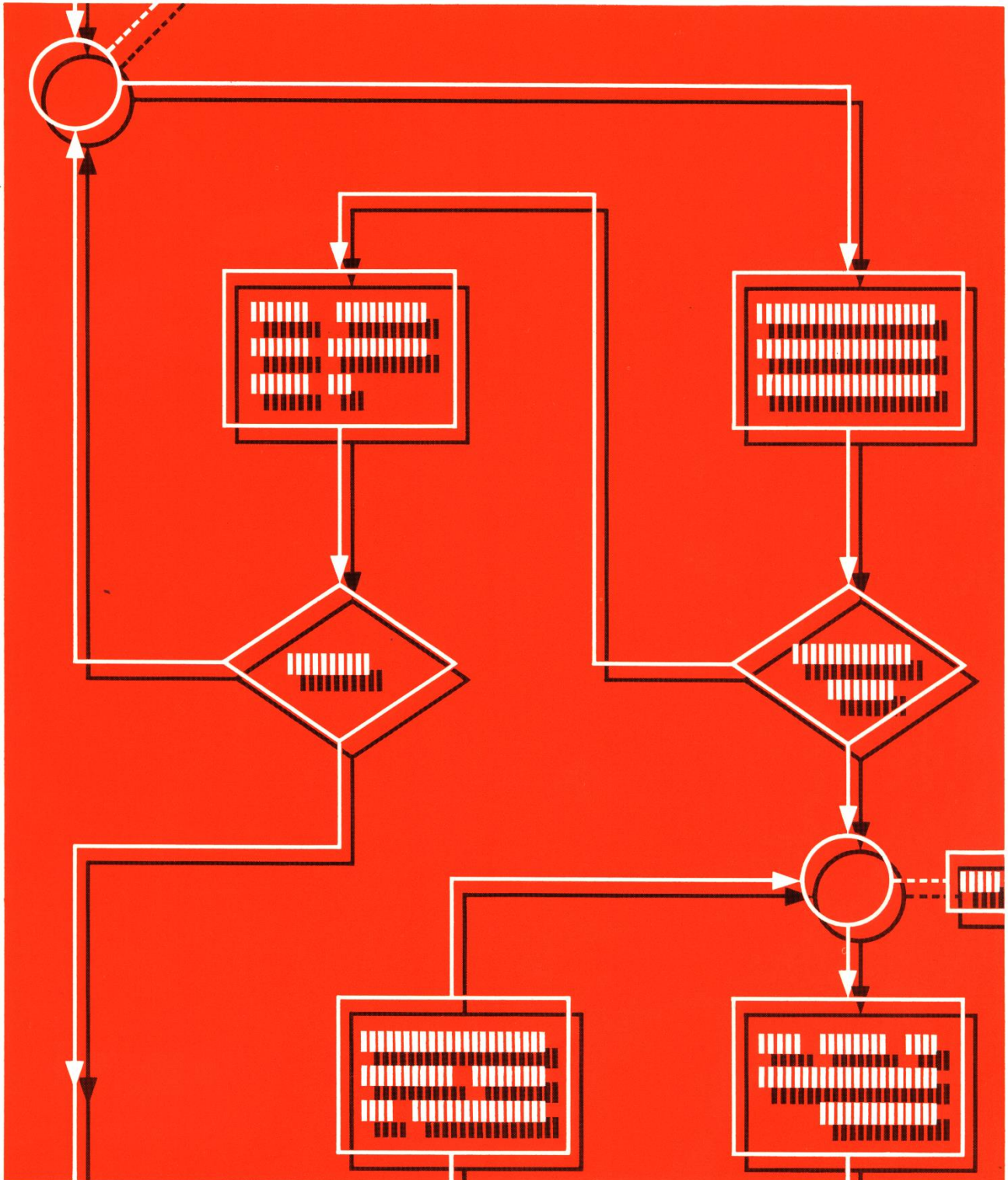


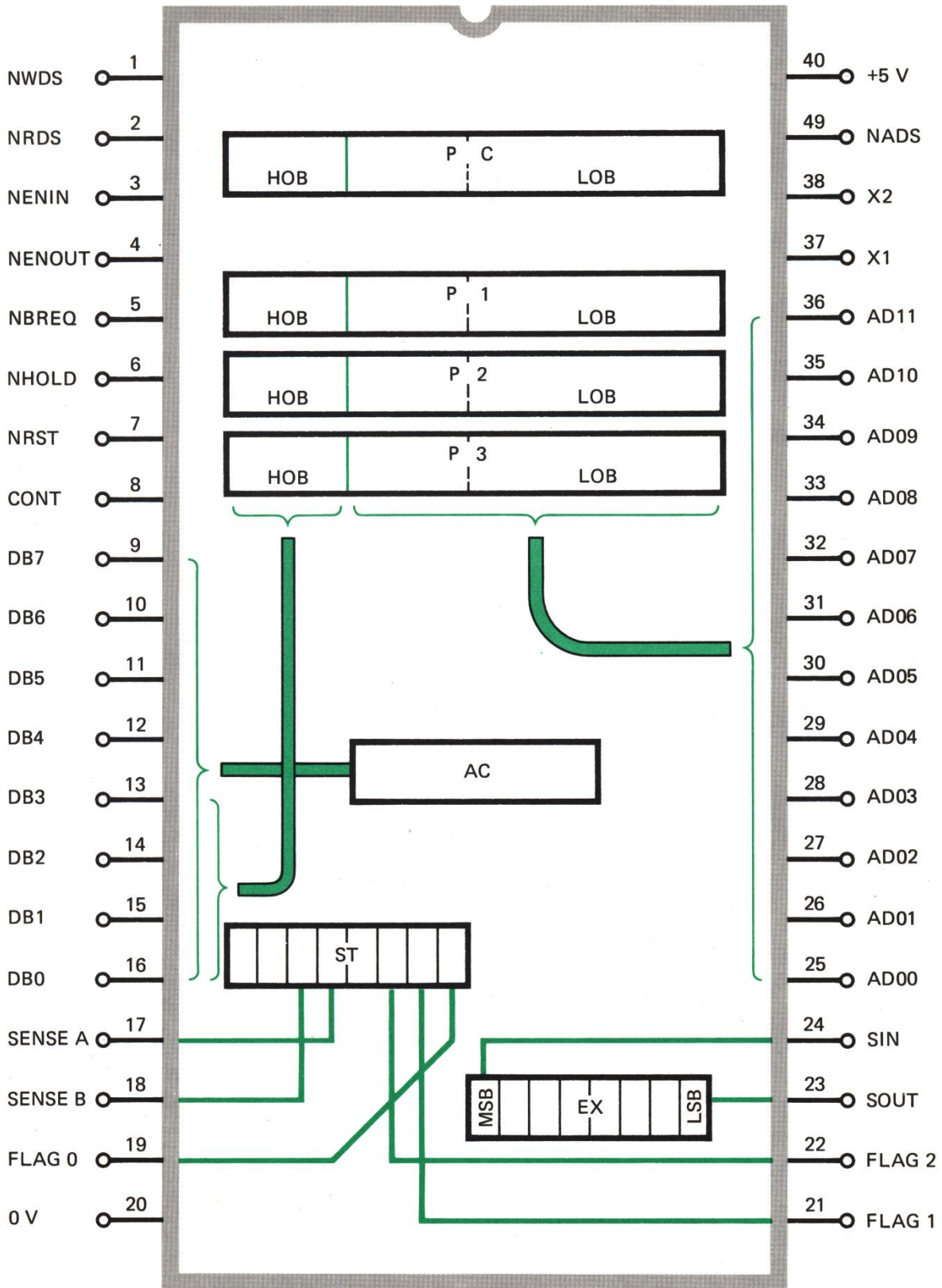
Mikroprozessor-Labor

Tabellen



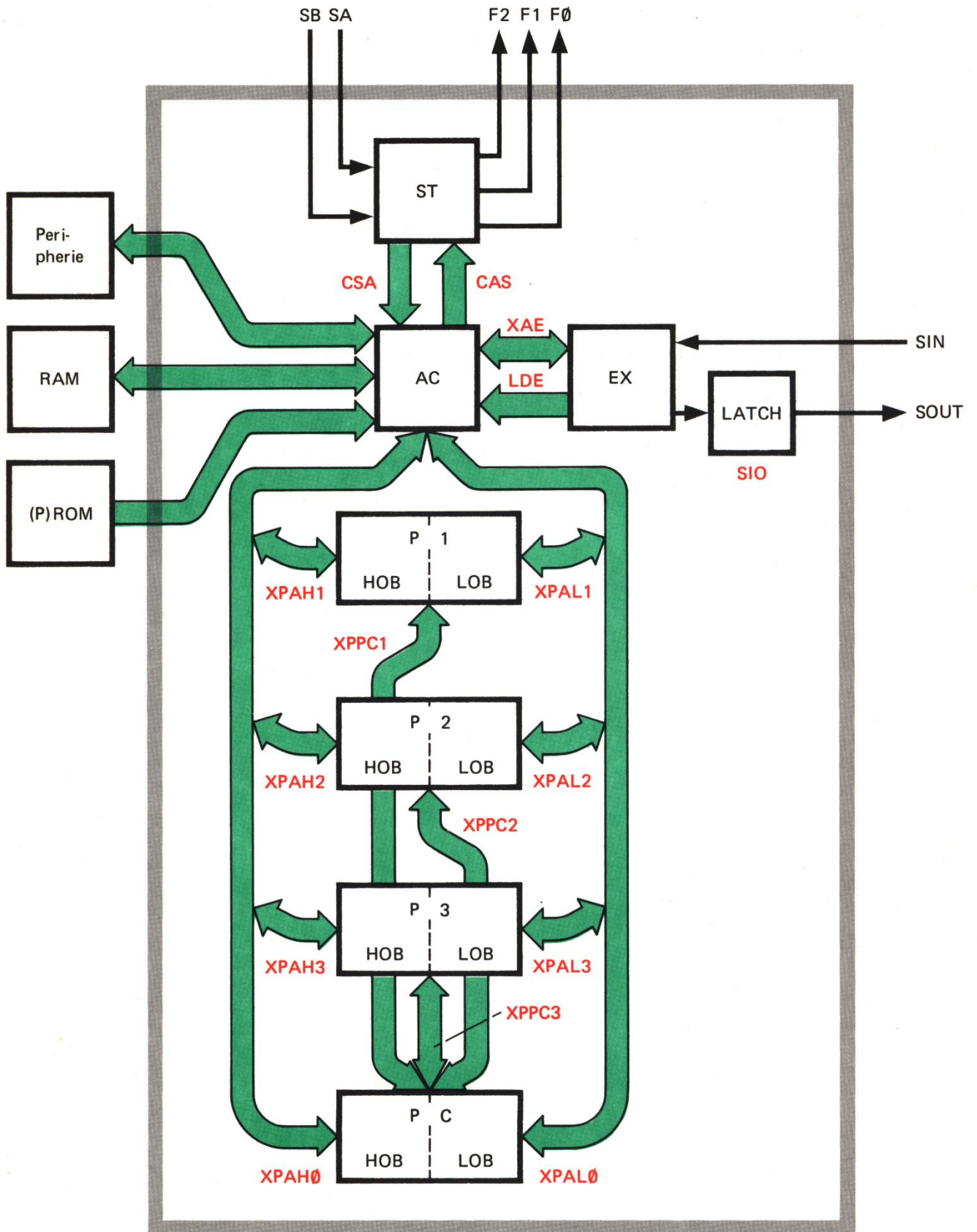


Anschlüsse des ISP-8A/600-Mikroprozessors (SC/MP-II)





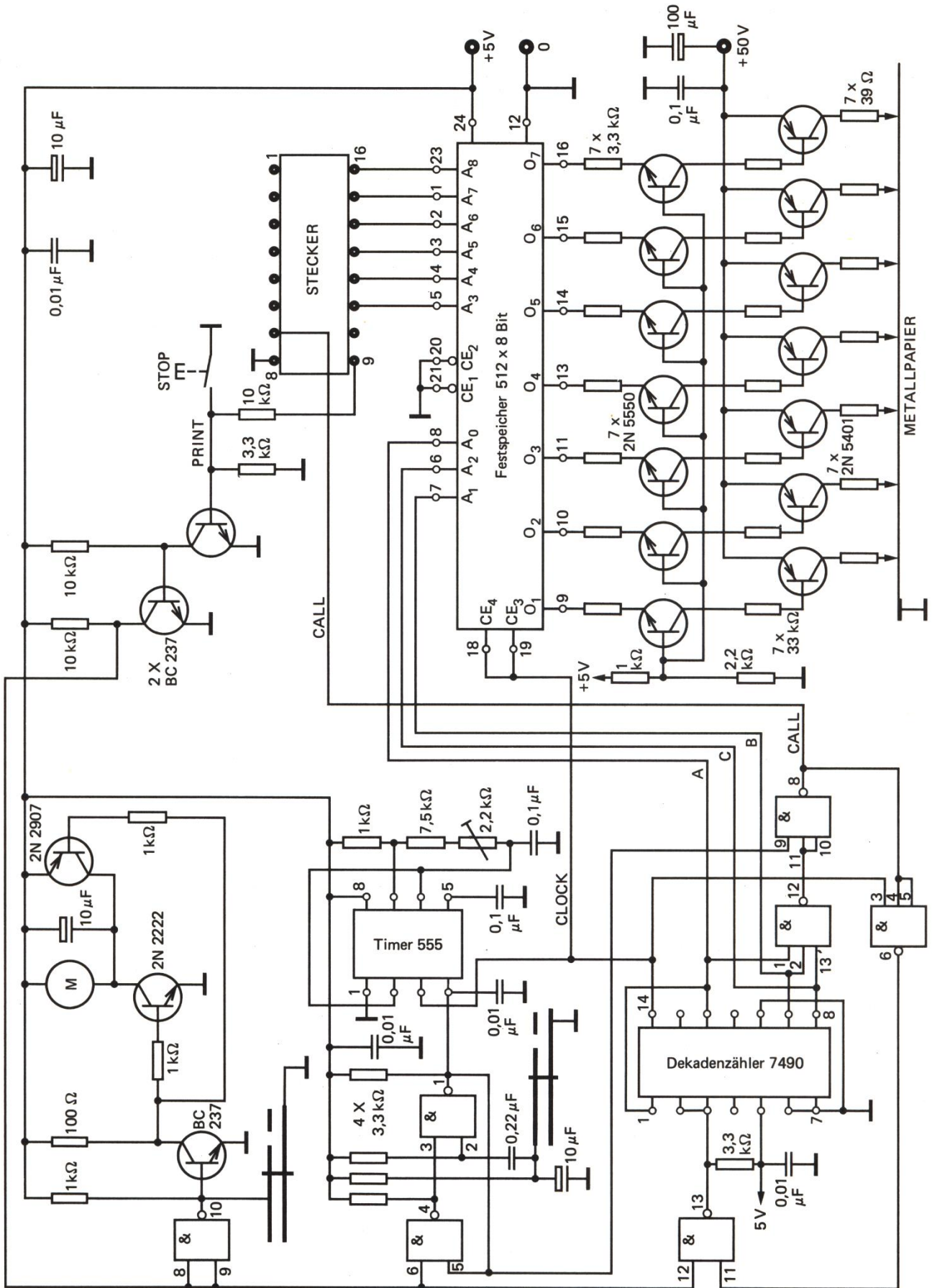
SC/MP-Architektur



T
2



SCHALTPLAN DES DRUCKERS





7-Bit-Code ASC II							
Steuerzeichen		Schriftzeichen		Schriftzeichen		Schriftzeichen	
Zeichen	Sedez.	Zeichen	Sedez.	Zeichen	Sedez.	Zeichen	Sedez.
NUL	00	SP	20	@	40	\	60
SOH	01	!	21	A	41	a	61
STX	02	"	22	B	42	b	62
ETX	03	#	23	C	43	c	63
EOT	04	\$	24	D	44	d	64
ENQ	05	%	25	E	45	e	65
ACK	06	&	26	F	46	f	66
BEL	07	'	27	G	47	g	67
BS	08	(28	H	48	h	68
HT	09)	29	I	49	i	69
LF	0A	*	2A	J	4A	j	6A
VT	0B	+	2B	K	4B	k	6B
FF	0C	,	2C	L	4C	l	6C
CR	0D	-	2D	M	4D	m	6D
SO	0E	.	2E	N	4E	n	6E
SI	0F	/	2F	O	4F	o	6F
DLE	10	0	30	P	50	p	70
DC ₁	11	1	31	Q	51	q	71
DC ₂	12	2	32	R	52	r	72
DC ₃	13	3	33	S	53	s	73
DC ₄	14	4	34	T	54	t	74
NAK	15	5	35	U	55	u	75
SYN	16	6	36	V	56	v	76
ETB	17	7	37	W	57	w	77
CAN	18	8	38	X	58	x	78
EM	19	9	39	Y	59	y	79
SUB	1A	:	3A	Z	5A	z	7A
ESC	1B	;	3B	[5B	{	7B
FS	1C	<	3C	\~	5C		7C
GS	1D	=	3D]	5D	}	7D
RS	1E	>	3E	↑{	5E	—	7E
US	1F	?	3F	—}	5F	DEL	7F

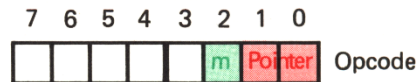
Die Bedeutung der Steuerzeichen wird auf einer anderen Tafel erläutert.

Die stark ausgedruckten Zeichen sind auf dem zum Lehrgang gehörenden Drucker verfügbar.



SC/MP-Befehlsliste

SPEICHER-BEZOGENE BEFEHLE



m = 0 : Pointer-indiziert
m = 1 : Autoindiziert

Mnemon. Code	Opcode Grundbef.	Operation	Mz	Seite
LD DISP	C0 DISP	(EA) → AC	18	S 58
ST DISP	C8 DISP	(AC) → EA	18	S 58
AND DISP	D0 DISP	(AC) ∧ (EA) → AC	18	S 171
OR DISP	D8 DISP	(AC) ∨ (EA) → AC	18	S 177
XOR DISP	E0 DISP	(AC) ⊕ (EA) → AC	18	S 178
DAD DISP	E8 DISP	(AC) ₁₀ + (EA) ₁₀ + (CY/L) → AC; (CY/L)	23	
ADD DISP	F0 DISP	(AC) + (EA) + (CY/L) → AC; (CY/L), (OV)	19	
CAD DISP	F8 DISP	(AC) + ~ (EA) + (CY/L) → AC; (CY/L), (OV)	20	

INKREMENT- U. DEKREMENT-BEFEHLE



Mnemon. Code	Opcode Grundbef.	Operation	Mz	Seite
ILD DISP	A8 DISP	(EA) + 1 → EA, AC	22	
DLD DISP	B8 DISP	(EA) - 1 → EA, AC	22	S 100

VERZÖGERUNGS-BEFEHL

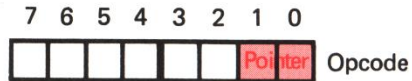
Mnemon. Code	Opcode	Operation	Mz	Seite
DLY DISP	8F DISP	(AC) nach - 1 ₁₀ gezählt $t_v = [2 \cdot (AC)_{10} + 514_{10} \cdot \text{Disp}_{10} + 13_{10}] \text{ Mz}$	13 bis 131593	S 120



SOFORT-BEFEHLE

Mnemon. Code	Opcode	Operation	Mz	Seite
LDI DISP	C4 DISP	DISP → AC	10	H 84
ANI DISP	D4 DISP	(AC) ^ DISP → AC	10	S 171
ORI DISP	DC DISP	(AC) v DISP → AC	10	S 177
XRI DISP	E4 DISP	(AC) ≠ DISP → AC	10	S 178
DAI DISP	EC DISP	(AC) ₁₀ + DISP ₁₀ + (CY/L) → AC; (CY/L)	15	
ADI DISP	F4 DISP	(AC) + DISP + (CY/L) → AC; (CY/L), (OV)	11	H 88
CAI DISP	FC DISP	(AC) + ~ DISP + (CY/L) → AC; (CY/L), (OV)	12	

SPRUNG-BEFEHLE



Mnemon. Code	Opcode Grundbef.	Operation	Mz	Seite
JMP DISP	90 DISP	(EA) → PC	11	S 77
JP DISP	94 DISP	Wenn (AC) = 00, (EA) → PC	9/11	S 92/111
JZ DISP	98 DISP	Wenn (AC) = 00, (EA) → PC	9/11	S 92/111
JNZ DISP	9C DISP	Wenn (AC) ≠ 00, (EA) → PC	9/11	S 92/111

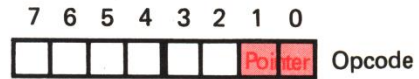
BEFEHLE MIT EXTENSION-REGISTER

Mnemon. Code	Opcode	Operation	Mz	Seite
LDE	40	(EX) → AC	6	H 64
XAE	01	(EX) ↔ (AC)	7	S 59
ANE	50	(EX) ^ (AC) → AC	6	S 171
ORE	58	(EX) v (AC) → AC	6	S 177
XRE	60	(EX) ≠ (AC) → AC	6	S 178
DAE	68	(EX) ₁₀ + (AC) ₁₀ + (CY/L) → AC; (CY/L)	11	
ADE	70	(EX) + (AC) + (CY/L) → AC; (CY/L), (OV)	7	
CAE	78	~ (EX) + (AC) + (CY/L) → AC; (CY/L), (OV)	8	

T
6



POINTER-AUSTAUSCH-BEFEHLE



Mnemon. Code	Opcode Grundbef.	Operation	Mz	Seite
XPAL	30	(AC) ↔ (P HIGH)	8	H 75
XPAH	34	(AC) ↔ (P LOW)	8	H 75
XPPC	3C	(P) ↔ (PC)	7	H 78

SCHIEBE- U. ROTATIONS-BEFEHLE, SERIELLER EIN- U. AUSGANG

Mnemon. Code	Opcode	Operation	Mz	Seite
SR	1C	(AC _i) → AC _{i-1} ; 0 → AC ₇	5	H 174
SRL	1D	(AC _i) → AC _{i-1} ; (CY/L) → AC ₇	5	
RR	1E	(AC _i) → AC _{i-1} ; (AC ₀) → AC ₇	5	H 176
RRL	1F	(AC _i) → AC _{i-1} ; (AC ₀) → CY/L; (CY/L) → AC ₇	5	
SIO	19	(EX _i) → EX _{i-1} ; SIN → EX ₇ ; (EX ₀) → SOUT	5	

VERSCHIEDENE 1-BYTE-BEFEHLE

Mnemon. Code	Opcode	Operation	Mz	Seite
HALT	00	H-Flag setzen	8	
CCL	02	0 → CY/L	5	
SCL	03	1 → CY/L	5	
DINT	04	0 → IE	6	S 195
IEN	05	1 → IE	6	S 195
CSA	06	(ST) → AC	5	S 84
CAS	07	(AC) → ST	6	H 67
NOP	08	(PC) + 1 → PC	5	S 161



ALPHABETISCHE BEFEHLSLISTE

Mnemon. Code	Opcode Grundbef.	Assembl. Format	Operation (englisch)	Mz	Seite
ADD	F0	@ DISP(P)	Add	19	
ADE	70		Add Extension	19	
ADI	F4	DISP	Add Immediate	11	H 88
AND	D0	@ DISP(P)	AND	18	
ANE	50		AND Extension	6	
ANI	D4	DISP	AND Immediate	10	
CAD	F8	@ DISP(P)	Complement and Add	20	
CAE	78		Complement and Add Extension	8	
CAI	FC	DISP	Complement and Add Immediate	12	
CAS	07		Copy AC to Status	6	H 67
CCL	02		Clear Carry/Link	5	
CSA	06		Copy Status to AC	5	S 84
DAD	E8	@ DISP(P)	Decimal Add	23	
DAE	68		Decimal Add Extension	11	
DAI	EC	DISP	Decimal Add Immediate	15	
DINT	04		Disable Interrupts	6	
DLD	B8	DISP(P)	Decrement and Load	22	S 100
DLY	8F	DISP	Delay	13-131593	S 120
HALT	00		H-FLAG	8	
IEN	05		Enable Interrupts	6	
ILD	A8	DISP(P)	Increment and Load	22	S 166
JMP	90	DISP(P)	Jump	11	S 77
JNZ	9C	DISP(P)	Jump if not Zero	9/11	S 92/111
JP	94	DISP(P)	Jump if Positive	9/11	S 92/111
JZ	98	DISP(P)	Jump if Zero	9/11	S 92/111
LD	C0	@ DISP(P)	Load	18	S 58
LDE	40		Load from Extension	6	H 64
LDI	C4	DISP	Load Immediate	10	H 84
NOP	08		No Operation	5	S 161
OR	D8	@ DISP(P)	OR	18	
ORE	58		OR Extension	6	
ORI	DC	DISP	OR Immediate	10	
RR	1E		Rotate Right	5	H 176
RRL	1F		Rotate Right with CY/L	5	
SCL	03		Set CY/L	5	
SIO	19		Serial Input/Output	5	
SR	1C		Shift Right	5	H 174
SRL	1D		Shift Right with CY/L	5	
ST	C8	@ DISP(P)	Store	18	S 58
XAE	01		Exchange AC and Extension	7	S 59
XOR	E0	@ DISP(P)	Exclusive OR	18	
XPAH	34	P	Exchange Pointer with AC (High)	8	H 75
XPAL	30	P	Exchange Pointer with AC (Low)	8	H 75
XPPC	3C	P	Exchange Pointer with PC	7	H 78
XRE	60		Exclusive OR Extension	6	
XRI	E4	DISP	Exclusive OR Immediate	10	

Programm: Verschieben eines Datenblocks um eine Speicherzelle

Die ersten 7 Bytes des ursprünglichen Programms müssen nach dem Verschieben neu geladen werden, wenn das ursprüngliche Programm bei 0420 beginnt.

Nr.	Label	Operation	Operand	Adresse	Op-code	Displacement	Bemerkungen
	STACK			041C			HOB } ANFANGSADRESSE LOB }
				1D			
				1E			HOB } ENDADRESSE + 1 LOB }
				1F			
1	ANSPR	LDI	05	0420	C4	05	ANSPRUNG DES SCHIEBE-PROGRAMMS
2		XPAH 2		22	36		
3		LDI	A8	23	C4	A8	
4		XPAL 2		25	32		
5		XPPC 2		26	3E		
1	START	LDI	04	05A9	C4	04	(P1) = 0420
2		XPAH 1		AB	35		
3		LDI	20	AC	C4	20	
4		XPAL 1		AE	31		
5		LD	-2(P1)	AF	C1	FE	(P2) = ENDADRESSE + 1
6		XPAH 2		B1	36		
7		LD	-1(P1)	B2	C1	FF	
8		XPAL 2		B4	32		
9	SHIEB	LD	@ FF(P2)	B5	C6	FF	EIN BYTE UM 1 SCHIEBEN, DECR. (P2)
10		ST	01(P2)	B7	CA	01	
11		XPAH 2		B9	36		HOB (P2)
12		XAE		BA	01		= HOB ANFANGSADRESSE ?
13		LDE		BB	40		
14		XPAH 2		BC	36		
15		LDE		BD	40		
16		XOR	-4(P1)	BE	E1	FC	
17		JNZ	SHIEB	C0	9C	F3	NEIN: NÄCHSTES BYTE SCHIEBEN!
18		XPAL 2		C2	32		LOB(P2)
19		XAE		C3	01		= LOB ANFANGSADRESSE?
20		LDE		C4	40		
21		XPAL 2		C5	32		
22		LDE		C6	40		
23		XOR	-3(P1)	C7	E1	FD	
24		JNZ	SHIEB	C9	9C	EA	NEIN: NÄCHSTES BYTE SCHIEBEN!
25	RETUR	ST	00(P2)	CB	CA	00	JA: LETZTES BYTE = 00
27		XPAH 2		CD	36		(P2) = 0000, ANF. BETRIEBSPRGRM.
28		LDI	00	CE	C4	00	
29		XPAL 2		D0	32		
30		ILD	-3(P1)	D1	A9	FD	NEUE ANFANGSADRESSE → STACK
31		JNZ	LOBEN	D3	9C	02	
32		ILD	-4(P1)	D5	A9	FC	
33	LOBEN	ILD	-1(P1)	D7	A9	FF	NEUE ENDADRESSE → STACK
34		JNZ	STOP	D9	9C	02	
35		ILD	-2(P1)	DB	A9	FE	
36	STOP	JMP	00(P2)	DD	92	00	(PC) = 0420, STOP



BETRIEBSPROGRAMM: EINGABE

