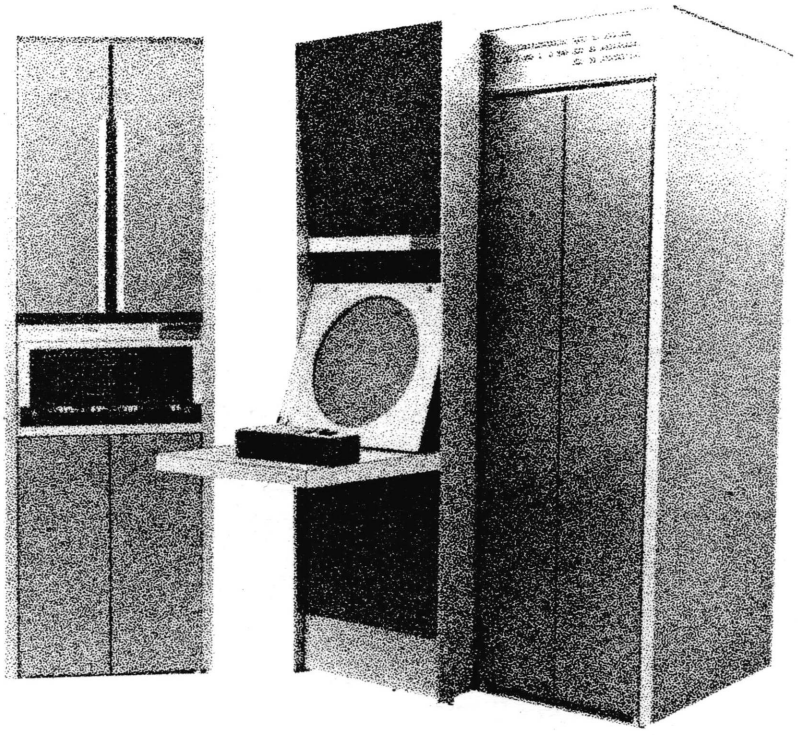
PROGRAMMED BUFFERED DISPLAY TYPE 338



DIGITAL EQUIPMENT CORPORATION • MAYNARD, MASSACHUSETTS

The 338 Programmed Buffered Display is a precision incremental CRT display driven by a small, high-speed, general purpose computer. The computer (PDP-8) is a single address, 12-bit fixed word-length machine using 2's complement binary arithmetic. A random access magnetic core memory with a complete cycle time of 1.5 μs is used to achieve a maximum computation rate of 314,000 additions per second.

The Type 338 Buffered Display permits rapid conversion of digital computer data into graphic and tabular form. Its combined capabilities offer the user an unusual degree of versatility and accuracy.

A self-contained unit with built-in control and power supplies, the Type 338 Buffered Display requires only logic level inputs for operation and may be easily connected to any digital system as a buffered display with processor, or it may stand alone as a powerful computer-driven display system.

The display shares the memory of the PDP-8 by means of the data break channel, which has direct access to memory, and operates on a display file which has been stored in memory by the computer. The display file contains control commands and data in a digital form which the Type 338 Display converts into vectors, points, or characters forming a full graphic picture which can be modified by external command.

Multiple modes of operation, fast point and vector plotting, multiple scale and intensity levels and automatic scissoring are all standard on the 338 Display. Standard I/O devices include an ASR-33 Teletype, high-speed light pen and a 12-pushbutton function box. A character generator, automatic search logic, zoom logic, a slave mode which can accommodate up to seven slaves, and various peripheral equipment, including interfaces to other computers, can be provided as options.

Interfacing the display portion of the 338 with the PDP-8, which acts as the controller, via the data break channel enables the display to obtain 12-bit data and command words from the memory of the PDP-8 at a rate of 1.5 μs per word without interrupting the program that is being executed. This is accomplished by stealing cycles from the PDP-8 which can be executing a program while the 338 is displaying information previously stored in memory. When an interrupt occurs, e.g. a light pen interrupt, the PDP-8 stops execution, services the interrupt, and resumes execution from where it was interrupted. The 338 is allowed to "steal" a maximum of one out of every four computer cycles (one every 6 μs), so that it will not completely tie up the PDP-8.

FEATURES

PDP-8 - A 12-bit per word computer with a non volatile, random-access core memory - 4096 words (1.5 μs read-readwrite cycle) expandable to 32K. Eight basic computer instructions plus microprogrammed I/O instructions which permit easy interfacing to external sources of data such as, other computers, dataphone, paper tape, disk and keyboard.

Software Package - Complete software package including FORTRAN, Symbolic Assembler, DDT, Floating Point Arithmetic, and Display maintenance and demonstration programs.

Data Aquisition by Cycle Stealing - Data and control instructions are received from the PDP-8 memory through the data break channel. The display has direct high-speed access (1.5 μs per word) to memory which allows information to be displayed at the same time the PDP-8 is executing another computer program.

Precision - The magnetic deflection and focusing system of the 338 Display insures precise data conversion and uniform spot repeatability. Spot location stability is ±0.05 in. over an eight hour period with an ambient temperature range of ±3°F. Spot size is approximately 0.015 in.

Fast Point Plotting - The 338 Display has a 9-3/8 in. square display area comprised of 1024 x 1024 points. Discrete points can be plotted on this area at a rate of one every 46 μs in point mode or every 40 μs in graph-plot mode. If the points are less than a certain distance from each other, the rate increases to about 20 μs per point (14 μs per point in graph-plot mode).

The plotting rate in other modes is 1.1 μs for intensified points and 300 ns for non-intensified points.

Multiple Modes and Instructions - Points, straight lines, curves and characters can be plotted by using any one or more of the seven data-state modes which include point mode, increment mode, vector mode, vector-continue mode, short vector mode, character mode and graph-plot mode. The 338 Display has 11 control-state instructions available.

Multiple Scale and Intensity Levels - The 338 Display has four scales which can be used in every data mode except point mode. Two flip-flops control the scale factors of X1, X2, X4 and X8. Lines are constructed as a series of points, rather than an analog fashion, eliminating intensity modulation and making the line intensity constant, regardless of the length of the line.

Automatic Scissoring - The 338 Display permits the programmer to draw pictures up to 75 in. x 75 in. Only a 9-3/8 in. x 9-3/8 in. square can be displayed at any one time. This "window" can be easily moved anywhere on the larger picture so that all positions may be viewed.

Easy Man/Machine Interaction - The user can easily communicate with the display system through the various input devices of the 338 Display, e.g., the light pen, the function box and the keyboard. The operator can use the light pen to tell the display that a change should be made to the portion of the picture to which the light pen is pointing. Subroutines enable the light pen to draw lines and figures, position symbols and erase elements of the picture.

The 12 pushbutton function box provides a simple means of communication with the system. Several display control state commands provide display file skips when the pushbuttons are in certain configurations. Reset buttons, for each bank of six pushbuttons, and a manual interrupt button, which causes the PDP-8 to interrupt, are also located on the function box.

Pressing a button complements an associated flip-flop. For reference, the pushbutton is lit when the flip-flop is in the 1 state. The PDP-8 can load the pushbutton flip-flops from the AC via an IOT command, and can read the contents of the flip-flop into the AC.

Rand tablets, trackballs, special keyboards and other I/O devices may be interfaced to the PDP-8 for added man/machine capabilities.

Communication with Display Registers - The contents of the x-y position registers, the display address counter, and the status of various display registers can be read into the accumulator of the PDP-8 via IOT instructions. Likewise, the PDP-8 can load certain registers in the display used for starting the display or setting up initial conditions.

Large Computer Interfacing - The 338 Display can be used as a terminal for a larger computer which can be interfaced to the PDP-8 via the data break channel or the program interrupt. Interfaces to the DEC PDP-10, CDC 3000 and 6000 series, IBM 7090, 94 and the 360 series are among those available.

PROGRAMMING THE 338 DISPLAY

The 338 Display receives the commands and data, which are needed to generate the picture, from a display file located in the memory of the PDP-8. The display file is simply a group of consecutive or linked locations in memory and contains 12-bit command and data statements, which have been inserted in memory by some external device (another computer, tape, or disk) or generated by the PDP-8.

The PDP-8 initializes the registers in the display (scale, intensity, break field, light pen, interrupt flags), loads the starting address of the display file into the display address counter and starts the display. The PDP-8 may now execute any other program; however, it must remain in operation, even if it is only a jump to itself, otherwise the display will not obtain a data break cycle to receive the next instruction from the PDP-8's memory. After the display is started, it will call on the PDP-8 only to service interrupts from the light pen, function box, edge flag, or stop flag. The PDP-8 can be programmed to modify the display

file upon interruption of any of these devices.

Due to the nature of the C.R.T., the information displayed must be refreshed about 30 times per second, implying that the display must loop through the display file at least that fast. Elements of the display file are linked by being in consecutive locations, by the use of jumps which are executed by the display or by jumps to subroutines which are executed and control then returned to the next sequential command following the subroutine jump. This last command greatly simplifies graphics programming. The last element of the display file is usually an instruction which tells the display to "jump" back to the starting address of the display file. "Skip" instructions cause the display processor to sense certain conditions and enable branching to different parts of the program if these conditions are met.

Display States - In fetching 12-bit instructions from the PDP-8's memory, the display processor interprets them as either control state or data state instructions. The display initially starts in control state and changes to data state only after an "enter data state" command is given. It remains in data state until the escape bit in a data state instruction is set. When the display processor finishes an instruction, it returns to fetch the next sequential instruction in memory except in the case of a "jump", where it goes to the location specified by the jump instruction, or a "Pop", which returns control from a subroutine to the next instruction in the main program.

Control State Instructions - Control state instructions set various registers which control the operation of the display, such as, address, scale, intensity and light pen status. These instructions are 12 bits (1 word) long except the "jump" and "push jump" which requires 24 bits (2 words), a 3 bit OP code, 5 bits to set parameters, 1 bit for a push jump (jump to subroutine), and a 15-bit address, so that 32K of memory may be referenced.

SET PARAMETER - The scale and intensity flip-flops can be changed and the light pen turned on or off.

SET MODE - The display can be stopped, display flags cleared, data state mode register set, sector bits (correspond to size of "paper") cleared, coordinate bits (position) cleared, and command to enter data state given.

JUMP 2 WORDS - The 1st word sets scale and light pen and loads the break field (three high-order bits of address to which the jump will be made). The word immediately following is interpreted as the lower 12 bits of the jump address. By setting bit 8 of the jump command to a 1, the instruction becomes a "push-jump" or a jump to subroutine. (See display subroutines.)

POP - Return from subroutine. (See display subroutines.)

CONDITIONAL SKIP - Allows pushbuttons to be selected and sensed for a certain condition. If the condition is true, the display skips the next two words in the data file. Display skips skip over two words so that jump instructions, which are two words long, can be skipped.

ARITHMETIC SKIP - The states of the pushbuttons are compared to the last six bits of the Arithmetic skip command. If all buttons are the same, the test succeeds and follows normal sequence. If it fails, the display skips two words.

SKIP ON FLAGS - The display skips two words if it senses that a selected flag is on.

COUNT - Enables the increasing or decreasing of scale and/or intensity in the display file and can also turn the blink on and off. The blink logic alternately turns the beam intensification on and off for 1/4 second.

60 CYCLE SYNC - Enables the programmer to synchronize the refreshing of the display with input line voltage. If it takes the display less than 1/60s to cycle through the display file, the scope will be refreshed 60 times per second; if greater than 1/60s and less than 1/30, it will be refreshed 30 times per second, etc.

Data State Instructions - Words fetched from memory, while the display is in data state, affect the deflection and intensification of the beam in some way, depending on the mode that the display is in.

INCREMENT MODE - This mode is used to draw alphanumeric characters, small symbols, and curves. A half-word instruction will cause the beam position to

be stepped one, two, or three times in one of eight directions. Direction 0 is to the right, direction 1 is up and to the right, etc.

VECTOR MODE - This mode is used to draw straight line segments. A two-word instruction causes the beam position to be stepped along a line represented by a 10-bit delta y and a 10-bit delta x, both of which have sign bits.

VECTOR CONTINUE MODE - This mode is used to draw a straight line to the edge of the screen. It is similar to Vector mode but causes the vector to be extended until an "edge" is encountered.

SHORT VECTOR MODE - This mode is used to draw figures composed of short line segments. A one-word instruction has a 4-bit delta y and a 4-bit delta x, plus sign bits. It is transformed within the display to the same format as Vector mode and operates in the manner.

The preceding modes are "incrementing," that is, they move the beam by counting the x- and y-position registers. The counting is done at 1.1 ± 0.1 μs per step on an intensified move; at 0.30 μs per step on a nonintensified move.

POINT MODE - This mode is used for random point plotting. A two-word instruction causes new y- and/or x-coordinates to be set into the y- and x-position registers.

GRAPH-PLOT MODE - This mode is used to draw curves of mathematical functions. A one-word instruction causes the y or x position register to be changed. At the same time, the other register is incremented by a count of one, two, four, or eight, depending on the current scale factor. This mode is useful for plotting curves resulting from a series of solutions of an equation.

Point and Graph-Plot modes operate at one of two rates, depending upon the position of the new point with respect to the previous point. The high-order three bits of the 10 y and 10 x position bits are compared to the corresponding bits in the new data words. If they are the same, the delay for beam-settling time is 7 ± 1 μs; if they differ, a 35-μs delay is used.

CHARACTER MODE - A half-word instruction, using the Type VC38 Character Generator. (See Character Generation.)

AUTOMATIC SCISSORING

The y and x positive registers of the Type 338 each contain 13 bits, representing a 75 in. x 75 in. drawing. The low-order 10 bits of each register are used to determine the beam position on a 9.375 in. x 9.375 in. square matrix which defines the viewing area. The high-order three bits of each register are called "sector bits." The beam intensification is allowed only when all the sector bits contain zero.

The purpose of using more bits than are necessary for covering the viewing matrix is to allow large drawings to be constructed. Although only a small segment of a large picture can be viewed at any one time, a simple method exists for "shifting" the "drawing" back and forth to allow viewing of any desired section.

The method is as follows.

a. Construct the picture by using modes other than point and graph-plot. (Although this restriction may be ignored in special cases, it is generally necessary.)

b. Construct the picture as a "closed figure." That is, draw an invisible vector from the end point to the starting point of the drawing.

c. At the end of the display file, before jumping back to the start of the figure, use the pushbutton "skip" instructions to determine whether the picture is to be shifted. If it is determined that a left shift of the figure is desired, draw a short invisible vector to the left. Repeat for right, up, and down.

d. Jump back to the start of the display file.

It should be noted that the "shifting routine" consists entirely of display file instructions. Comparing this method of scissoring to a computer routine method will leave little doubt that the "automatic" version is quite desirable.

DISPLAY JUMP AND SUBROUTINE INSTRUCTIONS

The display jump instruction has 15 address bits so that a jump may be executed to any location in the display file, with a 32K memory.

The display subroutine instructions are "push-jump," (an extension of the jump instruction), and "pop," the return from subroutine. The push-jump works in the following manner. The current state of the display (L.P. enable, mode, scale, intensity level) is stored, along with the return address, in two successive locations in the first 4K of PDP-8 memory. The locations are determined by a hardware push-down pointer in the Type 338 logic. (This pointer is set initially by an IOT instruction.) The normal jump is then executed.

To return from a subroutine, the "pop" instruction is executed. It has no address bits. Its function is to return the display to a previous state by sending the last words on the push-down list back to the display.

The "push-down list" approach to subroutines, as implemented on the Type 338, has certain advantages over the normal "jump to subroutine" used within a computer.

a. Memory space is conserved, since return address locations are not required in each subroutine in memory.

b. A subroutine can be called any number of times before return to the main routine.

c. Since the state of the display is saved and subsequently restored, subroutines are truly "transparent;" that is, they leave the state of the display after the return the same as before the subroutine call.

d. The subroutines can be allowed to change the state of the display if this is desired, by using one or more of the "inhibit restore" bits available in the "pop" instruction. The programmer can elect to independently inhibit restoration of mode, L.P., and scale, or intensity information.

CHARACTER GENERATION

Using the standard Type 338 modes: Alphanumeric or other often used characters may be constructed as subroutines. Using increment mode, short vector mode, or vector mode to draw each.

Using the optional Type VC38 Character Generator: This device is used if many alphanumerics are needed. It conserves memory (2 characters per PDP-8 memory location) and is slightly faster than the method described above. The Type VC38, when installed, is used by mode 5, Character Mode. A 6-bit or 7-bit byte is gated to the Type VC38, which causes the corresponding character to be drawn. Characters are generally drawn in a 5 x 7 matrix, although special characters may extend outside of the matrix.

338 BUFFERED DISPLAY INSTRUCTIONS

Group	Mnemonic Symbol	Octal Code	Operation
	LPOF	0040	Light Pen off.
	LPON	0060	Light Pen on.
	SC1	0400	Set Scale to x1.
	SC2	0500	Set Scale to x2.
	SC4	0600	Set Scale to x4.
	SC8	0700	Set Scale to x8.
	INT	0010	Set the Intensity for 0-7.
	EDS	1001	Enter Data State.
	CCB	1002	Clear Coordinate Bits.
	CSB	1004	Clear Sector Bits.
	point	1100	Set mode 0.
	incr	1110	Set mode 1.
	vec	1120	Set mode 2.
	vecon	1130	Set mode 3.
	svec	1140	Set mode 4.
	char	1150	Set mode 5.
	graph	1160	Set mode 6.
	CLDF	1200	Clear Display Flags.
	STOP	1400	Stop.
	JUMP	2000	Jump to address contained in last digit and the next word.
	PJMP	2010	Jump to subroutine addressed same as JUMP.
	POP	3000	Exit from Subroutine to next address after PJMP instruction.*
	PEDS	3001	Pop and enter data state.
	PNI	3002	Pop and inhibit restoring intensity.
	PNLS	3004	Pop and inhibit restoring Light Pen and scale.
	PNM	3010	Pop and inhibit restoring mode.
	SK1	4000	Skip if pushbuttons (0-5) indicated by binary representation of last two digits are on.
	SK2	5000	Same as SK1 but tests pushbuttons 6-11.
8a	INV	0400	Invert sense of test.
8a	CLAT	0200	Clear bits tested after test.
8a	COAT	0100	Complement bits tested after test.
	SK3	6000	Arithmetically compare pushbuttons (0-5) with last 2 digits of instruction; skip if not equal.
	SK4	6100	Same as SK3 but for buttons 6-11.
11	SKIP	6240	Unconditional skip (two locations).
11	SNSZ	6220	Skip if sector zero flag is not up.
11	SPB1	6210	Skip if pushbutton (0-5) hit flag is off.
11	SPB2	6204	Skip if pushbutton (6-11) hit flag is off.
11	CLPSI	6201	Clear light pen sense indicator.
11	SLPSI	6202	Skip if not light pen sense indicator.
12	SCUP	6340	Count scale up.
12	SCDN	6360	Count scale down.
12	INTUP	6310	Count intensity up.
12	INTDN	6314	Count intensity down.
12	BKON	6302	Turn blink on.
12	BKOF	6301	Turn blink off.
13	SG 0	6400	Set Slave Group 0.
14	SG 1	6500	Set Slave Group 1.
15	SG 2	6600	Set Slave Group 2.
16	SG 3	6700	Set Slave Group 3.
13a,14a,			
15a,16a,	SU0	0040	Turn Light Pen and Intensity off on unit 0.
	LP0	0060	Unit 0 Light Pen on.
	IT0	0050	Unit 0 Intensity on.
	SU1	0004	Turn Light Pen and Intensity off on unit 1.
	LP1	0006	Unit 1 Light Pen on.
	IT1	0005	Unit 1 Intensity on.
17	CACIF	7001	Clear AC Idle flag.
17	SACIF	7002	Skip on AC Idle flag.

*PJMP instruction is two words long.
All instructions within a group may be cascaded to form combined instructions.
Sub group instructions, denoted by "a," must be accompanied by their group instruction.
Instructions in Group 1 may be cascaded with instructions from groups 2, 4, 5, and 6.

PROGRAMMING EXAMPLE

This example of display programming is intended as a demonstration of some of the capabilities of the display. Rather than micro-programming the control state instructions, the assembler tables can be extended to include mnemonics for the various instructions.

This routine would cause a square, 1300 (octal) units on a side, to be displayed with the lower left corner at y = 100, x = 220.

Computer Instructions:

```
start,  CLA;            clear accumulator
        SIC;            IOT 145 - set initial conditions to zero
        TAD SA1;        break field register number to AC
        LBF;            IOT 155 - load break field register
        CLA
        TAD SA2;        starting address of display file to AC
        INIT            IOT 165 - initialize display
        jmp .;          (or go to another routine)
SA1,    4400;           select break field 1
SA2,    1356;           S.A. of display file
```

Display File:

```
11356,  SC1 INT7;       set scale to 1, set intensity to 7
        point eds;      set point mode, enter data state
        100;            set Y to 100
        4220;           set X to 220, enter control state
        vec eds;        set vector mode, enter data state
        4000;           delta y = 0, intensify vector
        1300;           delta x = 1300
        5300;           delta y = 1300, intensify vector
        0;              delta x = 0
        4000;           delta y = 0, intensify vector
        3300;           delta x = -1300
        7300;           delta y = -1300, intensify vector
        4000;           delta x = 0, enter control state
        jump 1;         jump instruction, go to
        1362;           loc 11362 to repeat vectors
```

IOT INSTRUCTIONS FOR 338

Buffered Display

Group	Mnemonic Symbol	Octal Code	Operation
1	RPDP	6051	Read push down pointer
	RXP	6052	Read X register
	RYP	6054	Read Y register
	RDAC	6061	Read contents of Display Address Counter
	RS1	6062	Read Status 1
	RS2	6064	Read Status 2
	RPB	6071	Read push buttons
	RSG1	6072	Read slave group 1
	RSG2	6074	Read slave group 2
2	SPDP	6135	Set push down pointer with contents of AC
	SPLP	6132	Skip on L.P. flag
	SPSP	6142	Skip on slave L.P. flag
	SIC	6145	Set initial conditions
	SPEF	6152	Skip on Edge flag
	LBF	6155	Load Break Field Register; Load six push buttons; stop display
	SPEC	6162	For Special Options
	INIT	6165	Initialize: Set the DAC with the contents of AC*
	SPSF	6171	Skip on stop flag
	SPMI	6172	Skip on manual interrupt flag
	RES	6174	Resume: After LP, Edge, External Stop

Group 1 transfers information from the display to the PDP-8.

Group 2 transfers information from the PDP-8 to the display.

*6164, with AC cleared, is used to resume after stop flag.

STATUS FORMAT

RS1 Bit

0: Light pen hit flag
1: Vertical edge flag
2: Horizontal edge flag
3: Stop flag
4: Sector 0 flag
5: Control flip-flop
6: Manual interrupt flag
7: PB hit flag
8: Display interrupt flag
9: Contents of the Break Field register
10: Contents of the Break Field register
11: Contents of the Break Field register

RS2 Bit

0: Half word flip-flop
1: Light Pen enable
2: High order Y register bit
3: High order X register bit
4: Scale
5: Scale
6: Mode
7: Mode
8: Mode
9: Intensity
10: Intensity
11: Intensity

RPB State of 12 push buttons.

RSG1 L.P. status, intensify status, and L.P. hit status for slaves 0, 1, 2, 3.

RSG2 L.P. status, intensify status, and L.P. hit status for slaves 4, 5, 6, 7.

SIC Set initial conditions format.

Bit

0: Enable edge flag interrupt
1: Enable L.P. flag interrupt
2,3: 00 Do not disable light pen on light pen hit.
01 Do not disable light pen on light pen hit.
10 L.P. re-enabled by next data request.
11 L.P. re-enabled only by a control state command 0.
4,5: Set Y dimension
00: 9.375"
01: 19.75"
10: 37.5"
11: 75.0"
6,7: Set X dimension; same as Y
8: Intensify all points
9: Inhibit edge flags
10: Enable interrupt on push button hit
11: Enable interrupt on stop flag

LBF Load Break Field and Set Push Buttons

Bit

0: Enable change of break field
1: New break field
2: New break field
3: New break field
4: Enable change of push button
5: 0 - push buttons 0-5
1 - push buttons 6-11
6-11: Set push buttons

If AC0, 4 = 0: IOT 154 stops display
IOT 151 tests "ext dsply stop" flag

DATA STATE MODE FORMAT

Mode	Function	Word	0	1	2	3	4	5	6	7	8	9	10	11
0	Point	1	Intensify	Inhibit Setting Y	10 Bit Y Coordinate									
		2	Esc (enter control state)	Inhibit Setting X	10 Bit X Coordinate									
1	Increment		Intensify	No. of Moves (00=Move Once and Esc.)		Direction of Moves (0-7)			Same as Bits 0-5					
2	Vector	1	Intensify	±	10 Bit Delta Y									
		2	Escape	±	10 Bit Delta X									
3	Vector Continue	1	Intensify	±	10 Bit Delta Y									
		2	Escape	±	10 Bit Delta X									
4	Short Vector		Intensify	±	4 Bit Delta Y				Esc.	±	4 Bit Delta X			
5	Character (Optional)		C_1						C_2					
6	Graph Plot		Escape	0 - Set Y, incrx 1 - Set X, incry	10 Bit X or Y Coordinate									
7	Spare													

CONTROL STATE

Instruction	OP Code (Bits 0, 1, 2)	3	4	5	6	7	8	9	10	11	
Parameters	0	Set Scale	S_0	S_1	Set L.P.	L.P.	Set Intensity	1_0	1_1	1_2	
Mode	1	Stop	Clear Display Flags	Set Mode	M_0	M_1	M_2	Clear Sector Bits	Clear Coord. Bits	Enter Data State	
Jump	2	Same As Parameters					Push	Break Field 0	Break Field 1	Break Field 2	**Low order 12-bits of JMP Address
Pop	3						Inhibit Restoring Mode	Inhibit Restoring L.P., Scale	Inhibit Restoring Intensity	Enter Data State	
Conditional Skip	4	Sense of Test*	Clear Selected Bits After Test	Complement Selected Bits After Test	T_0	T_1	T_2	T_3	T_4	T_5	
Conditional Skip	5				T_6	T_7	T_8	T_9	T_{10}	T_{11}	
Arithmetic Compare Push Buttons	6	0	0:	0: Bank 1	T_0	T_1	T_2	T_3	T_4	T_5	
	6	0		1: Bank 2	T_6	T_7	T_8	T_9	T_{10}	T_{11}	
Skip On Flags	6	0	1:	0:	Skip unconditionally	SNSZ Skip if not in Sector 0	SPBH1 Skip if PBHIT0-5 = 0	SPBH2 Skip if PBHIT6-11 = 0	Skip on $\overline{\text{LPSI}}$	Clear LPSI	
Count	6	0	1:	1:	Count Scale***	0: up 1: down	Count Intensity ***	0: up 1: down	Blink on	Blink off	
Set Slaves (Optional)	6	1	Group Number 0, 1, 2, 3		Set Unit 0	L.P.	Intensity	Set Unit 1	L.P.	Intensity	
AC Sync	7	Spares							Skip on IDLE flag	Clear IDLE flag	
Stored-Data Format Push Down Pointer	DAC Break Field	L.P.	S_0	S_1	M_0	M_1	M_2	1_0	1_1	1_2	**Low Order 12-bits from DAC

*0 - Skip if any of the indicated buttons are 0. 1 - Skip if any of the indicated buttons are 1.

**Second Word.

***Intensity and Scale will not overflow; for example, count down scale will have no effect is sc0, sc1 = 00. Count up scale will have no effect if sc0, sc1 = 11.

VC38 CHARACTER GENERATOR (OPTION) - The Type VC38 is designed for use on the Type 338 Precision Buffered Display, where it is activated by mode 5 (character mode). With the VC38, the alphanumeric characters, special symbols, etc., which make up a character set are stored in the PDP-8 memory, in increment mode or short vector mode. A 6-bit (or 7-bit, if desired) character code is used to locate a word in a dispatch table in memory.

If a 6-bit character code is being used, two characters may be specified by a single word in character mode. If a 7-bit code is used, only one character may be specified per word.

The starting address register is used to specify the location of the dispatch table and characters in memory. The SAR may be loaded by means of control characters, and also by an IOT instruction from the PDP-8 accumulator. The SAR represents the most significant six bits of a 15-bit memory address. The character code, obtained from the dispatch word, represents the least significant six (or seven) bits. A seventh SAR bit, corresponding to the octal position 100, is used with 6-bit characters, as a case bit, and may be set or cleared with a control character.

Words in the dispatch table are either display control characters (Bit 0 = "1") or specify the address at which the character routines start (Bit 0 = "0").

CHARACTER WORDS - When bit 0 = "0" indicating a character, bit 1 will equal "0" if the character is to be drawn in increment mode and "1" if drawn in short vector mode. The last nine bits of the starting address of the character routine are in bits 3 through 11. The address bit corresponding to 1000_8 is common to both the SAR and bit 2 of the dispatch word and may be specified in either or both places.

CONTROL CHARACTER WORDS - These permit changing intensity, scale, light pen status and other registers in the display and VC38.

Format for the Control Characters on the 338:

Bits 0 through 2

100:

bit 3 = 1, set scale bits
bits 4, 5, bits for SC0, SC1
bit 6 = 1, set light pen flag
bit 7, disable/enable light pen flag to cause a program interrupt
bit 8 = 1, set intensity bits
bits 9 through 11, bits for I0, I1, I2

NOTE: This format for bits 3 through 11 is the same as the parameters control state instruction on the 338 Display.

Bits 0 through 2

101:

bit 3, case bit
bit 4 = 1, set SAR 0 through 2 from bits 6 through 8
bit 5 = 1, set SAR 3 through 5 from bits 9 through 11

Bits 0 through 2

110:

bit 3 = 1, set code size (CHSZ)
bit 4 = CHSZ
bit 5 = 1, carriage return (clear X register)
bit 6 = 1, escape to control state
bit 7 = 1, count scale
bit 8 = 0, scale count up; = 1, scale count down
bit 9 = 1, count intensity
bit 10 = 0, intensity count up; = 1, intensity count down
bit 11, unused

IOT - The PDP-8 uses IOT's to initialize the VC-38 and read its status. Formats for the IOT's are as follows.

IOT303 (Initial Conditions to VC38):

C(AC):

0	1	2			5						
			CASE	CHSZ		SAR0	SAR1	SAR2	SAR3	SAR4	SAR5

CASE: When using 6-bit characters, the case bit is used as a seventh bit to allow referencing of either the lower or upper set of 64 dispatch words.

CHSZ: If this bit is 0, the character generator operates with 6-bit characters, packed two to a PDP-8 word. If the CHSZ bit is a 1, the character generator operates with 7-bit characters located in bits 5 through 11 of the PDP-8 word. In this mode, the case bit is unused.

IOT304 (VC38 status to AC):

NOTE: AC must be cleared before using this IOT, since the VC38 status is loaded into the AC by an OR transfer.

BIT:

0	1	2	3	4	5	6	7	8	9	10	11
CH ACT	CB		CASE	CHSZ		SAR0	SAR1	SAR2	SAR3	SAR4	SAR5

CHACT: If this bit is a 1, the VC38 character generator is active; that is, a character is being drawn or a control character is being executed.

CB: If this bit is a 0, the left (first) character of a data word is being executed; if it is a 1, the right (second) character is being executed. If CHSZ is set to 1 (7-bit character), the CB bit will be 1 whenever the CHACT bit is set.

SPEED OF OPERATION - A typical alphanumeric character is drawn in approximately 50 μs, including the time necessary to read data from memory. This time varies depending on the complexity of the character and on the ratio of intensified to non-intensified increments needed to describe the character. (Non-intensified steps take only 0.3 μs; intensified steps approximately 1.1 μs.)

Control characters are executed in approximately 8 to 10 μs, with the exception of carriage return which takes approximately 40 μs.

During the drawing of a character, the display address counter (DAC) is not used. It holds the location following that from which the character codes where obtained in the text string. The character address counter (CHAC) is used, and is gated to the address lines and indicator lamps when drawing the character. The CHAC is loaded from the SAR, the case bit (for 6-bit characters) and the code itself. This 15-bit location contains the starting address of the character routines for the character identified by the code.

VF-38 SEARCH LOGIC (OPTION) - The VF-38 searches the display file for increment mode, vector mode and/or POP instructions. This option is useful when these instructions must be found quickly so they can be modified.

Searching for increment words and for vector words is selected by IOT with bits 0 and 1 of the PDP-8 AC specifying which items are of interest--AC0: INC MODE; AC1: VECTOR, SHORT VECTOR, VECTOR CONTINUE MODES. When searching has been initiated, the display will stop and request an interrupt when a data word in the selected mode is read into the display input register, or if a POP instruction is read into the display input register. Jumps to subroutines are passed over.

Skip instructions test whether "data" or a "pop" has been found. A "resume" instruction causes the display to fetch the next item in the data table for inspection.

VS-38 SLAVE MODE (OPTION) - The VS-38 provides the control state with a slave mode which can operate up to seven slave displays from a 338 Display Processor. All the slaves and the main scope can display the same, different or any combination of information at the same time. Type 343 Slave Displays are used as the slave screens and each requires a driver. Control of the slaves is performed through the use of control state instructions. Each slave may have its own light pen but there can be only one function box per system.

VZ-38 ZOOM MODE (OPTION) - The VZ-38 provides the display operator with a manual switch which compresses an entire 75 in. x 75 in. drawing into a 9-3/8 in. square which can be displayed on the screen. This is accomplished by using the uppermost 10 bits of the 13-bit position register for beam deflection instead of the low-order 10 bits.

INTERFACES (OPTION) - Interfaces can be provided to the following as special systems.

Dataphone

Tablet:
B B & N Graficon

Trackballs

Joysticks

Other Processors
PDP-10
IBM 7090, 94 and 360
CDC 3000 and 6000 series

OTHER OPTIONS - The 338 Display may include any standard PDP-8 option. Complete lists of these options may be found in the PDP-8 Price List and the PDP-8 Users Handbook, Doc. No. F-85.