PDP-11

DATE: June 13

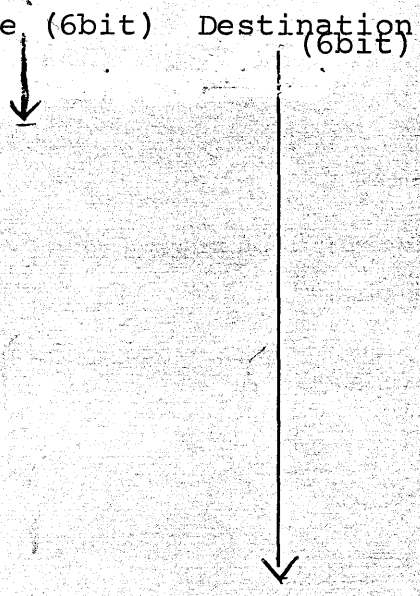
SUBJECT: PDP-11 EAE

TO: D. Saviers Lists A, C and Design Rev. Comm. FROM: Roger Cady

Proposed first pass at EAE instructions for PDP-11 are attached. This indicates the way in which these would be added within the frame-work of the basic instruction set.

Summary of Mnemonics:

MUL	A,B
DIV	A,B
*ROT±N	R,A
*ROTB±N	R,A
*LSH±N	R,A
*LSHD±N	R,A
*ASH±N	R,A
*ASHD±N	R,A
NOR	R,A
NORD	R,A



*R field optional. If blank, no index.

JUN 18 1969

PDP-11

DATE: June 13

SUBJECT: PDP-11 EAE

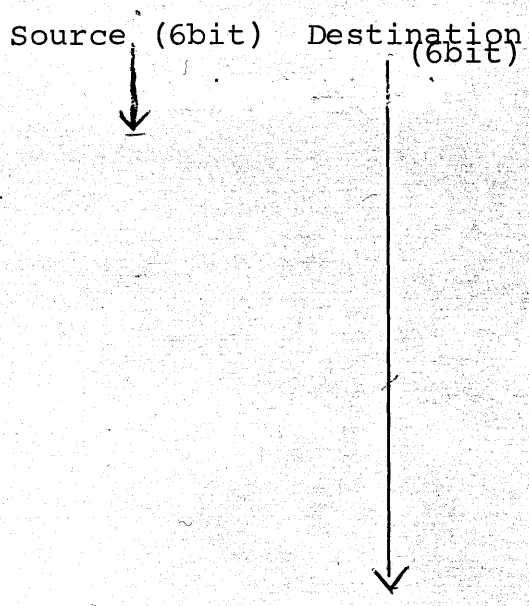
TO: Lists A, C and Design Rev. Comm.

FROM: Roger Cady

Proposed first pass at EAE instructions for PDP-11 are attached. This indicates the way in which these would be added within the framework of the basic instruction set.

Summary of Mnemonics:

MUL	A,B
DIV	A,B
*ROT±N	R,A
*ROTB±N	R,A
*LSH±N	R,A
*LSHD±N	R,A
*ASH±N	R,A
*ASHD±N	R,A
NOR	R,A
NORD	R,A



*R field optional. If blank, no index.

JUN 18 1969

PROPOSED EAE INSTRUCTIONS FOR PDP-11

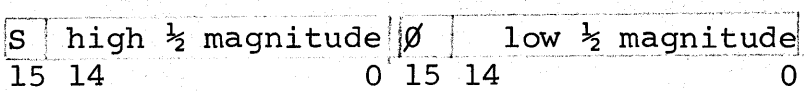
EAE will add the arithmetic power of multiply, divide, and a powerful multiple rotate, shift, and normalize group.

The adopted data structure is as follows:

Single Precision

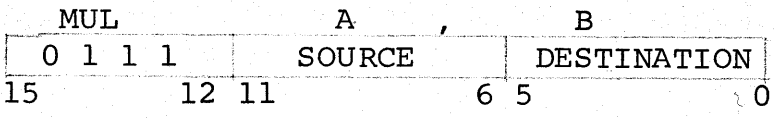
(Two's Complement)

Double Precision



All double word operands are assumed to be in this form, all double word results are stored in this form.

MULTIPLY

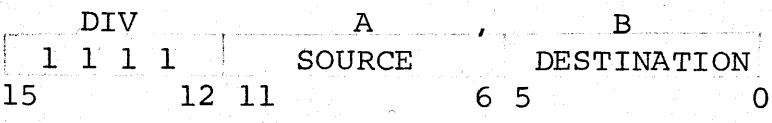


(SE)*(DE) ⟶ (DE), (DE+2)
(PC)+n ⟶ PC
operands are considered to be 16 point signed quantities

If both operands are maximum negative quantities, 1⟶V and operation will not take place. Otherwise N, Z set on result condition.

If destination is a register, the product will be stored in that register and the next register. (CAUTION: MUL A, %R5 will alter LP)

DIVIDE



$$\frac{(DE),\ (DE+2)}{(SE)} \longrightarrow (DE),\ \text{Remainder} \longrightarrow (DE+2)$$

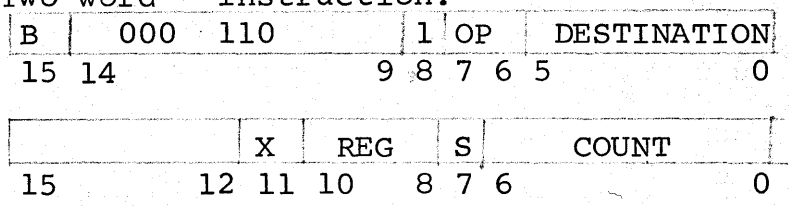
(PC)+n ⟶ (PC)

If both divisor and dividend are maximum negative quantities or if divisor is smaller than upper half of dividend 1⟶V otherwise 1⟶N if quotient negative 1⟶Z if quotient zero.

Register destination note same as in MUL

ROTATE/SHIFT GROUP

Two word instruction.



BITS B	OP		OPERATION	MNEMONIC
15	7	6		
0	0	0	Rotate with carry (word)	ROT
1	0	0	Rotate with carry (byte) In rotates, V gets complemented for every 1 shifted thru carry.	ROTB
0	0	1	Logical Shift (word)	LSH
1	0	1	Logical Shift (double word)	LSHD
0	1	0	Arithmetic Shift (word)	ASH
1	1	0	Arithmetic Shift (double word)	ASHD
0	1	1	Normalize (word)	NOR
1	1	1	Normalize (double word)	NORD

The second word determines direction and count for multiple Rotate/Shifts. The sign of the Count is the direction (+ = right, - = left). This is a Two's Complement 8 bit quantity.

ROT, ROTB, LSH, LSHD, ASH, ASHD:

If the X bit in the second word is set, the count is indexed by the contents of the register specified in the REG field.

NOR, NORD

The count and X bits are ignored if the instruction is normalize. The REG field determines where the normalize count is left after the operation. Previous contents of REG are destroyed.

Comments:

1. ROTB used to generate parity of byte data

2. Double word operations use the destination location and destination location +2.

3. Max effective count is 32. The effective count if indexed, after indexing) is determined by the least significant 5 bits.