

EXTERNAL CONTROL KEYS

ENABLE Program Interrupt
I Execute instruction in first location
C Execute one command cycle at a time
BREAKPOINT . . . Program control switch-tested by program

SPECIFICATIONS

OPERATION TIMES (22 bit word)

Add/subtract	12 μ sec.
Multiply	276 μ sec. (max.)
Divide	252 μ sec. (max.)
Square root	252 μ sec. (max.)

MEMORY

Type: Magnetostrictive delay lines.

Capacity: 1,808 words standard,
expandable to 15,888 words.

INPUT-OUTPUT

Standard: Automatic typewriter, paper tape punch and reader, 30 control inputs, 32 control outputs, high-speed block input-output (85 KC word rate).

Optional: High-speed paper tape punch and reader, magnetic tape units (six maximum) employing IBM 700 series format, punched card equipment.

DIMENSIONS

30 in high, 19 in. wide, 24 in. deep.

POWER REQUIREMENTS

115 volts, 60 cycles, at 100 watts.



Packard Bell Computer

A SUBSIDIARY OF PACKARD BELL ELECTRONICS
1905 ARMACOST AVENUE
LOS ANGELES 25, CALIFORNIA, GRANITE 8-4247

pb 250

COMMAND LIST

Packard Bell Computer

COMMAND LIST

OPERATIONS GROUPING

COMMAND STRUCTURE			
OP Code 6 bits	Address 13 bits	Sequence tag 1 bit	Index tag 1 bit

A and B Registers: One word registers, programmed independently or combined for multiplication, division, square root, and double precision operations.

C Register: For multiplication, division, tally, and control.

Operation	Mnemonic Code	Numeric Code	Description
Arithmetic	ADD	14	Add
	SUB	15	Subtract
	DPA	16	Double Precision Add
	DPS	17	Double Precision Subtract
	SQR	30	Square Root
	VLS	30	Variable Length Square Root
	DIV	31	Divide
	VLD	31	Variable Length Divide
	MUP	32	Multiply
	VLM	32	Variable Length Multiply
	CLA	45	Clear A
	CLB	43	Clear B
	CLC	44	Clear C
	GTB	41	Gray to Binary
CAM	56	Compare A and M	
Transfer	TAN	35	Transfer if A Negative
	TBN	36	Transfer if B Negative
	TCN	34	Transfer if C Negative
	TRU	37	Transfer Unconditionally
	TOF	75	Transfer on Overflow
	TES	77	Transfer on External Signal
Loading & Storing	LDA	05	Load A
	LDB	06	Load B
	LDC	04	Load C
	LDP	07	Load Double Precision
	IAC	01	Interchange A & C
	IBC	02	Interchange B & C
	STA	11	Store A
	STB	12	Store B
	STC	10	Store C
	STD	13	Store Double Precision
MCL	71	Move Command Line Block	
MLX	26	Move Line X to Line 7	
Logical & Shifting	EBP	40	Extend Bit Pattern
	AMC	42	AND M & C
	AOC	46	AND OR Combined
	EXF	47	Extract Field
	NAD	20	Normalize and Decrement
	LSD	21	Left Shift and Decrement
Control	RSI	22	Right Shift and Increment
	SAI	23	Scale Right and Increment
	NOP	24	No Operation
	HLT	00	Halt
Input-Output	DIU	50	Disconnect Input Unit
	RTK	51	Read Typewriter Keyboard
	RPT	52	Read Paper Tape
	RFU	53	Read Fast Unit
	LAI	55	Load A From Input Buffer
	CIB	57	Clear Input Buffer
	WOC	6X	Write Output Character
	PTU	70	Pulse to Specified Unit
BSO	72	Block Serial Output	
BSI	73	Block Serial Input	

NUMERIC ORDER

NUMERIC ORDER

Numeric Code	Mnemonic Code	Description
00	HLT	Halt
01	IAC	Interchange A & C
02	IBC	Interchange B & C
04	LDC	Load C
05	LDA	Load A
06	LDB	Load B
07	LDP	Load Double Precision
10	STC	Store C
11	STA	Store A
12	STB	Store B
13	STD	Store Double Precision
14	ADD	Add
15	SUB	Subtract
16	DPA	Double Precision Add
17	DPS	Double Precision Subtract
20	NAD	Normalize and Decrement
21	LSD	Left Shift and Decrement
22	RSI	Right Shift and Increment
23	SAI	Scale Right and Increment
24	NOP	No Operation
26	MLX	Move Line X to Line 7
30	SQR	Square Root
31	DIV	Divide
32	MUP	Multiply
34	TCN	Transfer if C Negative
35	TAN	Transfer if A Negative
36	TBN	Transfer if B Negative
37	TRU	Transfer Unconditionally
40	EBP	Extend Bit Pattern
41	GTB	Gray to Binary
42	AMC	And M & C
43	CLB	Clear B
44	CLC	Clear C
45	CLA	Clear A
46	AOC	AND OR Combined
47	EXF	Extract Field
50	DIU	Disconnect Input Unit
51	RTK	Read Typewriter Keyboard
52	RPT	Read Paper Tape
53	RFU	Read Fast Unit
55	LAI	Load A from Input Buffer
56	CAM	Compare A and M
57	CIB	Clear Input Buffer
6X	WOC	Write Output Character
70	PTU	Pulse to Specified Unit
71	MCL	Move Command Line Block
72	BSO	Block Serial Output
73	BSI	Block Serial Input
75	TOF	Transfer on Overflow
77	TES	Transfer on External Signal