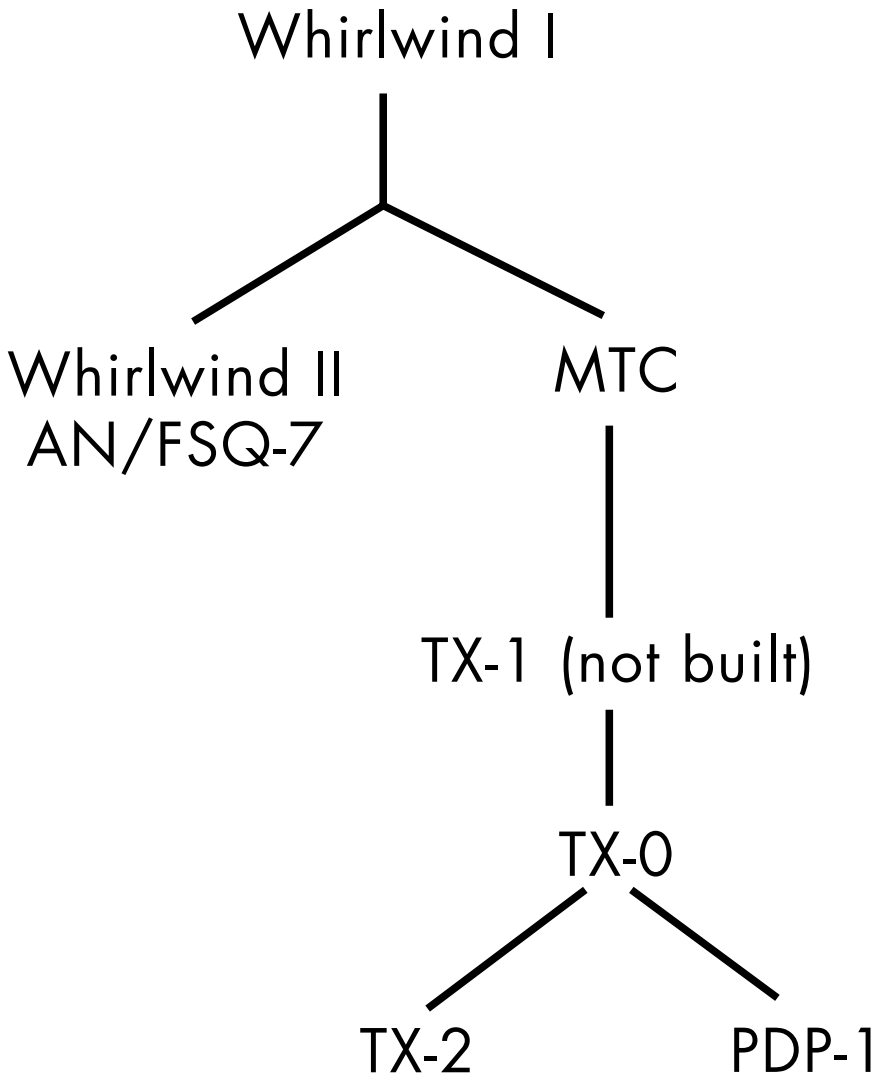
Family tree



PDP-1 —— (PDP-2,3)

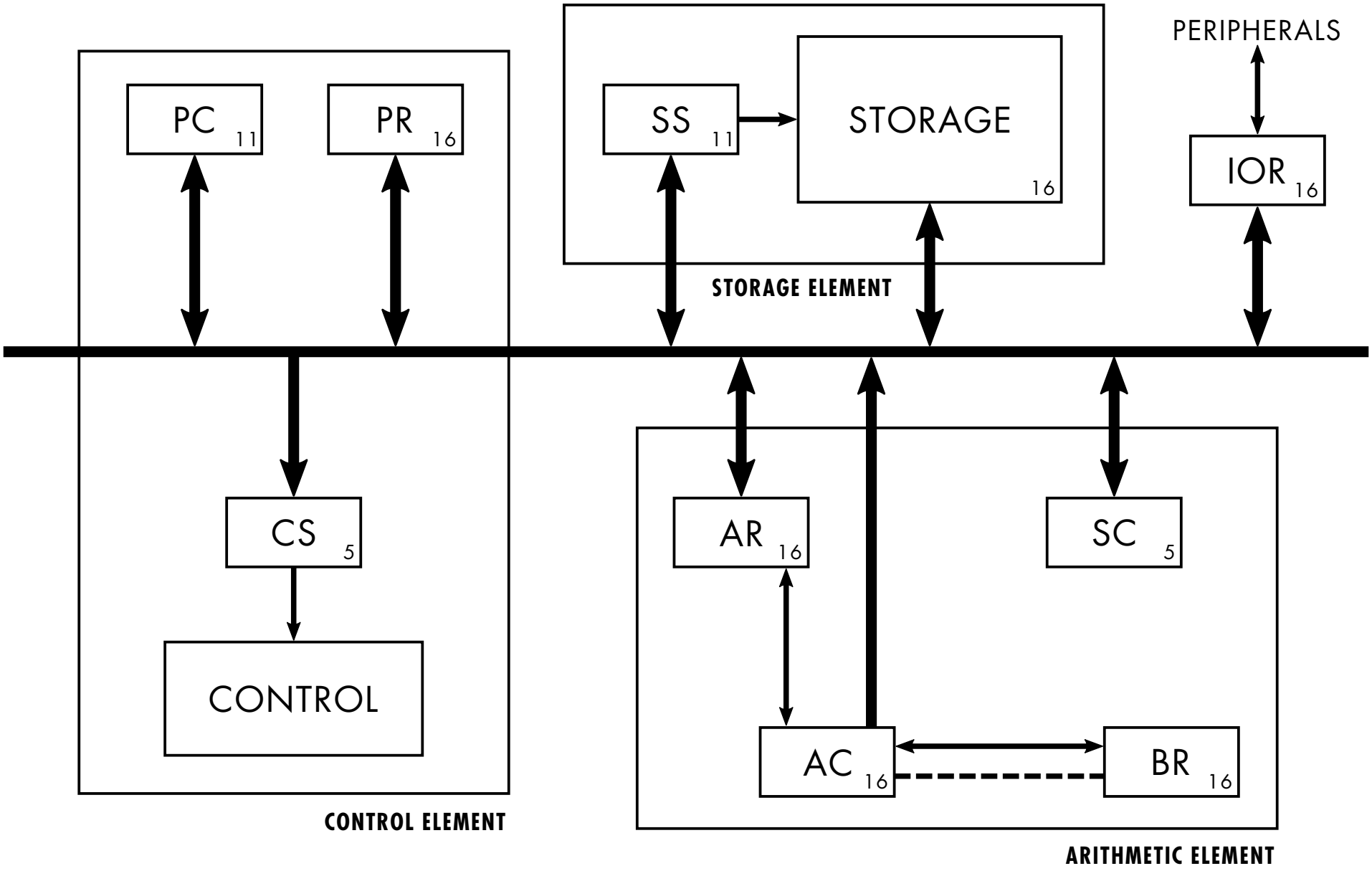
PDP-4 —— PDP-7,9,15

PDP-5 —— PDP-8,12

PDP-6 —— PDP-10

PDP-11

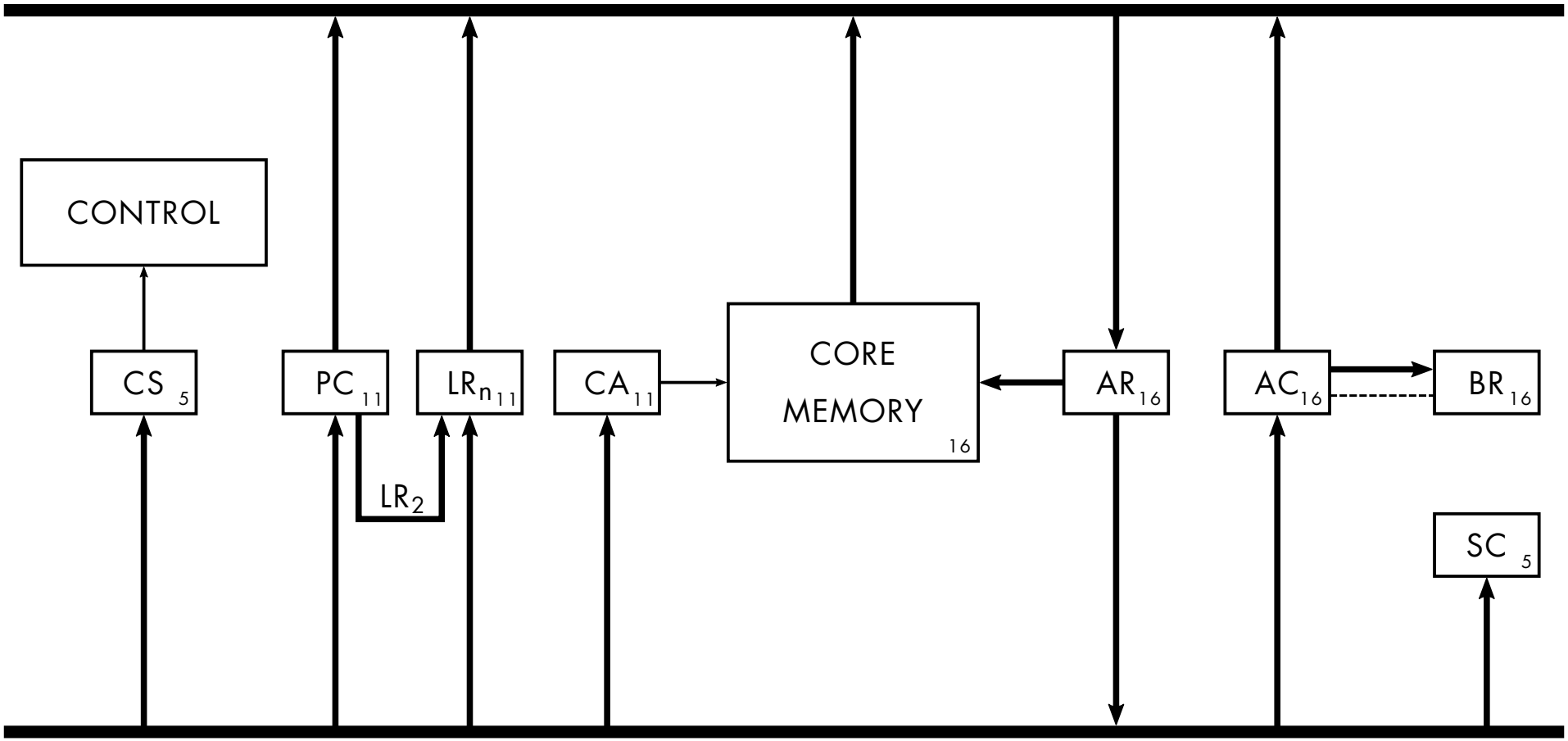
Whirlwind I block diagram



Whirlwind I summary

INSTRUCTION			AC	BR	AR	SAM	C(x)
00000	SI	SELECT IO UNIT					
00001		ILLEGAL					
00010	BI	BLOCK IN	x + n		x		first word
00011	RD	READ	IOR		IOR		
00100	BO	BLOCK OUT	x + n		x		
00101	RC	RECORD					
00110	SD	SUM OF DIGITS	AC ∨ C(x)		C(x)	0	
00111	CF	CHANGE FIELDS					
01000	TS	TRANSFER TO STORAGE					AC
01001	TD	TRANSFER DIGITS					$C(x)_{0\text{-}4}$,$AC_{5\text{-}15}$
01010	TA	TRANSFER ADDRESS					$C(x)_{0\text{-}4}$,$AR_{5\text{-}15}$
01011	CK	CHECK					
01100	AB	ADD BR	BR + C(x)		C(x)	0	BR + C(x)
01101	EX	EXCHANGE	C(x)		C(x)		AC
01110	CP	CONDITIONAL SP			y+1		
01111	SP	SUBPROGRAM			y+1		
10000	CA	CLEAR, ADD	C(x) + SAM	0	C(x)	0	
10001	CS	CLEAR, SUBTRACT	–C(x) + SAM	0	C(x)	0	
10010	AD	ADD	AC + C(x)		C(x)	0	
10011	SU	SUBTRACT	AC – C(x)		C(x)	0	
10100	CM	CLEAR, ADD MAG.	|C(x)|+SAM	0	|C(x)|	0	
10101	SA	SPECIAL ADD	AC + C(x)		C(x)	±1 or 0	
10110	AO	ADD ONE	C(x) + 1		C(x)	0	C(x) + 1
10111	DM	DIFFERENCE OF MAG.	|AC|–|C(x)|	AC	|C(x)|	0	
11000	MR	MULTIPLY, ROUND	$(AC \times C(x))_{LT}$ + r	0	|C(x)|	0	
11001	MH	MULTIPLY, HOLD	AC × C(x)	←	|C(x)|	0	
11010	DV	DIVIDE	±0	AC/C(x)	|C(x)|	0	
11011 0	SLR	SHIFT LEFT, ROUND	$(AC{:}BR \ll n)_{LT}$ + r	0		0	
11011 1	SLH	SHIFT LEFT, HOLD	AC:BR « n	←		0	
11100 0	SRR	SHIFT RIGHT, ROUND	$(AC{:}BR \gg n)_{LT}$ + r	0		0	
11100 1	SRH	SHIFT RIGHT, HOLD	AC:BR » n	←		0	
11101	SF	SCALE FACTOR	AC:BR « n	←	n	0	n
11110 0	CLC	CYCLE LEFT, CLEAR	$(AC{:}BR \text{ rot } n)_{LT}$	0			
11110 1	CLH	CYCLE LEFT, HOLD	AC:BR rot n	←			
11111	MD	MULTIPLY DIGITS	AC ∧ C(x)		–(final AC)		

MTC block diagram



MTC proposals

March 1952

M-1428

4 operations

ts
su
cp
io

8 operations

tc	cr
td	cp
ad	mr
su	io

16 operations

ts	tc
td	dc
ad	su
cr	sr
sp	cp
mr	lm
qr	qr*
qd	qh

tc: ts, clear AC
dc: td, clear AC

July 1952

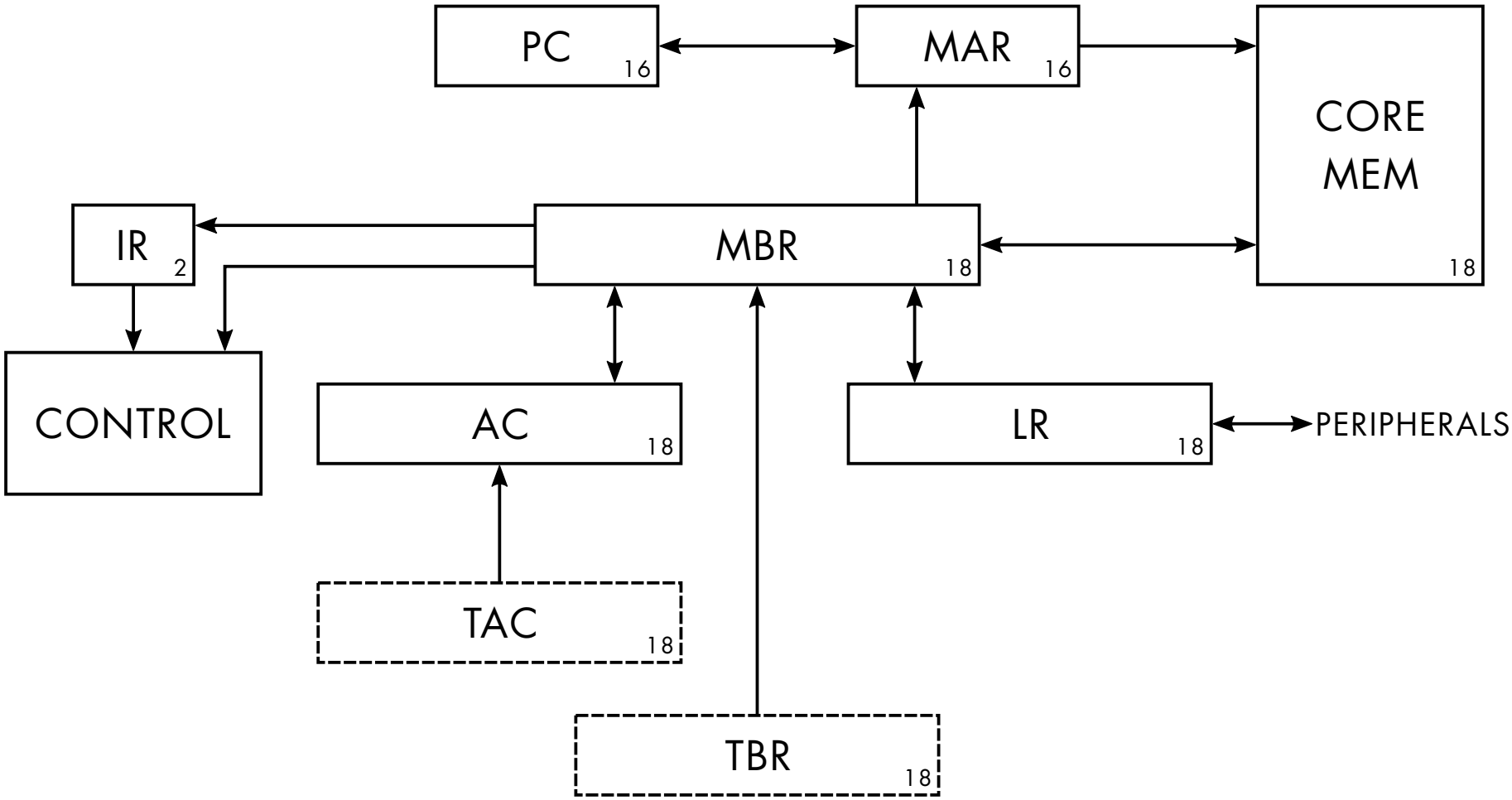
M-1562

0	1	2	3	4	5
1: Storage	1: In	1: 10 digits		1: Clear AC	
		0: 16 digits			
	0: Out	1: Comp. AR	1: Display	0:	
		0:	0: Add		
0: No Storage	1: Sub-program	1: cp	1: Alarm		
		0: sp	0:		
	0:	1: PETR	1: punch	1: sr	1: Shift
		0:	0:	0: cr	0:

Final

32 separate orders, similar to Whirlwind I

TX-0 block diagram



TX-0

1958

4k memory

2 3 4

0 1	000	001	010	011	100	101	110	111
00	STO							
01	ADD							
10	TRN							
11	OPR - See other tables							

6 7 8

4 5	000	001	010	011	100	101	110	111
00	NOP							
01	EX0	EX1	EX2	EX3	EX4	EX5	EX6	EX7
10		R1L	DIS	R3L	PNT		P6H	P7H
11	HLT							

	2	3	9	10	11	12	13	14	15	16	17
7											
8	1 CLL	1 CLR									
1					1 PEN				0		
					0				1 TAC		
2						1 COM				0 AMB 1	
										1 TBR 1	
3			0 MBL 1							1 LMB 0	
4			1 SHR 0				1 PAD				
			1 CYR 1								
7								1 CRY			

TX-0

July 1960

8k memory

0 1 \ 2 3 4	000	001	010	011	100	101	110	111
00	STO				SLR			
01	ADD				LLR			
10	TRN				TRA			
11	OPR - See other tables							

4 5 \ 6 7 8	000	001	010	011	100	101	110	111
00	NOP							
01	EX0	EX1	EX2	EX3	EX4	EX5	EX6	EX7
10		R1L	DIS	R3L	PNT		P6H	P7H
11	HLT							

	2	3	9	10	11	12	13	14	15	16	17
7											
8	1 CLL	1 CLR									
1					1 PEN				0		
					0				1 TAC		
2						1 COM				0 AMB 1	
										1 TBR 1	
3			0 MBL 1							1 LMB 0	
			0 0 ORL 1						1		
			0 ANL 1 1						1		
4			1 SHR 0				1 PAD				
			1 CYR 1								
7								1 CRY			

TX-0

June 1961

8k memory

2 3 4

0 1	000	001	010	011	100	101	110	111
00	STO				SLR			
01	ADD				LLR			
10	TRN				TRA			
11	OPR - See other tables							

6 7 8

4 5	000	001	010	011	100	101	110	111
00	NOP	TAC	TBR	PEN	SEL		RPF	SPF
01	EX0	EX1	EX2	EX3	EX4	EX5	EX6	EX7
10	CPY	R1L	DIS	R3L	PRT	TYP	P6H	P7H
11	HLT	CLL	CLR					

	2	3	9	10	11	12	13	14	15	16	17
7		1 AMB									
8	1 CLA										
2						1 COM					
3									1 ORB	0	1
									1 ANB	1	1
4			0 MBL	1	X				0 LMB	1	X
5			1 SHR	0	0		1 PAD				
			1 CYR	1	0						
7								1 CRY			

TX-0

September 1965

8k memory

2 3 4

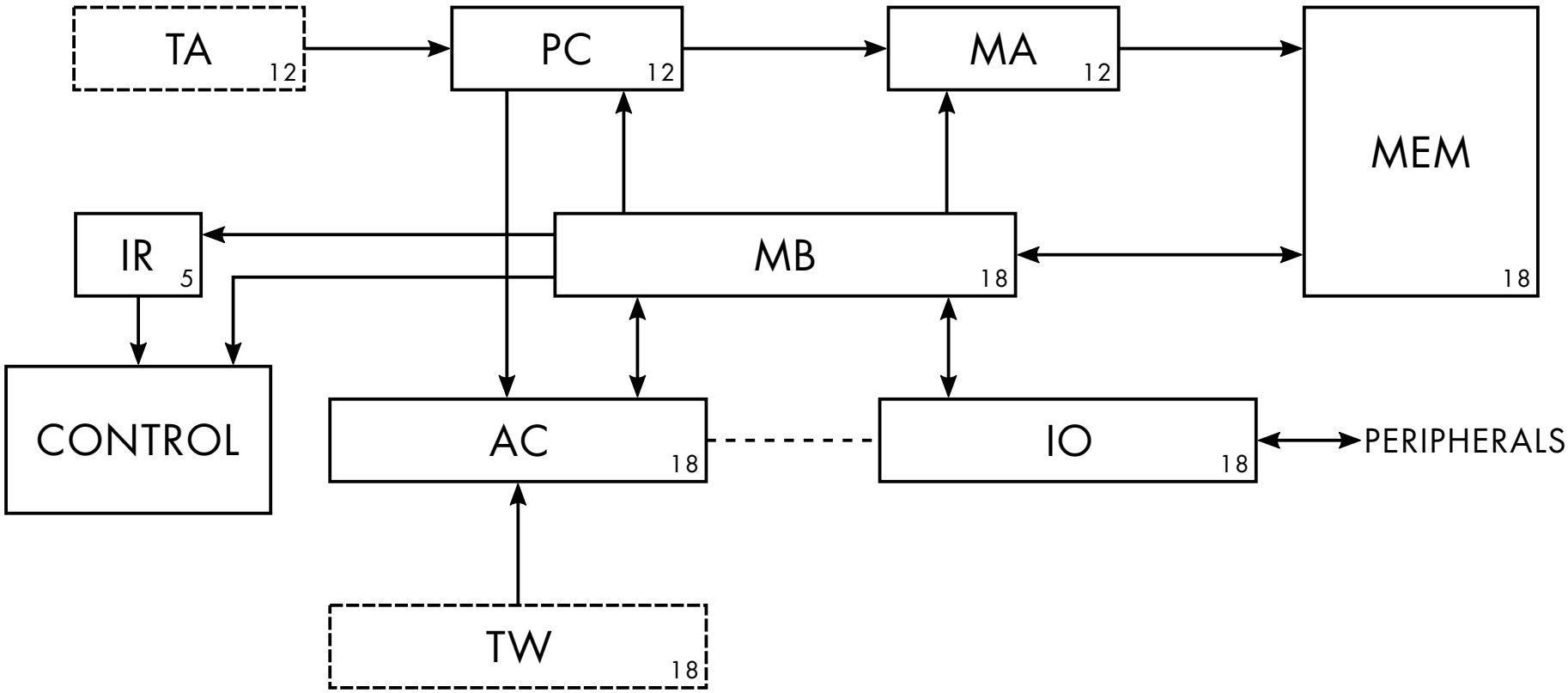
0 1	000	001	010	011	100	101	110	111
00	STO	STX	SXA	ADO	SLR	SLX	STZ	
01	ADD	ADX	LDX	AUX	LLR	LLX	LDA	LAX
10	TRN	TZE	TAX	TIX	TRA	TRX	TLV	(TPL)
11	OPR - See other tables							

6 7 8

4 5	000	001	010	011	100	101	110	111
00	NOP	TAC	TBR	PEN	SEL		RPF	SPF
01	EX0	EX1	EX2	EX3	EX4	EX5	EX6	EX7
10	CPY	R1L	DIS	R3L	PRT	TYP	P6H	P7H
11	HLT	CLL	CLR					

	2	3	9	10	11	12	13	14	15	16	17
7		1 AMB									
8	1 CLA										
2			0 XMB	X	1	1 COM					
3									1 ORB	0	1
									1 ANB	1	1
4			0 MBL	1	X				0 LMB	1	X
5							1 PAD				
6			1 SHR	0	0						
			1 CYR	1	0						
7								1 CRY			
8									0 MBX	X	1

PDP-1 block diagram



PDP-1 summary

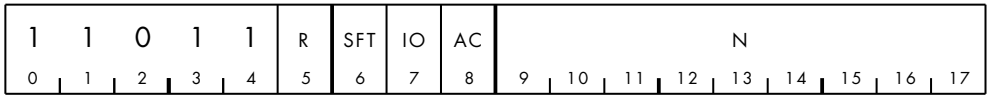
MAIN INSTRUCTIONS

00		
02 AND	AC ← AC ∧ C(Y)	AND
04 IOR	AC ← AC ∨ C(Y)	INCLUSIVE OR
06 XOR	AC ← AC ⊻ C(Y)	EXCLUSIVE OR
10 XCT	execute C(Y)	EXECUTE
12 JFD		JUMP TO FIELD
14		
16 CAL	JDA 100	CALL
17 JDA	C(Y) ← AC. AC ← PC. PC ← Y+1	JUMP, DEPOSIT AC
20 LAC	AC ← C(Y)	LOAD AC
22 LIO	IO ← C(Y)	LOAD IO
24 DAC	C(Y) ← AC	DEPOSIT AC
26 DAP	$C(Y)_{6\text{-}17}$ ← AC	DEPOSIT ADDR. PART
30 DIP	$C(Y)_{0\text{-}5}$ ← AC	DEPOSIT INST. PART
32 DIO	C(Y) ← IO	DEPOSIT IO
34 DZM	C(Y) ← 0	DEPOSIT ZERO
36		
40 ADD	AC ← AC + C(Y)	ADD
42 SUB	AC ← AC - C(Y)	SUBTRACT
44 IDX	C(Y), AC ← C(Y)+1	INDEX
46 ISP	IDX. AC ≥ 0: SKIP	INDEX, SKIP IF POSITIVE
50 SAD	AC ≠ C(Y): SKIP	SKIP IF DIFFERENT
52 SAS	AC = C(Y): SKIP	SKIP IF SAME
54 MUS		MULTIPLY STEP
56 DIS		DIVIDE STEP
60 JMP	PC ← Y	JUMP
62 JSP	AC ← PC. PC ← Y	JUMP, SAVE PC
64 SKP		SKIP GROUP
66 SFT		SHIFT GROUP
70 LAW	AC ← Y/~Y	LOAD AC WITH
72 IOT		IO TRANSFER
74		
76 OPR		OPERATE

SKIP GROUP

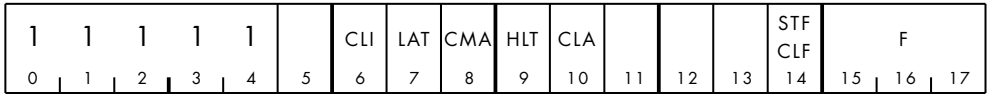
650000 SKP	reverse SKIP condition	SKIP
642000 SPI	IO_0 = 0: SKIP	SKIP ON PLUS IO
641000 SZO	OV = 0: SKIP	SKIP ON ZERO OV
640400 SMA	AC_0 = 1: SKIP	SKIP ON MINUS AC
640200 SPA	AC_0 = 0: SKIP	SKIP ON PLUS AC
640100 SZA	AC = +0: SKIP	SKIP ON ZERO AC
6400X0 SZS	SW_n = 0: SKIP	SKIP ON ZERO SWITCH
64000X SZF	F_n = 0: SKIP	SKIP ON ZERO FLAG

SHIFT GROUP



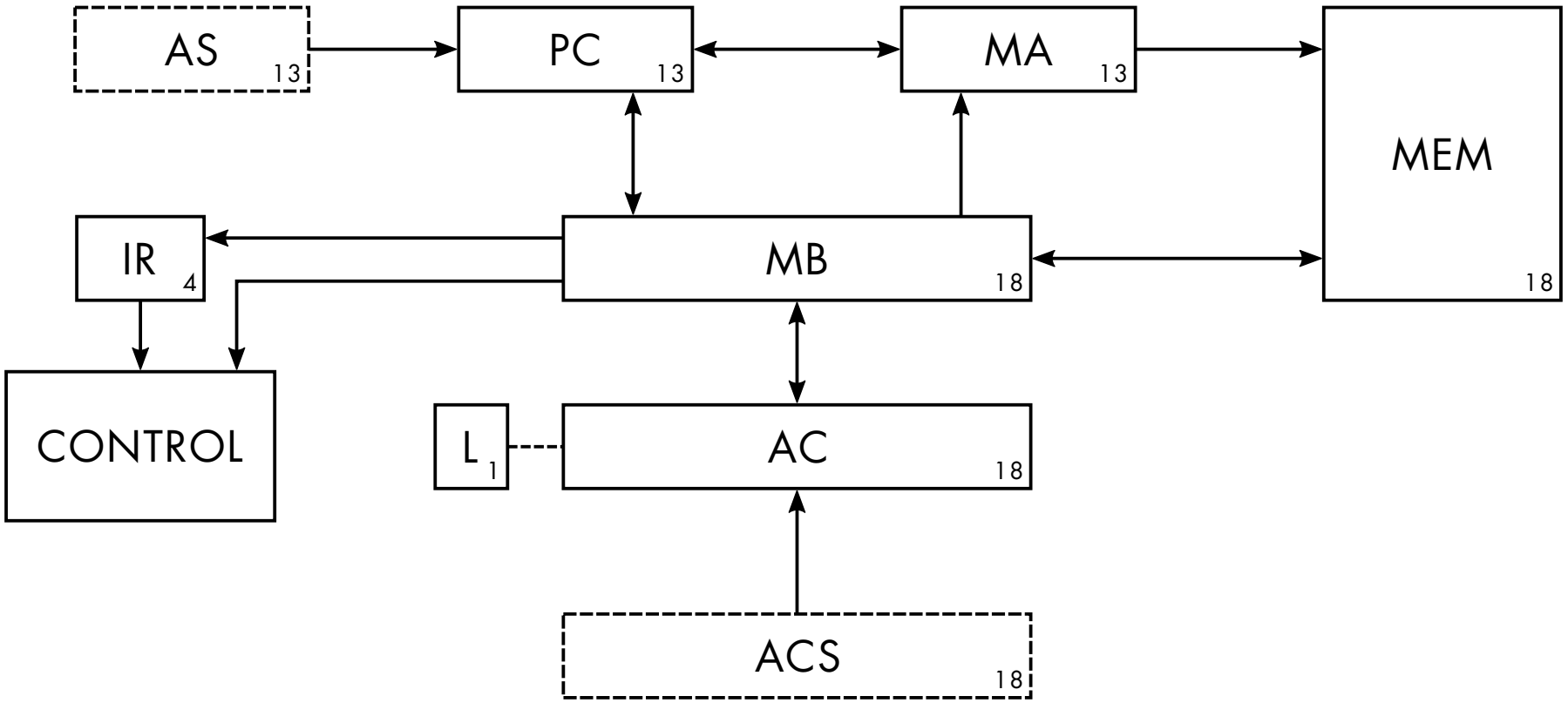
SFT: 0: rotate (R__). 1: shift (S__)
IO:AC: 01: AC (_A_)
10: IO (_I_)
11: AC:IO (_C_)
R: 0: left (__L). 1: right (__R)
N: steps (number of 1's)

OPERATE GROUP



764000 CLI	IO ← 0	CLEAR IO
762000 LAT	AC ← AC ∨ TW	LOAD AC FROM TW
761000 CMA	CMA ← ~CMA	COMPLEMENT AC
760400 HLT	HALT	HALT
760200 CLA	AC ← 0	CLEAR AC
760100 LAP	AC ← AC ∨ PC	LOAD AC FROM PC
76000X CLF	Fn ← 0	CLEAR FLAG
76001X STF	Fn ← 1	SET FLAG

PDP-4 block diagram

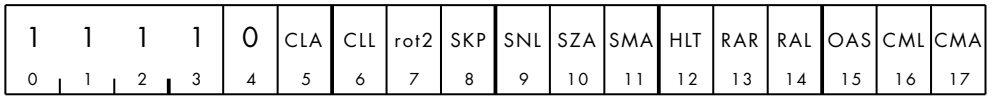


PDP-4 summary

MAIN INSTRUCTIONS

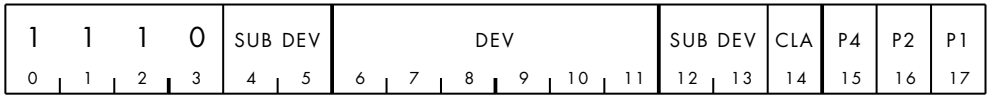
00 CAL	JMS 20	CALL
04 DAC	C(Y) ← AC	DEPOSIT AC
10 JMS	C(Y) ← PC. PC ← Y+1	JUMP TO SUBROUTINE
14 DZM	C(Y) ← 0	DEPOSIT ZERO
20 LAC	AC ← C(Y)	LOAD AC
24 XOR	AC ← AC ⊻ C(Y)	XOR
30 ADD	AC ← AC + C(Y)	ONE'S COMP. ADD
34 TAD	L:AC ← AC + C(Y)	TWO'S COMP. ADD
40 XCT	execute C(Y)	EXECUTE
44 ISZ	C(Y) ← C(Y)+1. C(Y) = 0: SKIP	INDEX, SKIP IF ZERO
50 AND	AC ← AC ∧ C(Y)	AND
54 SAD	AC ≠ C(Y): SKIP	SKIP IF AC DIFFERENT
60 JMP	PC ← Y	JUMP
64		
70 IOT		IO TRANSFER
74 OPR		OPERATE
76 LAW	AC ← inst. word	LOAD AC WITH INST.

OPERATE GROUP

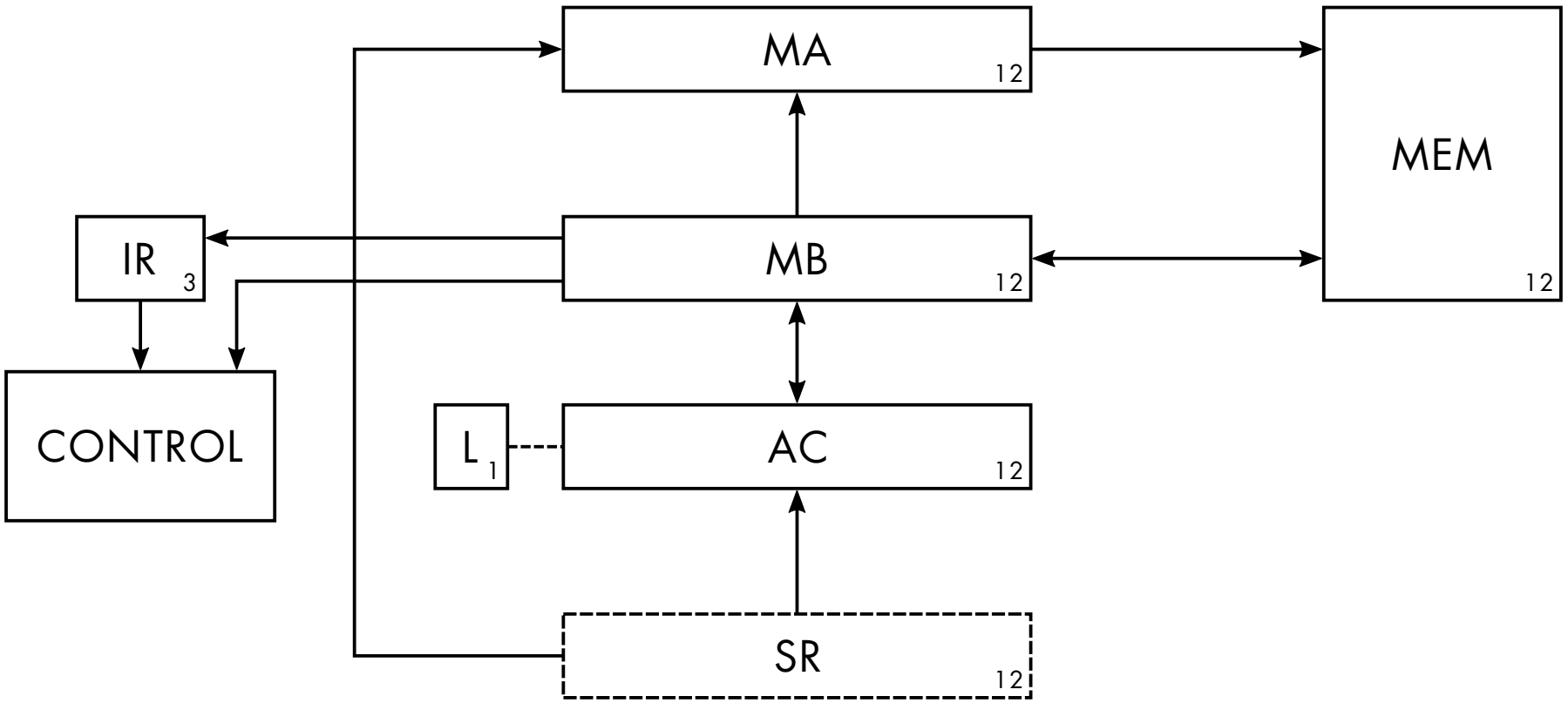


750000 CLA	AC ← 0		CLEAR AC
744000 CLL	L ← 0		CLEAR L
741000 SKP	reverse SKIP condition		SKIP
740400 SNL	L ≠ 0: SKIP	OR	SKIP ON NON-ZERO L
740200 SZA	AC = 0: SKIP		SKIP ON ZERO AC
740100 SMA	AC < 0: SKIP		SKIP ON MINUS AC
741400 SZL	L = 0: SKIP	AND	SKIP ON ZERO L
741200 SNA	AC ≠ 0: SKIP		SKIP ON NON-ZERO AC
741100 SPA	AC ≥ 0: SKIP		SKIP ON PLUS AC
740040 HLT	HALT		HALT
740020 RAR	L:AC ← rotate L:AC right		ROTATE AC RIGHT
740010 RAL	L:AC ← rotate L:AC left		ROTATE AC LEFT
742020 RTR	RAR twice		ROTATE TWICE RIGHT
742010 RTL	RAL twice		ROTATE TWICE LEFT
740004 OAS	AC ← AC ∨ SR		OR AC SWITCH REG.
740002 CML	L ← ~L		COMPLEMENT L
740001 CMA	AC ← ~AC		COMPLEMENT AC

IO TRANSFER

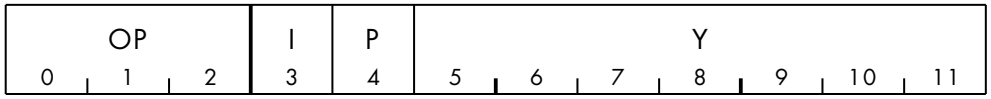


PDP-5 block diagram



PDP-5 summary

MAIN INSTRUCTIONS



P=1: Y ← Y + CURRENT PAGE
I=1: Y ← C(Y)

0 AND	AC ← AC ∧ C(Y)	AND
1 TAD	L:AC ← AC + C(Y)	TWO'S COMP. ADD
2 ISZ	C(Y) ← C(Y)+1. C(Y) = 0: SKIP	INDEX, SKIP IF ZERO
3 DCA	C(Y) ← AC. AC ← 0	DEPOSIT, CLEAR AC
4 JMS	C(Y) ← PC. PC ← Y+1	JUMP TO SUBROUTINE
5 JMP	PC ← Y	JUMP
6 IOT		IO TRANSFER
7 OPR		OPERATE

IO TRANSFER

INTERRUPT SYSTEM

6001 ION	INTERRUPTS ON
6002 IOF	INTERRUPTS OFF

KEYBOARD

6031 KSF	FLAG = 1: SKIP
6032 KCC	FLAG ← 0. AC ← 0
6034 KRS	AC ← AC ∨ BUF
6036 KRB	KCC. KRS
RECEIVED	BUF ← char. FLAG ← 1

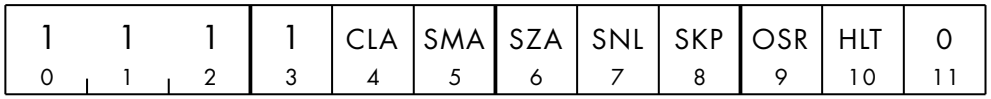
TELEPRINTER

6041 TSF	FLAG = 1: SKIP
6042 TCF	FLAG ← 0
6044 TPC	BUF ← AC. *XMIT*
6046 TLS	TCF. TPC
XMITTED	FLAG ← 1

OPERATE GROUP 1

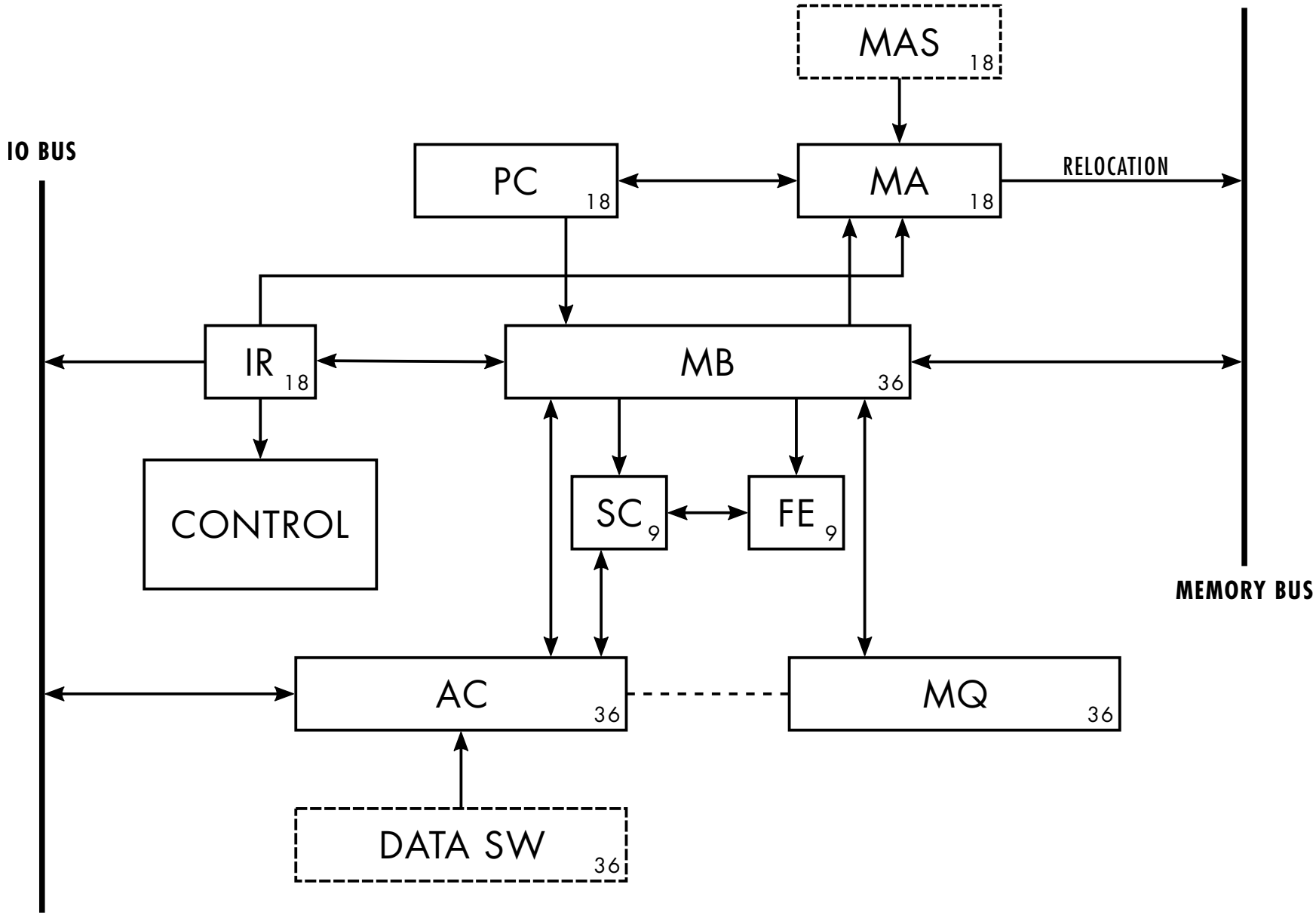
7200 CLA	AC ← 0	CLEAR AC
7100 CLL	L ← 0	CLEAR L
7040 CMA	AC ← ~AC	COMPLEMENT AC
7020 CML	L ← ~L	COMPLEMENT L
7010 RAR	L:AC ← rotate L:AC right	ROTATE AC RIGHT
7004 RAL	L:AC ← rotate L:AC left	ROTATE AC LEFT
7012 RTR	RAR twice	ROTATE TWICE RIGHT
7006 RTL	RAL twice	ROTATE TWICE LEFT
7001 IAC	L:AC ← AC + 1	INDEX AC
7041 CIA	L:AC ← ~AC + 1	COMPLEMENT, INDEX AC

OPERATE GROUP 2



7600 CLA	AC ← 0		CLEAR AC
7500 SMA	AC < 0: SKIP	OR	SKIP ON MINUS AC
7440 SZA	AC = 0: SKIP		SKIP ON ZERO AC
7420 SNL	L ≠ 0: SKIP		SKIP ON NON-ZERO L
7510 SPA	AC ≥ 0: SKIP	AND	SKIP ON POSITIVE AC
7450 SNA	AC ≠ 0: SKIP		SKIP ON NON-ZERO AC
7430 SZL	L = 0: SKIP		SKIP ON ZERO L
7410 SKP	reverse SKIP condition		SKIP
7404 OSR	AC ← AC ∨ SR		OR SWITCH REGISTER
7402 HLT	HALT		HALT

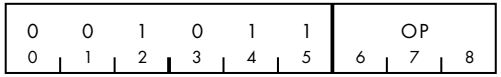
PDP-6 block diagram



PDP-6 summary

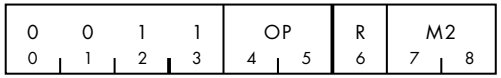
UUO

FSC AND BYTE



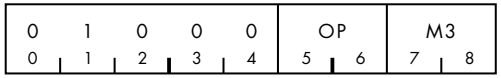
OP: 2/FSC, 3/IBP, 4/ILDB, 5/LDB, 6/IDPB, 7/DPB

FLOAT



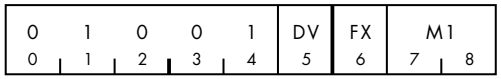
OP: 00/FAD, 01/FSB, 10/FMP, 11/FDV
R: 0/—, 1/ROUND

FULL WORD TRANSFER



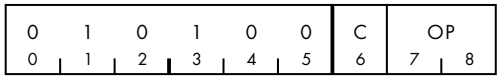
OP: 00/MOVE, 01/MOVS, 10/MOVN, 11/MOVM

MULTIPLY/DIVIDE



DV: 0/MUL, 1/DIV
FX: 0/INTEGER, 1/FIXED POINT

SHIFT/ROTATE

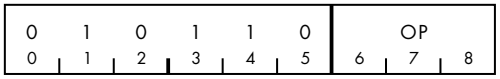


C: 0/SINGLE AC, 1/COMBINED
OP: 00/ASH, 01/ROT, 10/LSH

MISC

OP: 0/EXCH, 1/BLT, 2/AOBJP, 3/AOBJN
4/JRST, 5/JFCL, 6/XCT

STACK AND SUBROUTINES



OP: 0/PUSHJ, 1/PUSH, 2/POP, 3/POPJ
4/JSR, 5/JSP, 6/JSA, 7/JRA

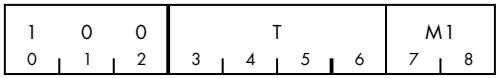
ADD/SUBTRACT

SB: 0/ADD, 1/SUBTRACT

ARITHMETIC TESTING AND MODIFICATION

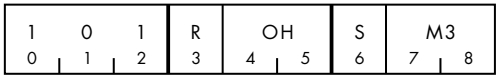
OP: 0/ACCP, 1/MEMAC, 2/MEMAC+1, 3/MEMAC-1
M: 0/NO MEM; CAI-JUMP, 1/MEM; CAM-SKIP
N: NEGATE CONDITION
E: SKIP/JUMP IF EQUAL TO ZERO
L: SKIP/JUMP IF LESS THAN ZERO

BOOLEAN



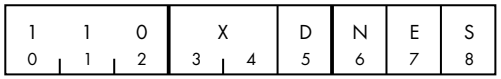
T: TRUTH TABLE

HALF WORD TRANSFER



R: DST: 0/LEFT, 1/RIGHT
OH: OTHER HALF: 00/—, 01/ZERO, 10/ONE, 11/S.EXT
S: SWAP SRC

LOGICAL TESTING AND MODIFICATION



X: MASKED BITS: 00/—, 01/ZERO, 10/COMP, 11/ONE
D: 0/IMMEDIATE, 1/DIRECT
N: NEGATE CONDITION
E: SKIP IF EQUAL TO ZERO
S: SWAP SRC

IO TRANSFER

OP: 000/BLKI, 001/DATAI, 010/BLKO, 011/DATAO
100/CONO, 101/CONI, 110/CONSZ, 111/CONSO

M1: 00/TO AC
01/IMM, TO AC
10/TO MEM
11/TO BOTH

M2: 00/TO AC
01/TO AC,AC+1
10/TO MEM
11/TO BOTH

M3: 00/MEM TO AC
01/IMM TO AC
10/AC TO MEM
11/MEM TO MEM