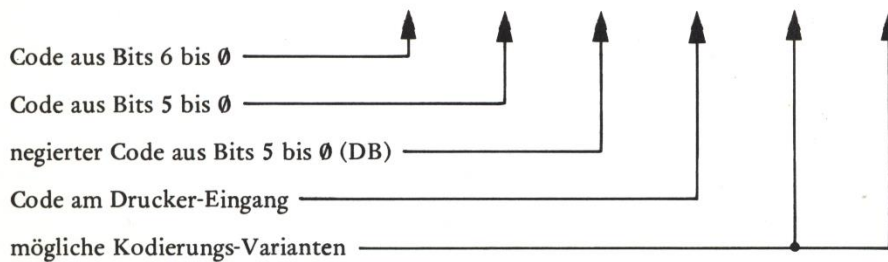




Programm: Betriebsprogramm
Handhabung des AscII-Codes, – Zahlen/Sonderzeichen

Blatt 4 c

Z	7	6	5	4	3	2	1	0	A	B	C	D	E	F
SP	1	0	1	0	0	0	0	0	20	20	1F	E0	A0	60
!	0	0	1	0	0	0	0	1	21	21	1E	E1	A1	61
“	1	0	1	0	0	0	1	0	22	22	1D	E2	A2	62
#	0	0	1	0	0	0	1	1	23	23	1C	E3	A3	63
\$	1	0	1	0	0	1	0	0	24	24	1B	E4	A4	64
%	0	0	1	0	0	1	0	1	25	25	1A	E5	A5	65
&	1	0	1	0	0	1	1	0	26	26	19	E6	A6	66
'	0	0	1	0	0	1	1	1	27	27	18	E7	A7	67
(	1	0	1	0	1	0	0	0	28	28	17	E8	A8	68
)	0	0	1	0	1	0	0	1	29	29	16	E9	A9	69
*	1	0	1	0	1	0	1	0	2A	2A	15	EA	AA	6A
+	0	0	1	0	1	0	1	1	2B	2B	14	EB	AB	6B
,	1	0	1	0	1	1	0	0	2C	2C	13	EC	AC	6C
–	0	0	1	0	1	1	0	1	2D	2D	12	ED	AD	6D
.	1	0	1	0	1	1	1	0	2E	2E	11	EE	AE	6E
/	0	0	1	0	1	1	1	1	2F	2F	10	EF	AF	6F
0	1	0	1	1	0	0	0	0	30	30	0F	F0	B0	70
1	0	0	1	1	0	0	0	1	31	31	0E	F1	B1	71
2	1	0	1	1	0	0	1	0	32	32	0D	F2	B2	72
3	0	0	1	1	0	0	1	1	33	33	0C	F3	B3	73
4	1	0	1	1	0	1	0	0	34	34	0B	F4	B4	74
5	0	0	1	1	0	1	0	1	35	35	0A	F5	B5	75
6	1	0	1	1	0	1	1	0	36	36	09	F6	B6	76
7	0	0	1	1	0	1	1	1	37	37	08	F7	B7	77
8	1	0	1	1	1	0	0	0	38	38	07	F8	B8	78
9	0	0	1	1	1	0	0	1	39	39	06	F9	B9	79
:	1	0	1	1	1	0	1	0	3A	3A	05	FA	BA	7A
;	0	0	1	1	1	0	1	1	3B	3B	04	FB	BB	7B
<	1	0	1	1	1	1	0	0	3C	3C	03	FC	BC	7C
=	0	0	1	1	1	1	0	1	3D	3D	02	FD	BD	7D
>	1	0	1	1	1	1	1	0	3E	3E	01	FE	BE	7E
?	0	0	1	1	1	1	1	1	3F	3F	00	FF	BF	7F



Programm: Betriebsprogramm
Handhabung des ASCII-Codes, — Erläuterung

Blatt 4 d

Die **Spalte Z** enthält die zu druckenden Zeichen.

Die **Spalten 7 bis 0** enthalten den zugehörigen ASCII-Code in sedezimaler Schreibweise.

Die **Spalten 7 und 6** werden im Betriebsprogramm nicht benötigt.

Nur der Vollständigkeit halber:

Normalerweise ist das Bit 7 ein Kontrollbit (Parity-Bit):

Bit 7 = 1, wenn die Spalten 6 bis 0 ungeradzahlige Bit-Werte aufweisen:
1, 3, 5, 7

Bit 7 = 0, wenn die Spalten 6 bis 0 geradzahlige Bit-Werte aufweisen:
0, 2, 4, 6

Die **Spalten 5 bis 0** machen eine eindeutige Aussage über das auszudruckende Zeichen, — die Spalten 7 und 6 sind nicht relevant.

Aus diesem Grunde werden mit dem Befehl $\textcircled{449}$ ANI 3F aus dem im „Scratchpad Drucker“ abgelegten Zeichen nur die Bits 5 bis 0 herausgesiebt.

Die **Spalte A** enthält die Sedezimal-Ausdrücke der Bits 6 bis 0 (mit Bit 7 = 0), die auch in Tafel 4 des Lehrgangs angegeben sind.

Die **Spalte B** enthält die Sedezimal-Ausdrücke der „ausgesiebten“ Bits 5 bis 0 (mit Bit 7 = 0 und Bit 6 = 0), die nach Ausführung des o.g. Befehls $\textcircled{449}$ ANI 3F übrig bleiben. (Im Falle der Zahlen/Sonderzeichen sind die Spalten A und B identisch).

Durch den Befehl $\textcircled{450}$ XRI 3F wird die Spalte B (Bits 5 bis 0) negiert. Es entstehen dadurch die Sedezimal-Ausdrücke in **Spalte C**. Diese Negierung ist erforderlich, weil die Signale des Datenbusses hinter dem Zweiweg-Bus-Treiber DS 8834 negativ anstehen.

Wenn man mit dem anschließenden Befehl $\textcircled{451}$ ST 00 (P1), z.B. um den Buchstaben F zu drucken, aus Spalte C $39_{16} = 0011\ 1001$ auf den Datenbus gibt, erscheint am Eingang des Druckers:

$$1100\ 0110 = C6_{16}.$$

Diese erneut negierten Werte sind in **Spalte D** angegeben.

Da die Bits 6 und 7 nicht relevant sind, würden für die Zahlen/Sonderzeichen auch die Kodierungen zum richtigen Ergebnis führen, die der Vollständigkeit halber noch in den **Spalten E und F** aufgeführt sind.

Durch das Byte $C0_{16} = 1100\ 0000$ oder auch $11XX\ XXXX$ auf dem negativen Datenbus, bzw. (Befehle $\textcircled{465}$ und $\textcircled{466}$) $0011\ 1111 = 3F_{16}$ oder auch $00XX\ XXXX$ am Drucker-Eingang, wird der Drucker angehalten. (Vergl. auch Seite H 158 des Lehrgangs: „Druckvorgang anhalten“: Tastschalter an der Frontplatte betätigen,

8, 0, 6, AD, F, F, IN Tastschalter loslassen. Dadurch wird

1111 1111 auf den negativen Datenbus gegeben!)

Programm: Betriebsprogramm, –
Handhabung des ASCII-Codes

Blatt 4 e

Beispiel: Register-Ausdruck

[illegible]



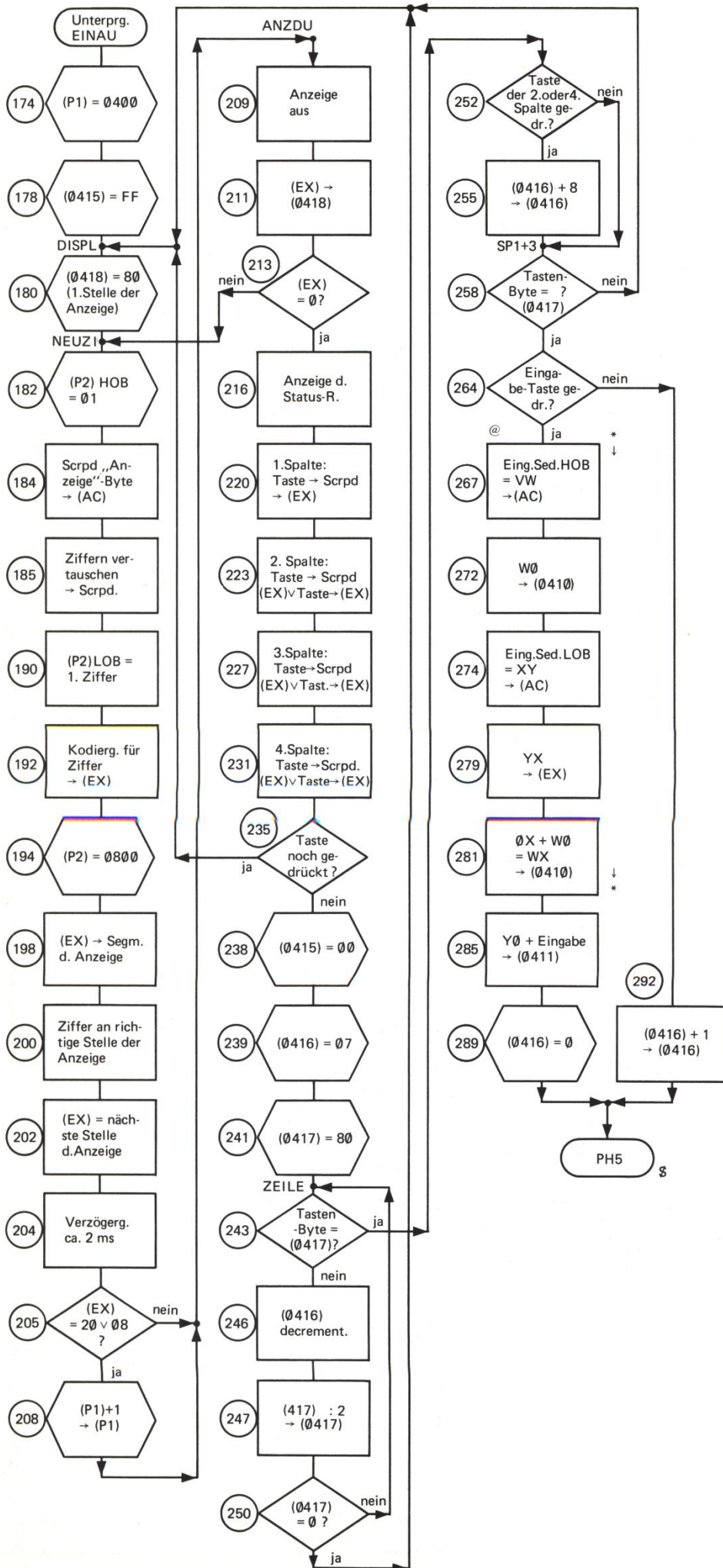
Programm: Betriebsprogramm
UP „Ein-Aus“

Blatt 5

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen		
158				0100	C0		„0“	DB 7 6 5 4 3 2 DB 0 4 2 0 C 4 DP G F E D C B A DB 7 6 5 4 3 2 1 0 DB A F B E G C D DP	Kodierung des Display
159				1	F9		„1“		
160				2	A4		„2“		
161				3	B0		„3“		
162				4	99		„4“		
163				5	92		„5“		
164				6	82		„6“		
165				7	F8		„7“		
166				8	80		„8“		
167				9	90		„9“		
168				A	88		„A“		
169				B	83		„B“		
170				C	C6		„C“		
171				D	A1		„D“		
172				E	86		„E“		
173				F	8E		„F“		
174	EINAU	LDI	04	0110	C4	04	(P1) = 0400	Ein-Aus, - Display	
175		XPAH 1		2	35				
176		LDI	00	3	C4	00			
177		XPAL 1		5	31				
178		LDI	FF	6	C4	FF			
179		ST	15 (P1)	8	C9	15			
180	DISPL	LDI	80	011A	C4	80			
181		ST	18 (P1)	C	C9	18			
182	NEUZI	LDI	01	E	C4	01			
183		XPAH 2		0120	36				
184		LD	12 (P1)	1	C1	12	(0412, 0413, 0414) = Anzeige-Adr. HOB, -Adr. LOB, -Opcode, → (AC), beide Ziffern vertauschen, Byte in umgekehrter Reihenfolge wieder abspeichern → (0412, 0413, 0414) 1. Ziffer heraussieben, (P2) LOB = anzuzeigende Ziffer		
185		RR		3	1E				
186		RR		4	1E				
187		RR		5	1E				
188		RR		6	1E				
189		ST	12 (P1)	7	C9	12			
190		ANI	0F	9	D4	0F			
191		XPAL 2		B	32				
192		LD	00 (P2)	C	C2	00			
193		XAE		E	01				
194		LDI	08	F	C4	08	(P2) = 0800 (Extern)		
195		XPAH 2		0131	36				
196		LDI	00	2	C4	00			
197		XPAL 2		4	32				
198		XAE		5	01		Kodierung für Ziffer		
199		ST	04 (P2)	6	CA	04	→ Segmente der Anzeige		
200		LD	18 (P1)	8	C1	18	Ziffer		
201		ST	05 (P2)	A	CA	05	→ richtige Stelle der Anzeige		
202		SR		C	1C		(EX) = nächste Stelle der Anzeige		
203		XAE		D	01				
204		DLY	02	E	8F	02	Verzögerung: ca. 2 ms		

Kodierung
des Display

Ein-Aus,
Display



Betriebsprogramm Unterprogramm EIN-AUS

* Die eingegebene Ziffer wird jeweils von rechts her eingeschoben:

Alter Stand:

V W X Y

Neuer Stand:

W X Y E

§ Nach dem Rücksprung ins Hauptprogramm enthält die Speicherzelle (0416) entweder 0, (wenn eine oder mehrere Eingabetasten gedrückt wurden) oder die Tastenkodierung, (wenn eine Funktionstaste gedrückt wurde).

Weiter:

(0410) = Eingabe HOB
(0411) = Eingabe LOB

(0415) } →
(0417) }

uninteressant!

@ Wenn Eingabetaste gedrückt, enthält (0416) Tastenkodierung



Programm: Betriebsprogramm, -
Ein-Aus

Blatt 6

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen	
205		LDE		0140	40		neues Byte? ((EX) = 20 V 08 ?)	Ein-Aus, - Display
206		ANI	28	1	D4	28		
207		JZ	ANZDU ↓	3	98	02	nein: Sprung nach ANZDU	
208		LD	@ 01 (P1)	5	C5	01	(P1) = (P1) + 1, - da neues Byte	
209	ANZDU	LDI	00	0147	C4	00	Anzeige aus	
210		ST	05 (P2)	9	CA	05		
211		LDE		B	40			
212		ST	18 (P1)	C	C9	18	(0418) = 40, 20, 10, 08, 04, 02, 01, 00	
213		JNZ	NEUZI ↑	E	9C	CE	(0418) ≠ 0 → Rückspr. nach 182	
214		LDI	00	0150	C4	00	(P1) LOB = 00 ((P1) = 0400)	
215		XPAL 1		2	31			Anzeige SR
216		LD	09 (P1)	3	C1	09	CY/L, OV, IE, werden angezeigt	
217		XRI	C8	5	E4	C8		
218		ST	07 (P2)	7	CA	07	→ (0807)	Ein-Aus, - Tasten
219		NOP		9	08			
220		LD	00 (P2)	A	C2	00	(0800) → (0418)	
221		ST	18 (P1)	C	C9	18	(Tasten, 1. Sp. → Scratchpad)	
222		XAE		E	01		→ (EX)	
223		LD	01 (P2)	F	C2	01	(0801) → (0419)	
224		ST	19 (P1)	0161	C9	19	(Tasten, 2. Sp. → Scratchpad)	
225		ORE		3	58		(0418) ∨ (0419) → (EX)	
226		XAE		4	01			
227		LD	02 (P2)	5	C2	02	(0802) → (041A)	
228		ST	1A (P1)	7	C9	1A	(Tasten, 3. Sp. → Scratchpad)	
229		ORE		9	58		(0418) ∨ (0419) ∨ (041A) → (EX)	
230		XAE		A	01			
231		LD	03 (P2)	B	C2	03	(0803) → (041B)	
232		ST	1B (P1)	D	C9	1B	(Tasten, 4. Sp. → Scratchpad)	
233		ORE		F	58		(0418) ∨ (0419) ∨ (041A) ∨ (041B)	
234		XAE		0170	01		→ (EX)	
235		LD	15 (P1)	1	C1	15	FF ∧ (EX) ≠ 0 ? (Taste noch gedr. ?)	
236		ANE		3	50		ja: Rücksprung nach 180	
237		JNZ	DISPL ↑	4	9C	A4	(alle anderen Tasten werden gesperrt)	
238		ST	15 (P1)	6	C9	15	(0415) = 00	
239		LDI	07	8	C4	07	(0416) = 07 (8 Zeilen!)	
240		ST	16 (P1)	A	C9	16		
241		LDI	80	C	C4	80	(0417) = 80 (für 8. Zeile)	
242		ST	17 (P1)	E	C9	17		
243	ZEILE	LD	17 (P1)	0180	C1	17	(0417) = (EX) = Tastenbyte?	
244		XRE		2	60		ja: Sprung nach 252	
245		JZ	SPALT ↓	3	98	0B		
246		DLD	16 (P1)	5	B9	16	(0416) - 1 → (0416) für nächste Zeile	
247		LD	17 (P1)	7	C1	17	(0417) geteilt durch 2	
248		SR		9	1C		→ (0417) = 0 ?	
249		ST	17 (P1)	A	C9	17		
250		JNZ	ZEILE ↑	C	9C	F2	nein: Rücksprung nach 243	
251	ANZEI	JMP	DISPL ↑	018E	90	8A	Rücksprung nach 180	



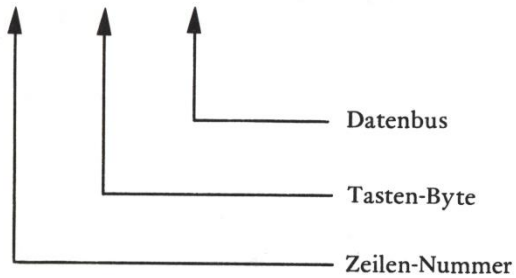
Programm: Betriebsprogramm

Blatt 7 a

Kodierung der Tasten = (0416) → grüne Werte

Tasten-Aufdruck → rote Werte

		Spalte 1		Spalte 2		Spalte 3		Spalte 4	
Per./Tasten		0800		0801		0802		0803	
Scratchp.		0418		0419		041A		041B	
1	01	0	00	8	08	PC	01	PT	09
2	02	1	01	9	09	P1	02	SI	0A
3	04	2	02	A	0A	P2	03	GO	0B
4	08	3	03	B	0B	P3	04	AD	0C
5	10	4	04	C	0C	AC	05	ME	0D
6	20	5	05	D	0D	EX	06	IN	0E
7	40	6	06	E	0E	BP	07		0F
8	80	7	07	F	0F	0802	08	0803	10

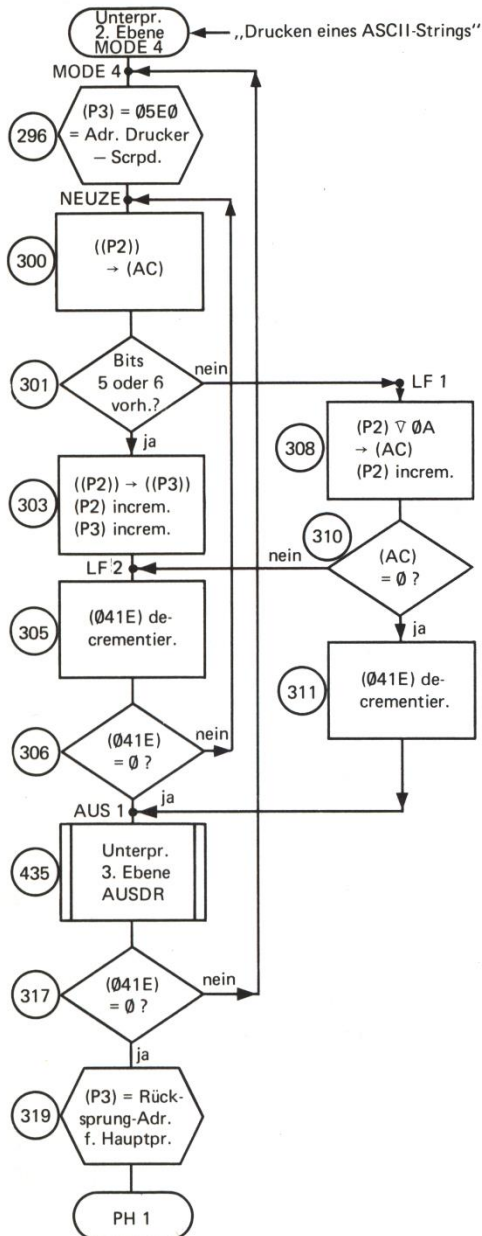




Programm: Betriebsprogramm
UP „Ein-Aus“

Blatt 7

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen	
252	SPALT	LD	19 (P1)	0190	C1	19	(0419) ∨ (041B) ≠ 0 ? (Keine Taste der 2. oder 4. Spalte?)	Ein-Aus, – Tasten (Forts.)
253		OR	1B (P1)	2	D9	1B	nein: Sprung nach 258	
254		JZ	SP 1+3 ↓	4	98	06		
255		LDI	08	6	C4	08		
256		OR	16 (P1)	8	D9	16	08 ∨ (0416) → (0416)*	
257		ST	16 (P1)	A	C9	16		
258	SP 1 + 3	LD	00 (P2)	019C	C2	00	(0800) ∨ (0801) ∨ (0802) ∨ (0803)	
259		OR	01 (P2)	E	DA	01	∇ (0417) = 0 ?	
260		OR	02 (P2)	01A0	DA	02	Stimmt Tasten-Byte mit Inhalt von (0417) überein?	
261		OR	03 (P2)	2	DA	03	nein: Rücksprung nach 251	
262		XOR	17 (P1)	4	E1	17		
263		JNZ	ANZEI ↑	6	9C	E6		
264		LD	18 (P1)	8	C1	18	(0418) ∨ (0419) = 0 ? (Es ist keine Eingabe-Taste gedrückt!)	
265		OR	19 (P1)	A	D9	19	ja: Sprung nach 292	
266		JZ	FUNKT ↓	C	98	25		
267		LD	10 (P1)	E	C1	10	(0410) = Eingabe Sed. HOB = VW	
268		RR		01B0	1E			
269		RR		1	1E		VW vertauschen: WV	
270		RR		2	1E			
271		RR		3	1E			
272		ANI	F0	4	D4	F0	W0 herausziehen in 0410 abspeichern	
273		ST	10 (P1)	6	C9	10		
274		LD	11 (P1)	8	C1	11	(0411) = Eingabe Sed. LOB = XY	
275		RR		A	1E			
276		RR		B	1E		XY vertauschen: YX	
277		RR		C	1E			
278		RR		D	1E			
279		XAE		E	01			
280		LDE		F	40		abspeichern in EX: → (EX) → (AC)	
281		ANI	0F	01C0	D4	0F	0X herausziehen	
282		CCL		2	02		CY/L löschen	
283		ADD	10 (P1)	3	F1	10	(0410) hinzu addieren	
284		ST	10 (P1)	5	C9	10	in 0410 abspeichern (HOB)	
285		LDE		7	40		(EX) → (AC)	
286		ANI	F0	8	D4	F0	Y0 herausziehen	
287		ADD	16 (P1)	A	F1	16	(0416) addieren	
288		ST	11 (P1)	C	C9	11	in 0411 abspeichern (LOB)	
289		LDI	00	E	C4	00		
290		ST	16 (P1)	01D0	C9	16	(0416) = 00	
291		XPPC 3	↑	2	3F		Rückspr. nach „Eingabe“ 36	
292	FUNKT	ILD	16 (P1)	3	A9	16	(0416) incrementieren *	
293		XPPC 3	↑	5	3F		Rückspr. nach „Eingabe“ 36	
294				6	00			
295				7	00			
							* (0416) enthält die Kodierung der Tasten, – bei Eingabe-Tasten: Zeilen-Nr. minus 1 (+8), – bei Funkt.-Tasten: Zeilen-Nr. (+8)	



Betriebsprogramm: Unterprogramm MODE 4 (Unterprogramm in der 2. Ebene,- Rückspr. direkt ins Hauptprogramm)

Relevante Parameter, — unmittelbar nach dem
Ansprung des Unterprogramms:

(P1) = 0400 (134), (P2) = Anz.-Adresse (22)

(041E) = Anzahl der Zeichen (116)

(041F) = Modus (Druckart-Kode) = 04 (114)

(041C) = (P3)HOB alt } Rückspr.-Adr.
(041D) = (P3)LOB alt } f.Hauptpr.

Im Unterprogramm werden verwendet:

(P3) = Adr. für Unterpr. d.3.Ebene
(P2) = effektive Druckadresse

- 301 Hier wird untersucht, ob es sich um ein druckbares Zeichen handelt
- 308 → 310 Hier wird untersucht, ob es sich um einen „Zeilen-Vorschub“-Befehl handelt

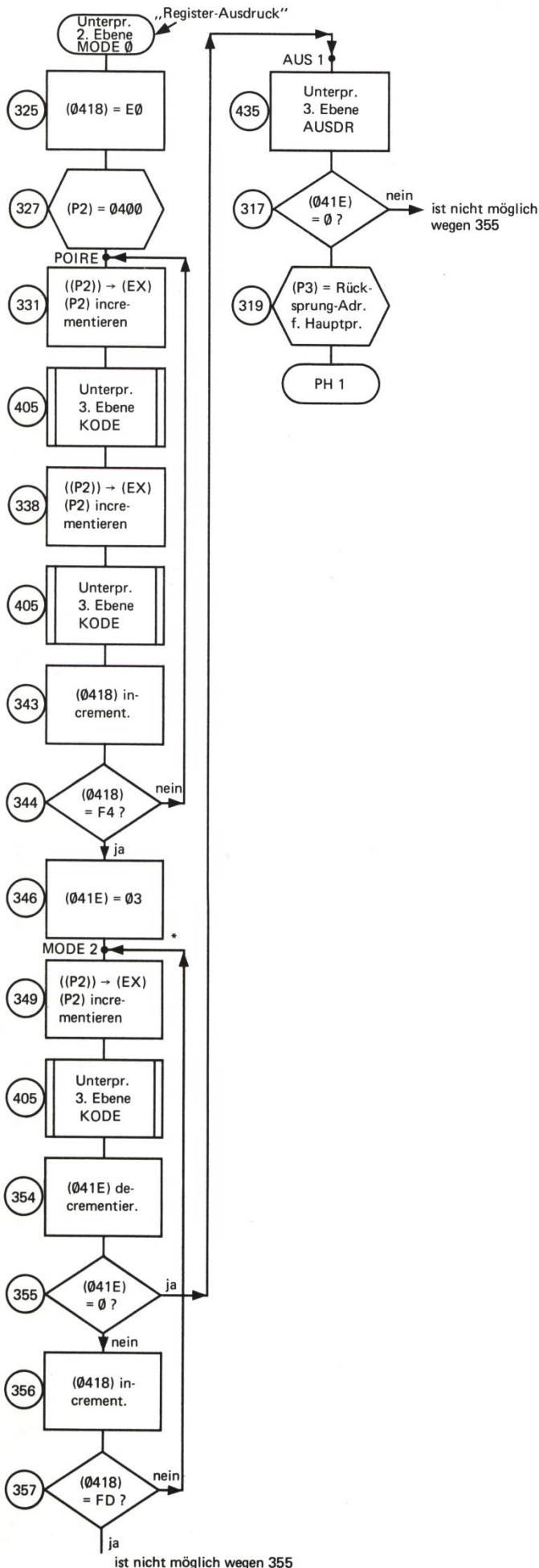


Programm: Betriebsprogramm, —

Blatt 8

UP „Mode 4“ und „Mode 0“

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen		
296	MODE 4	LDI	05	01D8	C4	05	(P3) = 05E0	(P2) = Anzeige-Adresse (noch von 22 her)	ASCII-String
297		XPAH 3		A	37				
298		LDI	E0	B	C4	E0			
299		XPAL 3		D	33				
300	NEUZE	LD	00 (P2)	E	C2	00	Inhalt der angezeigten Adresse laden, — Bits 5 und 6 vorhanden?		
301		ANI	69	01E0	D4	60	nein: Sprung nach 308		
302		JZ	LF1 ↓	2	98	0A	((P2)) → ((P3))		
303		LD	@ 01 (P2)	4	C6	01	(P2) und (P3) incrementieren		
304		ST	@ 01 (P3)	6	CF	01	Zeichenzähler = 0 ?		
305	LF 2	DLD	1E (P1)	8	B9	1E	nein: Rücksprung nach 300		
306		JNZ	NEUZE ↑	A	9C	F2	Sprung nach 312		
307		JMP	AUS 1 ↓	C	90	08	nächstes Zeichen = 0A ?		
308	LF 1	LDI	0A	E	C4	0A	nein: Rücksprung nach 305		
309		XOR	@ 01 (P2)	01F0	E6	01	Zeichenzähler decrementieren		
310		JNZ	LF2	2	9C	F4			
311		DLD	1E (P1)	4	B9	1E			
312	AUS 1	LDI	02	6	C4	02	(P3) = 02AF	Ausdrucken	
313		XPAH 3		8	37				
314		LDI	AF	9	C4	AF			
315		XPAL 3		B	33				
316		XPPC 3	↓	C	3F				
317		LD	1E (P1)	D	C1	1E	Sprung nach AUSDR. 435		
318		JNZ	MODE 4 ↑	F	9C	D7	wenn Zeichenzähler (041E) noch nicht = 0, → Rücksprung nach 296		
319		LD	1C (P1)	0201	C1	1C	$(P3) = \left\{ \begin{matrix} (041C) \\ (041D) \end{matrix} \right\} (P3) \text{ alt}$	Rückspr. n. Betr.-Pr. „Eingabe“	
320		XPAH 3		3	37				
321		LD	1D (P1)	4	C1	1D			
322		XPAL 3		6	33				
323		XPPC 3	↑	7	3F		Rücksprung nach 122		
324		JMP	MODE 4 ↑	8	90	CE	Rücksprung nach 296		
325	MODE 0	LDI	E0	A	C4	E0	(0418) = E0	Register Inhalt	
326		ST	18 (P1)	C	C9	18			
327		LDI	04	E	C4	04	(P2) = 0400		
328		XPAH 2		0210	36				
329		LDI	00	1	C4	00			
330		XPAL 2		3	32				
331	POIRE	LD	@ 01 (P2)	4	C6	01	((P2)) → (EX), (P2) increment.		
332		XAE		6	01				
333		LDI	02	7	C4	02	(P3) = 0280	Kodieren, — Scrpd. Dr. laden	
334		XPAH 3		9	37				
335		LDI	80	A	C4	80			
336		XPAL 3		C	33				
337		XPPC 3	↓	D	3F		Sprung nach KODE 405		
338		LD	@ 01 (P2)	E	C6	01	((P2)) → (EX), (P2) increment.		
339		XAE		0220	01				
340		LDI	80	1	C4	80	(P3) = 0280	Kodieren Scr.-Dr. laden	
341		XPAL 3		3	33				
342		XPPC 3	↓	4	3F		Sprung nach KODE 405		



Betriebsprogramm: Unterprogramm MODE 0 (Unterprogramm in der 2. Ebene,- Rücksprung direkt ins Hauptprogramm)

* In der Schleife 331 → 344 werden
die Inhalte der Pointer-Register
P0 bis P3 kodiert und in das
Drucker-Scratchpad geladen.

* In der Schleife 349 → 357 werden
die Inhalte der Arbeits-Register
AC, SI und EX kodiert und in das
Drucker-Scratchpad geladen.

Relevante Parameter, - unmittelbar nach dem
Ansprung des Unterprogramms:

(P1) = 0400 (134) (041F) = Modus = 00 (114)

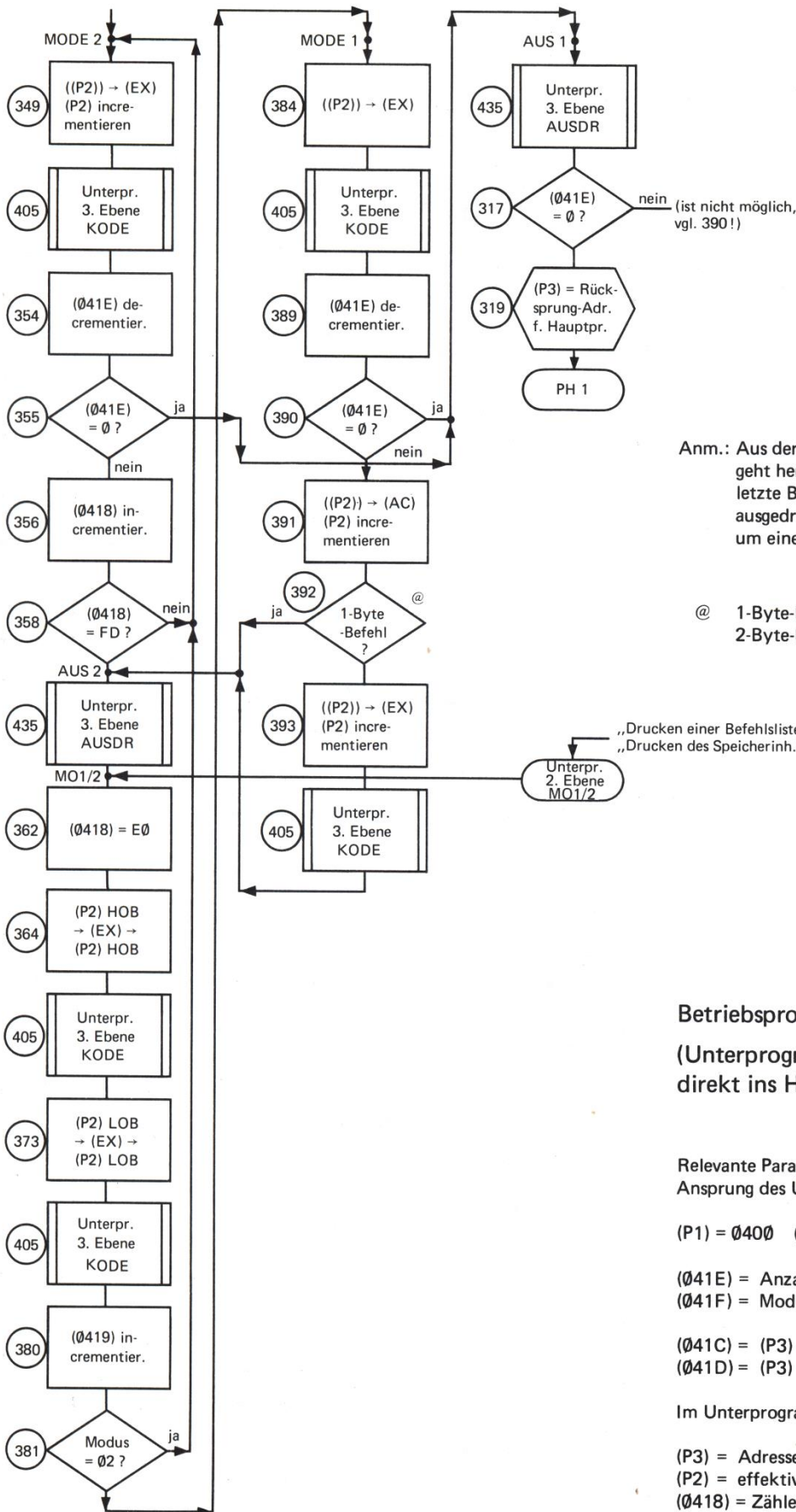
(041C) = (P3)HOB alt } (P3)alt = Rücksprung-Adr.
(041D) = (P3)HOB alt } für Hauptprogramm

Im Unterprogramm werden verwendet:

(P2) = Register-Byte,-effektive Druckadresse
(P3) = Adresse für Unterprogramme der 3. Ebene

(0418) = Zähler für Drucker-Scratchpad (für P0, P1, P2, P3)

(041E) = Zähler für AC, ST, EX



Anm.: Aus der Programmfolge (390) → (393) geht hervor, daß der letzte Befehl nur unvollständig ausgedruckt wird, wenn es sich um einen 2-Byte-Befehl handelt!

@ 1-Byte-Befehle: positiv
2-Byte-Befehle: negativ

Betriebsprogramm: Unterprogramm MO 1/2
(Unterprogramm in der 2. Ebene, - Rücksprung direkt ins Hauptprogramm)

Relevante Parameter, - unmittelbar nach dem Ansprung des Unterprogramms:

(P1) = 0400 (134), (P2) = Anzeige-Adresse (22)
(041E) = Anzahl der Befehle, bzw. Bytes (116)
(041F) = Modus (Druckart-Kode) = 01 V 02 (114)
(041C) = (P3) HOB alt } (P3)alt = Rücksprung-Adr. für Hauptprogramm
(041D) = (P3) LOB alt }

Im Unterprogramm werden verwendet:

(P3) = Adresse für Unterprogramme der 3. Ebene
(P2) = effektive Druck-Adresse
(0418) = Zähler für Drucker-Scratchpad

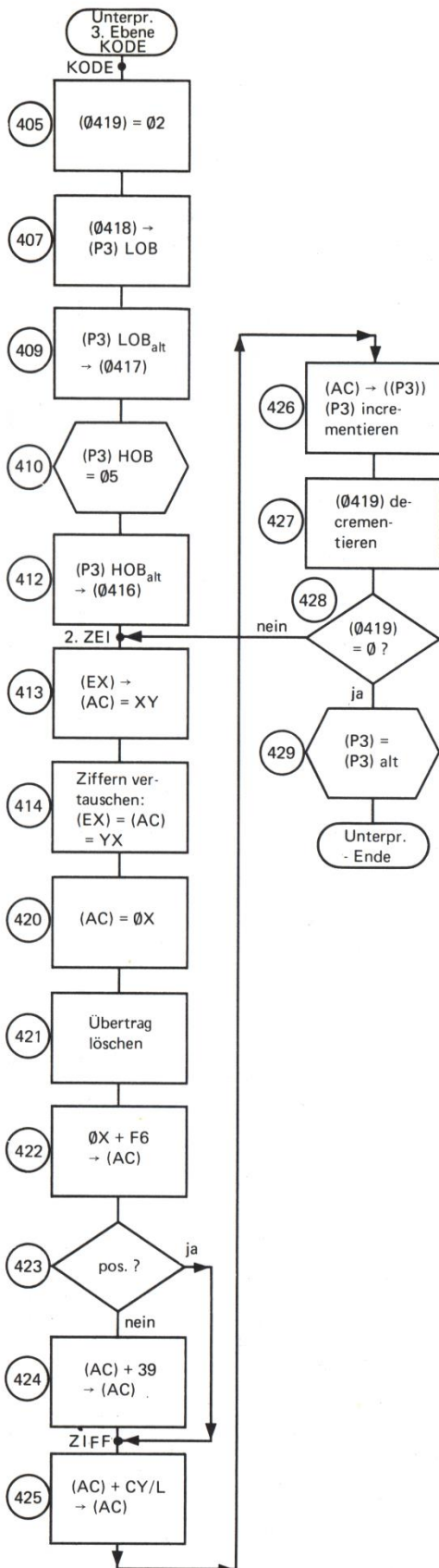


Programm: Betriebsprogramm, —

Blatt 9

UP „Mode 4“ und „Mode 0“

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen		
343		ILD	18 (P1)	0225	A9	18	(0418) = F4 ? nein: Rücksprung nach 331	Register - Inhalt	
344		XRI	F4	7	E4	F4			
345		JNZ	POIRE ↑	9	9C	E9			
346		LDI	03	B	C4	03	(041E) = 03 (für AC, ST, EX)	Kod., Scrp. - pd. Dr. laden	
347		ST	1 E (P1)	D	C9	1E			
348		NOP		F	08				
349	MODE 2	LD	@ 01 (P2)	0230	C6	01	((P2)) → (EX)	Aus- drucken	
350		XAE		2	01		(P2) incrementieren		
351		LDI	80	3	C4	80	(P3) = 0280		
352		XPAL 3		5	33		Sprung nach KODE 405	Kodieren, - Scrp. Dr. laden	
353		XPPC 3	↓	6	3F				
354		DLD	1E (P1)	A	B9	1E			
355		JZ	AUS 1 ↑	9	98	BB	((041E)) - 1 = 0 ? ja: Rücksprung nach 312	Befehlsliste Speich.-Inh.	
356		ILD	18 (P1)	B	A9	18	(0418) incrementieren, - = FD ?		
357		XRI	FD	D	E4	FD	nein: Rücksprung nach 349		
358		JNZ	MODE 2 ↑	F	9C	EF	(P3) = 02AF	Kodieren, - Scrp. Dr. laden	
359	AUS 2	LDI	AF	0241	C4	AF			
360		XPAL 3		3	33				
361		XPPC 3	↓	4	3F		Sprung nach AUSDR 435	Befehlsliste Speich.-Inh.	
362	MO 1/2	LDI	E0	5	C4	E0	(0418) = E0 = Effektive Adresse im Scrp. Drucker		
363		ST	18 (P1)	7	C9	18			
364		XPAH 2		9	36		(P2) HOB in (EX) zwischenspeichern → (P2) HOB	(P2) = Anzeige- Adresse	
365		XAE		A	01				
366		LDE		B	40				
367		XPAH 2		C	36		(P3) = 0280	Kodieren, - Scrp. Dr. laden	
368		LDI	02	D	C4	02			
369		XPAH 3		F	37				
370		LDI	80	0250	C4	80	Sprung nach KODE 405	Kodieren, - Scrp. Dr. laden	
371		XPAL 3		2	33				
372		XPPC 3	↓	3	3F				
373		XPAL 2		4	32		(P2) LOB in (EX) zwischenspeichern → (P2) LOB	(P2) = Anzeige- Adresse	
374		XAE		5	01				
375		LDE		6	40				
376		XPAL 2		7	32		(P3) = 0280	Kodieren, - Scrp. Dr. laden	
377		LDI	80	8	C4	80			
378		XPAL 3		A	33				
379		XPPC 3	↓	B	3F		Sprung nach KODE 405	Befehlsliste Speich.-Inh.	
380		ILD	18 (P1)	C	A9	18	(0418) incrementieren		
381		LD	1F (P1)	E	C1	1F	(041F) = Modus = 02 ?		
382		SR		0260	1C		ja: Rücksprung nach 349	Kodieren, - Scrp. Dr. laden	
383		JNZ	MODE 2 ↑	1	9C	CD			
384	MODE 1	LD	00 (P2)	3	C2	00	(041F) = Modus = 01		
385		XAE		5	01		ja: ((P2)) → (EX)	Kod., Scrp. - Dr. laden	
386		LDI	80	6	C4	80	(P3) = 0280		
387		XPAL 3		8	33				
388		XPPC 3		9	3F		Sprung nach KODE 405		



Betriebsprogramm: Unterprogramm KODE (Unterprogramm in der 3. Ebene)

Relevante Parameter, unmittelbar nach dem Ansprung des Unterprogramms:

(P1) = 0400

(P3) = Rücksprung-Adresse

(0418) = Zähler für Drucker-Scrpd.

Im Unterprogramm werden verwendet:

(P3) = effektive Druck-Adresse

(0419) = Zähler für 2 Ziffern

(0416) = (P3) HOB alt } (P3) alt = Rückspr.
(0417) = (P3) LOB alt } Adresse

(EX) = zu kodierendes Byte

Beispiel: (EX) = XY = 4F

413 (AC) = 4F	413 (AC) = F4
414 (AC) = F4	414 (AC) = 4F
420 (AC) = 04	420 (AC) = 0F
422 (AC) = FA	422 (AC) = 05, CY/L = 1
424 (AC) = 33, CY/L = 1	
425 (AC) = 34	425 (AC) = 06

(1.Durchlauf)

(2.Durchlauf)

Am Ende des Durchlaufs steht im AC jeweils die Kodierung für das zu druckende Zeichen:

34 für „4“ und 06 für „F“

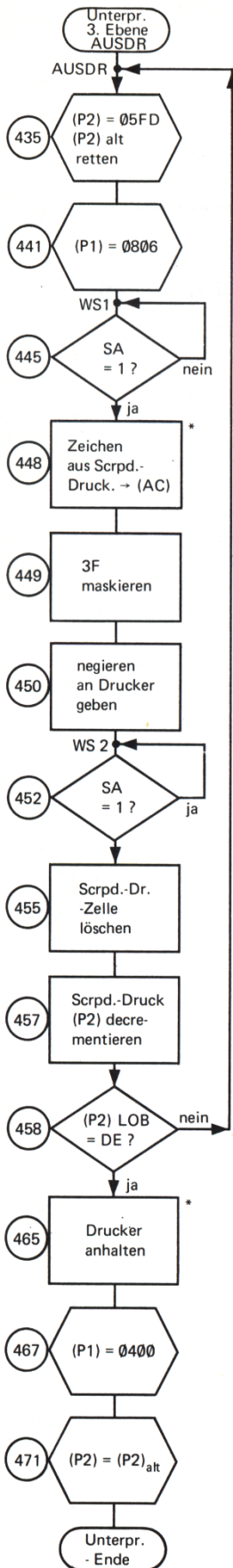
(Siehe hierzu Sonderkapitel „Handhabung des ASCII-Codes“!)



Programm: Betriebsprogramm, —
UP „Kode“

Blatt 10

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen		
389		DLD	1E (P1)	026A	B9	1E	Anzahl der Befehle / Bytes = 0?	Befehlsliste Speicherinh.	↓
390		JZ	AUS 1 ↑	C	98	87	ja: Rücksprung nach 312		
391		LD	@ 01 (P2)	E	C6	01	1-Byte-Befehl ?		
392		JP	AUS 2 ↑	0270	94	CF	ja: Rücksprung nach 359		
393		LD	@ 01 (P2)	2	C6	01	((P2)) → (EX)		
394		XAE		4	01		(P2) incrementieren	Kodieren Scr.-Dr. laden	↓
395		LDI	80	5	C4	80	(P3) = 0280		
396		XPAL 3		7	33				
397		XPPC 3	↓	8	3F		Sprung nach KODE 405		
398		JMP	AUS 2 ↑	9	90	C6	Rücksprung nach 359		
399		HALT		B	00				
400		HALT		C	00				
401		HALT		D	00				
402		HALT		E	00				
403		HALT		F	00				
404		HALT		0280	00				
405	KODE	LDI	02	1	C4	02	(0419) = 02 (Zähler)	(EX) Kodieren und im Scratchpad- Drucker ablegen	↑
406		ST	19 (P1)	3	C9	19			
407		LD	18 (P1)	5	C1	18	(0418) → (P3) LOB		
408		XPAL 3		7	33		Rücksprung-Adr. retten:		
409		ST	17 (P1)	8	C9	17	(P3) LOB, alt → (0417)		
410		LDI	05	A	C4	05	05 → (P3) HOB		
411		XPAH 3		C	37		Rücksprung-Adr. retten:		
412		ST	16 (P1)	D	C9	16	(P3) HOB, alt → (0416)		
413	2. ZEI	LDE		F	40		(EX) → (AC) = XY		
414		RR		0290	1E		Ziffern vertauschen:		
415		RR		1	1E				
416		RR		2	1E				
417		RR		3	1E				
418		XAE		4	01				
419		LDE		5	40		YX → (EX) → (AC)		
420		ANI	0F	6	D4	0F	0X → (AC)		
421		CCL		8	02		Übertrag löschen		
422		ADI	F6	9	F4	F6	0X + F6 → (AC) = positiv ?		
423		JP	ZIFF ↓	B	94	02	ja: Sprung nach 425		
424		ADI	39	D	F4	39	(AC) + 39 → (AC)		
425	ZIFF	ADI	00	F	F4	00	(AC) + CY/L → (AC)		
426		ST	@ 01 (P3)	02A1	CF	01	(AC) → ((P3)), → (P3) incrementieren		
427		DLD	19 (P1)	3	B9	19	(0419) - 1 → (0419) → (AC) = 0 ?		
428		JNZ	2. ZEI ↑	5	9C	E8	nein: Rücksprung nach 413		
429		LD	16 (P1)	7	C1	16	(P3) HOB = (0416) (P3) LOB = (0417) } → P3 (P3) = Rücksprungadresse		
430		XPAH 3		9	37				
431		LD	17 (P1)	A	C1	17			
432		XPAL 3		C	33				
433		ST	18 (P1)	D	C9	18			
434		XPPC 3		F	3F		Rücksprung		



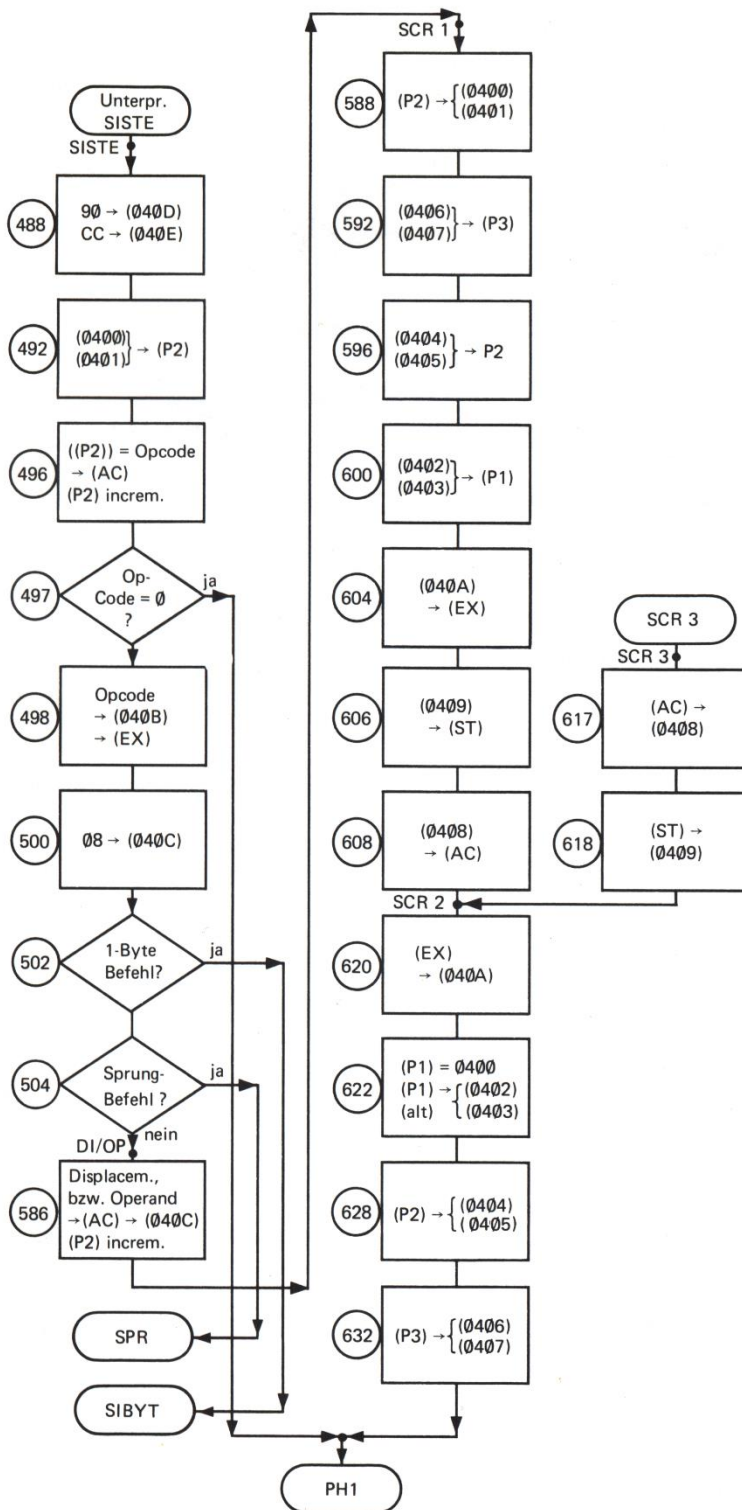
Betriebsprogramm: Unterprogramm AUSDR (Unterprogramm in der 3. Ebene)

* Den Befehlen (448) bis (450) und (465) ist das Sonderkapitel „Handhabung des ASCII-Codes“ gewidmet.

Relevante Parameter,- unmittelbar nach dem Ansprung des Unterprogramms:

(P3) = Rücksprung-Adresse

(P2) = effektive Druck-Adresse



Betriebsprogramm: Unterpr. SISTE

(Unterprogramm in der 1. Ebene)

Hier: Programmteil ohne Sprung-Befehle
und ohne Single-Byte-Befehle

Relevante Parameter, - unmittel-
bar nach Ansprung des Unter-
programms:

(P1) = 0400 (18)

(P3) = 03CF (101)

@ 532 → 535 : Ermittlung des effektiven Pointer-Registers LOB (Scrpd.-Zellen 0401, 0402, 0404, 0406), – Inhalt der angesprochenen Zelle → (P2) LOB

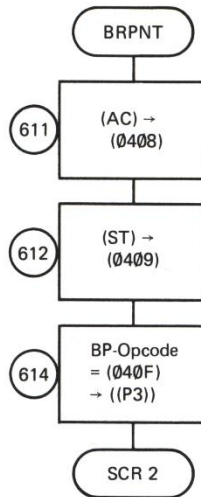
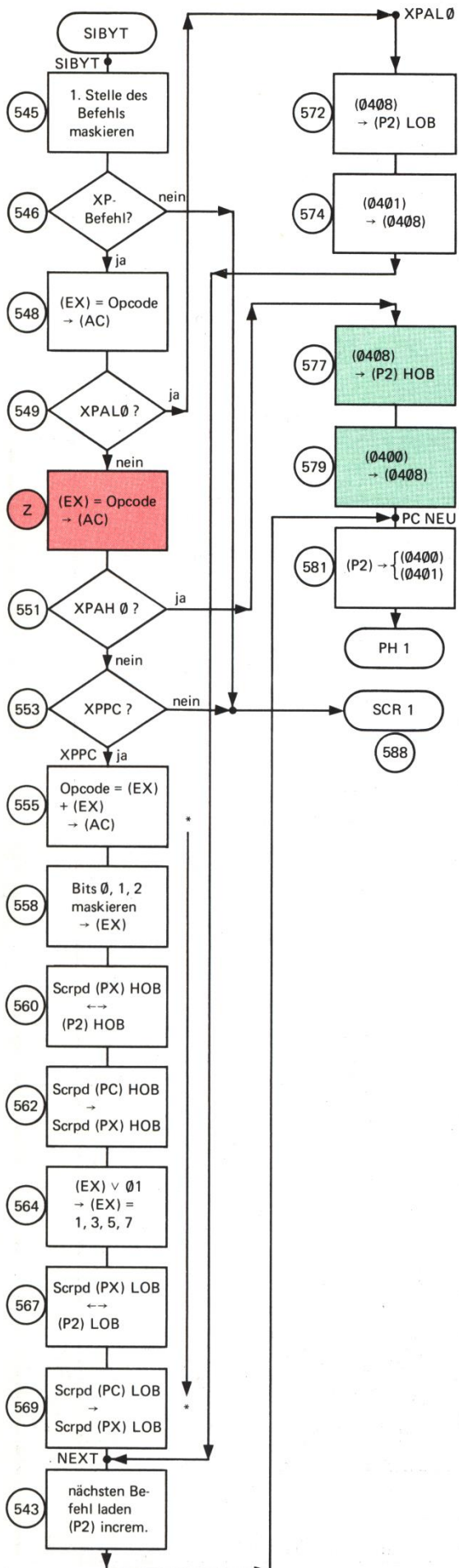
Betriebsprogramm: Unterprogramm SISTE
(Unterprogramm in der 1. Ebene)
Hier: Programmteil „Sprung-Befehl“



Programm: Betriebsprogramm, –
UP „SINGLE-STEP“

Blatt 12

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen	
478	BACK	LDI	CA	02F4	C4	CA	(P3) LOB = CA	
479		XPAL 3		6	33			
480		XPPC 3	↑	7	3F		Rücksprung nach PH1 10	
481	VOR	LDI	A8	8	C4	A8	(P3) = 03A8	
482		XPAL 3		A	33			
483		LDI	03	B	C4	03		
484		XPAH 3		D	37			
485		XPPC 3	↓	E	3F			
486		HALT		F	00			
487		HALT		0300	00			
488	SISTE	LDI	90	1	C4	90	(040D) = 90 (040E) = CC	90CC = Rückspr.-Bef. zum Label SCR 3 (617)
489		ST	0D (P1)	3	C9	0D		
490		LDI	CC	5	C4	CC	(0400) → (P2) HOB } Scrpd. f. (0401) → (P2) LOB } PC → (P2)	
491		ST	0E (P1)	7	C9	0E		
492		LD	00 (P1)	9	C1	00		
493		XPAH 2		B	36			
494		LD	01 (P1)	C	C1	01		
495		XPAL 2		E	32		nächster Befehl = 0 ? ja: Rücksprung nach 478	
496		LD	@ 01 (P2)	F	C6	01		
497		JZ	BACK ↑	0311	98	E1		
498		ST	0B (P1)	3	C9	0B		
499		XAE		5	01			
500		LDI	08	6	C4	08	(040C) = 08	
501		ST	0C (P1)	8	C9	0C		
502		LDE		A	40		(EX) = positiv (00 ÷ 7F) ?	
503		JD	SIBYT ↓	B	94	47	ja: Sprung nach 545	
504		ANI	F0	D	D4	F0	1. Stelle aussieben:	
505		XRI	90	F	E4	90	Sprungbefehl?	
506		JNZ	VOR ↑	0321	9C	D5	nein: Rücksprung nach 481	
507	SPR	LDE		3	40		(EX) = 0C, 08, 04 ?	
508		ANI	0C	4	D4	0C	(bedingter Sprung?)	
509		JZ	JMP 90 ↓	6	98	1A	nein: Sprung nach 523	
510		XRI	04	8	E4	04	(EX) = 04 ?	
511		JZ	JP 94 ↓	A	98	10	ja: Sprung nach 520	
512		XRI	0C	C	E4	0C	(EX) = 08 ?	
513		JZ	JZ 98 ↓	E	98	06	ja: Sprung nach 517	
514	JNZ 9C	LD	08 (P1)	0330	C1	08	(0408) = 0 ?	
515		JZ	NEXT ↓	2	98	2C	ja: Sprung nach 543	
516		JMP	JMP 90 ↓	4	90	0C	Sprung nach 523	
517	JZ 98	LD	08 (P1)	6	C1	08	(0408) = 0 ?	
518		JNZ	NEXT ↓	8	9C	26	nein: Sprung nach 543	
519		JMP	JMP 90 ↓	A	90	06	Sprung nach 523	
520	JP 94	LD	08 (P1)	C	C1	08	(0408) = positiv ?	
521		XRI	FF	E	E4	FF		
522		JP	NEXT ↓	0340	94	1E	nein: Sprung nach 543	



Betriebsprogramm: Unterprogramm SISTE (Unterprogramm in der 1. Ebene)

Hier: Programmteile „Ein-Byte-Befehle“
und „Breakpoint“

* (555) → (569) : Ermittlung des eff.
Pointerregisters,-
Register-Austausch: Beisp.: „3F“

(555) : $3F + 3F = 7E \rightarrow (AC)$

(558) : $7E \wedge 07 = 06 \rightarrow (EX)$

(560) : $(0406) \rightarrow (AC) \leftrightarrow (P2)HOB$

(562) : $Scrpd (PC)HOB = (0400) \rightarrow$
 $Scrpd (P3)HOB = (0406)$

(564) : $(EX) = 06 \vee 01 = 07 \rightarrow (EX)$

(567) : $(0407) \rightarrow (AC) \leftrightarrow (P2)LOB$

(569) : $Scrpd (PC)LOB = (0401) \rightarrow$
 $Scrpd (P3)LOB = (0407)$

Programmteil „Breakpoint“ wird aktiviert
durch Betätigung der Taste „GO“ (101)

Erklärung zu den Befehlen (577) → (579) : Siehe
Blatt 14 a

Programm: Betriebsprogramm, —
UP „SINGLE-STEP“

Blatt 13

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen		
523	JMP 90	LD	00 (P2)	0342	C2	00	Displacement → (P2) LOB (EX) = Opcode → (AC), Ermittlung des angesprochen. Pointer- registers (0400, 0402, 0404, 0406) → Inhalt der entsprechenden Scratchpadzelle → (P2) HOB	SINGLE-STEP Sprung-Befehle (Fortsetzung)	
524		XPAL 2		4	32				
525		LDE		5	40				
526		CCL		6	02				
527		ADE		7	70				
528		ANI	07	8	D4	07			
529		XAE		A	01				
530		LD	(EX) (P1)	B	C1	80			
531		XPAH 2		D	36				
532		LDE		E	40				
533		ORI	01	F	DC	01	Ermittlg. des angesprochen. Pointer- registers (0401, 0403, 0405, 0407) → Inhalt der entsprechenden Scratchpadzelle → (P2) LOB		
534		XAE		0351	01				
535		LD	(EX) (P1)	2	C1	80			
536		XPAL 2		4	32				
537		XAE		5	01		Displacement → (EX); (P2) um Displacement erhöhen		
538		LD	@ (EX) (P2)	6	C6	80			
539		LD	0B (P1)	8	C1	0B	Sprungbefehl PC-relativ ?		
540		ANI	03	A	D4	03			
541		JNZ	NEXT ↓	C	9C	02	nein: Sprung nach 543		
542		LD	@ 01 (P2)	E	C6	01	nächstes Byte laden, (P2) erhöhen		
543	NEXT	LD	@ 01 (P2)	0360	C6	01	nächstes Byte laden, (P2) erhöhen		
544		JMP	PC NEU ↓	2	90	3D	Sprung nach 581		
545	SIBYT	ANI	F0	4	D4	F0	1. Stelle maskieren: XP-Befehl (30 ÷ 3F) ? nein: Sprung nach 588		SINGLE-STEP Single-Byte-Befehle
546		XRI	30	6	E4	30			
547		JNZ	SCR1 ↓	8	9C	43			
548		LDE		A	40		XPAL 0 (30) ?		
549		XRI	30	B	E4	30	ja: Sprung nach 572		
550		JZ	XPAL 0 ↓	D	98	22			
551		XRI	34	F	E4	34			
552		JZ	XPAH 0 ↓	0371	98	27	ja: Sprung nach 577		
553		ANI	08	3	D4	08	XPPC (3C ÷ 3F) ? nein: Sprung nach 588		
554		JZ	SCR1 ↓	5	98	36			
555	XPPC	LDE		7	40		(EX) = Opcode → (AC)		
556		CCL		8	02		Ermittlung des angesproch. Point.- registers (0400, 0402, 0404, 0406), – Displacement → (EX) Scrpd. (PX) HOB → (P2) HOB		
557		ADE		9	70				
558		ANI	07	A	D4	07			
559		XAE		C	01				
560		LD	(EX) (P1)	D	C1	80	Scrpd. (PC) HOB → Scrpd. (PX) HOB		
561		XPAH 2		F	36				
562		LD	00 (P1)	0380	C1	00			
563		ST	(EX) (P1)	2	C9	80	Ermittlung des angesproch. Point.- registers (0401, 0403, 0405, 0407), –Displacement → (EX)		
564		XAE		4	01				
565		ORI	01	5	DC	01			
566		XAE		7	01				
567		LD	(EX) (P1)	8	C1	80	Scrpd. (PX) LOB → (P2) LOB		
568		XPAL 2		A	32				



Programm: Betriebsprogramm
Unterprogramm SISTE

Blatt 14 a

Man kann sich leicht davon überzeugen, daß der Befehl XPAH 0 nicht im Single-Step-Verfahren funktioniert.

0420 C4 05 Durch die beiden Befehle auf den Plätzen 0420 und
0422 34 0422 wird ein kleines Blinkprogramm angesprungen,
das bei 0523 beginnt.
0523 C4 02
0525 07 GO → normale Funktion
0526 8F FF
0528 C4 05 SI → 1. Schritt C4 05 wird ausgeführt
052A 07
052B 8F FF SI → **kein** Einzelschritt, — Übergang auf „GO“
052D 90 F4

Der Befehl XPAL 0 dagegen wird im Single-Step-Verfahren normal ausgeführt:

0420 C4 44 GO → normale Funktion
0422 30
SI → 1. Schritt C4 05 wird ausgeführt
0445 C4 02
0447 07 SI → 2. Schritt 30 wird ebenfalls ausgeführt!
0448 8F FF
044A C4 05 SI → 3. Schritt C4 02
044C 07
044D 8F FF SI → 4. Schritt 07, usw.
044F 90 F4

Das Programm funktioniert mit: 551 = XRI 04 036F = E4 04



Programm: Betriebsprogramm, —
UP „SINGLE-STEP“

Blatt 14

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen	
569		LD	01 (P1)	038B	C1	01	Scrpd. (PC) LOB → Scrpd. (PX) LOB	SINGLE-STEP Single-Byte-Befehle
570		ST	(EX) (P1)	D	C9	80		
571		JMP	NEXT ↑	F	90	CF	Rücksprung nach 543	
572	XPAL 0	LD	08 (P1)	0391	C1	08	Scrpd. (AC) → (P2) LOB	
573		XPAL 2		3	32			
574		LD	01 (P1)	4	C1	01	Scrpd. (PC) LOB → Scrpd. (AC)	
575		ST	08 (P1)	6	C9	08		
576		JMP	NEXT ↑	8	90	C6	Rücksprung nach 543	
577	XPAH 0	LD	08 (P1)	A	C1	08	Scrpd. (AC) → (P2) HOB	
578		XPAH 2		C	36			
579		LD	00 (P1)	D	C1	00	Scrpd. (PC) HOB → Scrpd. (AC)	
580		ST	08 (P1)	F	C9	08		
581	PC NEU	XPAH 2		03A1	36		(P2) HOB → (0400) } (P2) → Scrpd. (P2) LOB → (0401) } für PC	
582		ST	00 (P1)	2	C9	00		
583		XPAL 2		4	32			
584		ST	01 (P1)	5	C9	01		
585		JMP	RÜCK ↓	7	90	50	Rücksprung nach PH 1	
586	DI/OP	LD	@ 01 (P2)	9	C6	01	2. Byte des Befehls laden und in 040C abspeichern, (P2) increm.	SINGLE-STEP
587		ST	0C (P1)	B	C9	0C		
588	SCR 1	XPAH 2		D	36		(P2) HOB → (0400)	
589		ST	00 (P1)	E	C9	00		
590		XPAL 2		03B0	32		(P2) LOB → (0401)	
591		ST	01 (P1)	1	C9	01		
592		LD	06 (P1)	3	C1	06	(0406) → (P3) HOB	
593		XPAH 3		5	37			
594		LD	07 (P1)	6	C1	07	(0407) → (P3) LOB	
595		XPAL 3		8	33			
596		LD	04 (P1)	9	C1	04	(0404) → (P2) HOB	
597		XPAH 2		B	36			
598		LD	05 (P1)	C	C1	05	(0405) → (P2) LOB	
599		XPAL 2		E	32			
600		LD	02 (P1)	F	C1	02	(0402) → (P1) HOB	
601		XPAH 1		03C1	35			
602		LD	40 (P0)	2	C0	40	(0403) → (P1) LOB	
603		XPAL 1		4	31			
604		LD	44 (P0)	5	C0	44	(040A) → (EX)	
605		XAE		7	01			
606		LD	40 (P0)	8	C0	40	(0409) → (ST)	
607		CAS		A	07			
608		LD	3C (P0)	B	C0	3C	(0408) → (AC)	
609		JMP	SCR 2 ↓	D	90	3C	Sprung nach 620	
610		NOP		F	08			
611	BRPNT	ST	37 (P0)	03D0	C8	37	(AC) → (0408)	BRPNT
612		CSA		2	06		(ST) → (0409)	
613		ST	35 (P0)	3	C8	35		

Programm: Betriebsprogramm, –
UP „SINGLE STEP“

Blatt 15

[illegible]