PDP-11

ENGINEERING SPECIFICATIONS

NUMBER: ES:0001 - Errata
DATE: 3 April 1969

BASIC PDP-11/30 DESCRIPTION
and
INSTRUCTION SET ERRATA

Originated by: H. McFarland

Approved by:

COMPANY CONFIDENTIAL

H. L. McFarland
Architecture

Hardware

Systems Software

Rick Merrill
Applications Software

Roger C. Cady
Project Manager

ABSTRACT

Errata for description and instruction set for basic PDP-11/30 system dated 28 March 1968 (ES:0001).

Errata for Engineering Specification ES0001

Cover Sheet: Date should be 28 March 1969.

Page 1: Add "(1 word)." after the reference to 2 bytes at the end of opening paragraph.

Page 3: Definition of .

Replace "location" with "instruction".

Page 6: Address Boundaries

Replace the second sentence of the first paragraph with: "Bytes (except a byte immediate reference or on the LP Stack) may be on even or odd boundaries. Auto-increment and auto-decrement of PC and LP is forced to ±2.

Page 7: MOV

Condition codes N and Z operate as in ADD.
Condition codes V and C are not changed.

Page 9: BRANCH

Delete "from location of Branch instruction", and add "from updated PC (as are all relative address calculations)." Delete "If condition not met: (PC)+2⟶(PC)" and add "If condition not met: Next instruction is executed".

Page 11: ROTATE/SHIFT

B	000110	0	OP	DESTINATION
15	14 9	8	7 6	5 0

		Operation	Mnemonics
OP:	Bit 7 = 0	Rotate	ROT
	Bit 7 = 1	Shift	SHF
	Bit 6 = 0	Right	+
	Bit 6 = 1	Left	-

Bit 8 is always Ø indicating a shift or rotate of one bit.

Page 12: EXECUTE

Replace description with:
"The instruction at ADDRESS of the low 256 memory words is executed." During the execution the PC becomes a page Ø address, and is appropriately modified by address calculation. Unless the executed instruction modifies PC the next instruction is after the EXC instruction. This location is also used in the JSR, WAIT, HALT and Interrupt instructions. An error trap will indicate the EXC instruction. It is illegal to execute an EXC instruction.

Page 12: TRAP

Delete mention of this instruction on basic machine.

Page 13: PUSH/POP

Change group name from PUSH/POP to OPERATE. Delete "basic Push/Pop instructions" and substitute "Return from Interrupt".

Page 14: PUSH/POP (cont.)

Delete PUS, POS and PUSP instructions. Recode RTI for X = 00010. The required access to the ST register previously provided by PUS and POS is provided by memory reference instructions to a last page address. All visible internal register will have such addresses. On Wait and Halt instructions, the PC will point to the next instruction.

Page 14: Traps on Basic Machine

Replace the last sentence with: "The trap results in ST and the instruction address causing the trap being pushed onto the LP Stack. A new PC and ST are obtained from the trap locations. Illegal addresses on LP or PC during TRAP sequences cause a HALT.

Appendix A: Rotate/Shift

Correct coding as follows;

ROT+1, A	0060
ROTB+1, A	1060
ROT-1, A	0061
ROTB-1, A	1061
SHF+1, A	0062
SHFB+1, A	1062
SHF-1, A	0063
SHFB-1, A	1063

Appendix A: Subroutine Return

The second RTS R,coded 00021,should be RTS @ R

Appendix A: Operate Group, Condition Codes

SEC	000261
SEV	000262
SEZ	000264
CLZ	000244

Appendix A: Operate Group, Push/Pop

Delete Push/Pop in above name. Delete PUS, POS, PUSP instructions, recode RTI to 000002.

Appendix B: Change the symbol for code 010000 to (RØ)+. Change the symbol for code 011000 to @(RØ)+.

Attached Instruction Sheet: Correct per above.

Page 3: Definition of "AB"

Replace 041102 with 041101

Page 8: AND coding

Change code within bits 14 to 12 to 101.

Page 8: GOTO

Change last line to "Condition codes operate as in MOV.

Page 13: 5th line from bottom - change "Reminder"to"Remainder"

Page 4: Note after syntax definitions.

NOTE: Care is necessary when using m = 7, as this register is incremented during instruction fetch, index, addressing, and immediate data and address references. In general, the PC is incremented to the next word immediately after use. For instance, when using addresses of the form A(7), the PC points to the word containing A when fetching A, but is incremented by 2 before being added to A to compute the effective address.