

**TEK SPS BASIC
DPO Driver Package
VO1
CP71171**

TEKTRONIX

TEKTRONIX®



TEKTRONIX®

**TEK SPS BASIC
DPO Driver Package
VO1
CP71171**

TEKTRONIX

INSTRUCTION MANUAL

Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077

Serial Number _____

SOFTWARE PRODUCT WARRANTY

Unless otherwise provided, Tektronix warrants this software product for one year from the date of installation. Any problem which the customer diagnoses as a software defect can be reported by the customer in a Software Performance Report and submitted to Tektronix, Inc. For problems occurring in current, unaltered releases of software, Tektronix will respond to Software Performance Reports via the Software Maintenance Newsletter. This newsletter will be provided at no cost to the owner of this software product for one year from the date of product installation. The Software Maintenance Newsletter will contain information for correcting or bypassing verified problems and will give notice of availability of corrected software.

Software updates released by Tektronix, Inc., within one year from product installation will be provided to the customer at no charge on the standard distribution media specified by the applicable software data sheet. If media other than the standard distribution media is requested, the customer will only be charged the current cost of the optional media.

LIMITATION OF WARRANTY

Any attempt by any person other than an authorized representative of Tektronix to repair or service this software product or to make any modifications to hardware or software, without the express prior written consent of Tektronix, will void all warranties. Tektronix makes no other warranties, express or implied, and specifically disclaims all warranties of merchantability or fitness for a particular purpose. Tektronix further disclaims any warranty that use of this product in conjunction with products supplied by buyer does not infringe any patent. Tektronix specifically excludes all special incidental and consequential damages for breach of any warranty. Tektronix' sole obligation for breach of any warranty shall be to repair or replace as set forth above.

SOFTWARE LICENSE

This software product, including subsequent improvements or updates, is furnished under a license for use on a single controller. It may only be copied, in whole or in part (with the proper inclusion of the Tektronix copyright notice on the software), for use on that specific controller.

Specification and price change privileges are reserved.

Although the material in this manual has been thoroughly edited and checked for accuracy, Tektronix, Inc., makes no guarantees against typographical or human errors. Also, Tektronix assumes no responsibility or liability, consequential or otherwise, of any kind arising from misinterpretation or misuse of the material in this manual. The contents of this manual are subject to change without notice.

Copyright © 1976 by Tektronix, Inc., Beaverton, Oregon. Printed in the United States of America. All rights reserved.

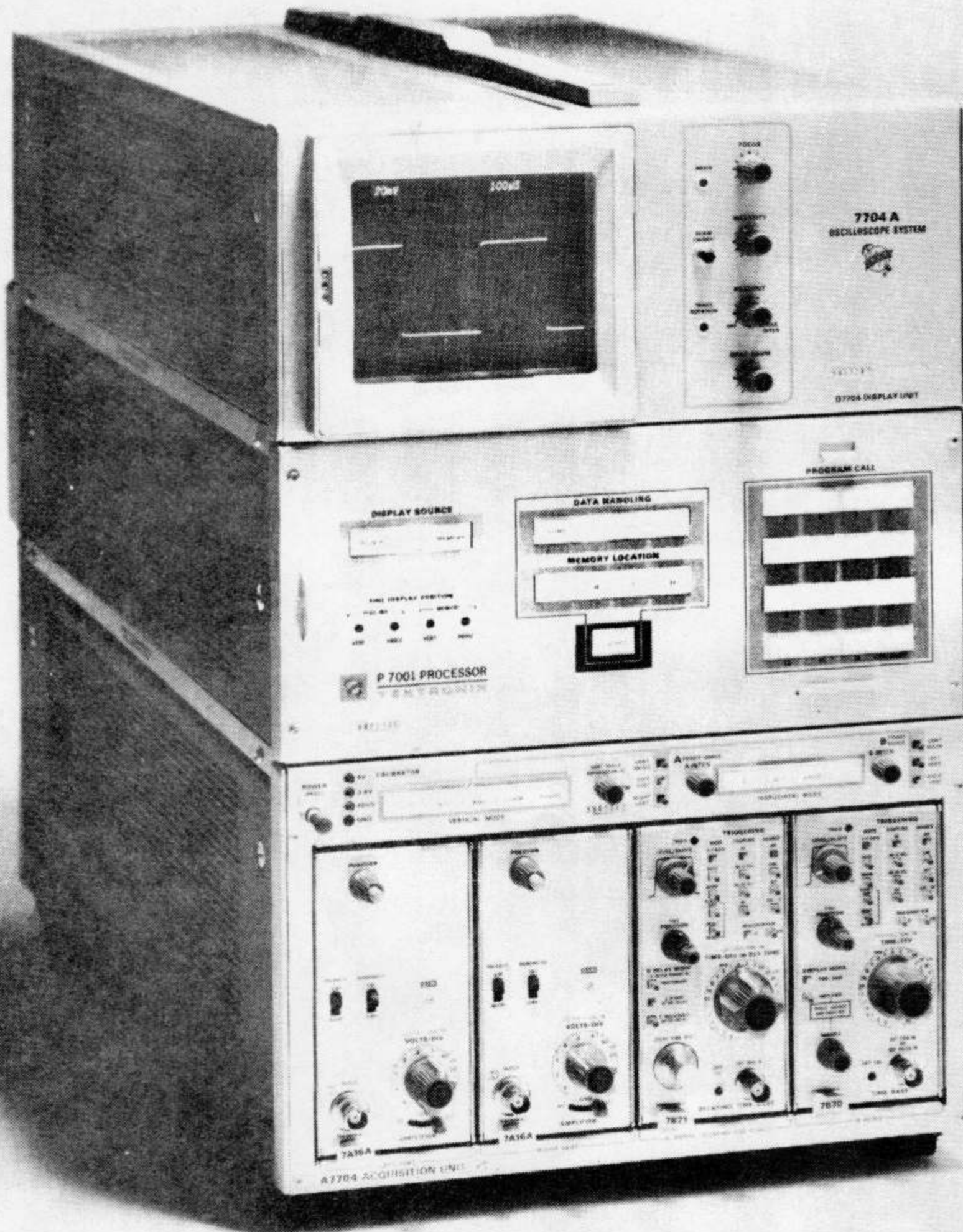
U.S.A. and foreign TEKTRONIX products covered by U.S. and foreign patents and/or patents pending.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

DPO DRIVER PACKAGE

TABLE OF CONTENTS

SECTION 1 - INTRODUCTION TO DPO DRIVER	1-1
SECTION 2 - THE DIGITAL PROCESSING OSCILLOSCOPE	2-1
SECTION 3 - THE DPO DRIVER	3-1
Loading the DPO Driver	3-1
Putting Instruments On-Line	3-2
Waveform Acquisition and Transfer	3-4
Storing Waveforms with P7001 Pushbuttons	3-4
Waveform Storage with the PUT Command	3-8
Acquiring Waveforms Into Controller Memory	3-11
Sending Data to the DPO	3-12
DPO Signal Averaging	3-13
Reading the A/D Converter	3-16
Reading and Setting DPO Status	3-17
Reading the Status of the Single Sweep	3-17
Arming the Single Sweep	3-18
Selecting the DPO Display Source	3-18
Recognizing DPO Interrupts	3-19
Display and Acquisition of ASCII Text	3-22
Displayable Characters	3-24
Displaying Messages on the DPO CRT	3-24
Labeling DPO Memory Locations	3-25
Acquiring DPO Readout Information	3-26



1599-07

The Digital Processing Oscilloscope.

DPO DRIVER PACKAGE

SECTION 1

INTRODUCTION TO DPO DRIVER

This manual describes the Digital Processing Oscilloscope Driver for TEK SPS BASIC software and is meant to accompany the TEK SPS BASIC Software System manual for systems including one or more DPOs. TEK SPS BASIC is a modular software containing a system BASIC programming language, which all systems include. This system BASIC is described in the TEK SPS BASIC Software System manual. The TEK SPS BASIC Software System manual also describes certain non-resident commands which control the instruments and peripheral I/O devices. A basic understanding of these commands will help in understanding this manual.

NOTE

The DPO (DPO.SPS) and R7912 (TD.SPS) drivers should not be used simultaneously unless each type of instrument is confined to separate CP1100/CP Bus Interface cards (inside the controller). This is because the DPO and R7912 use the same interrupt vectors.

The DPO driver can individually communicate with up to four DPOs cabled to a system. This communication may be summarized as:

- 1) reading the status
- 2) setting the status
- 3) acquiring waveforms in floating-point format
- 4) acquiring ASCII readout information
- 5) sending waveforms to the DPO memory for display
- 6) displaying ASCII text on the DPO CRT
- 7) signal averaging

Before discussing the detailed aspects of each one of these operations, let's take a brief look at how the DPO works. This will aid your understanding of each of the preceding operations.

SECTION 2

THE DIGITAL PROCESSING OSCILLOSCOPE

The Digital Processing Oscilloscope is a computer compatible oscilloscope. This compatibility comes from taking a standard Tektronix 7704A Oscilloscope, separating the Acquisition and Display Units, and reassembling them with a P7001 Processor placed between them. Fig. 2-1 illustrates this. Except for the P7001, the DPO is operated just like a conventional oscilloscope. Hence the operation of the 7704A oscilloscope mainframe and its plug-ins are not discussed here.

The P7001 Processor serves as a three-way interface between the two portions of the 7704A Oscilloscope and the controller (minicomputer). On its one arm, it accepts inputs from the acquisition portion of the oscilloscope so that waveforms can be digitized and stored in memory. On its other arm, it directs outputs to the display portion of the oscilloscope so that digitized and processed waveforms from the controller can be displayed on the CRT. The two "arms" of the P7001 join to provide a two-way communication link with the controller. The result is a computerized 175-megahertz oscilloscope.

With the aid of the block diagram in Fig. 2-2 let's take a closer look at the P7001 Processor to see how it stores, transfers, and displays data. First, the Signal Interface block acts as a traffic interchange for signal paths. This interface provides a path for analog waveforms from the A7704 Acquisition Unit to the Sample and Hold circuitry and the Analog-to-Digital Converter. It also provides a path for analog waveforms to the D7704 Display Unit, where they can be displayed in the normal oscilloscope format. Furthermore, a path exists through the Signal Interface for digitized waveforms received from the P7001 Memory via the Display Generator.

Waveforms from the plug-in units exist in the form of an amplified analog signal. The DPO converts this to a digitized waveform that can be processed by the controller. The first step in digitizing is accomplished by the Sample and Hold circuit.

The Sample and Hold circuit samples waveforms at a 6.5 microsecond rate and outputs a horizontal and vertical analog value for each sample. The vertical value is the instantaneous voltage appearing at the Y-axis input. The horizontal value is the instantaneous value of the ramp or X-axis voltage.

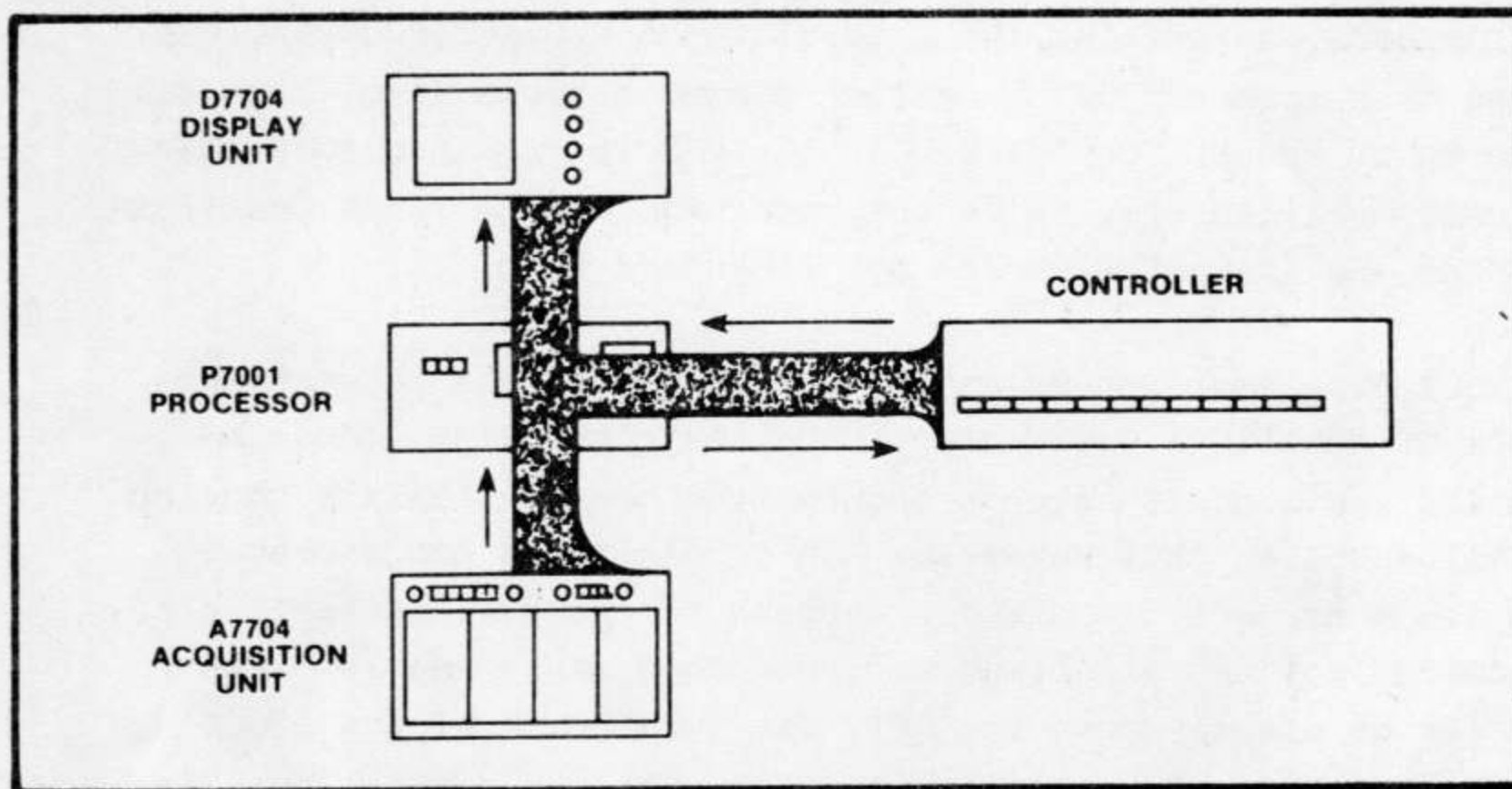


Fig. 2-1. The P7001 Processor gives compatibility to the 7704A Oscilloscope.

To fully represent one trace on the DPO CRT, it is necessary to sample 512 points. On sweep speeds slower than 0.5 ms/div, all 512 samples can usually be taken during one sweep on the DPO, thereby providing real-time sampling. Further sweeps merely provide updated samples. However, on sweep speeds faster than 0.5 ms/div all 512 samples cannot be taken during one sweep. Thus, sampling of repetitive waveforms is done on an equivalent-time basis. In this case, sampling occurs at different points on the trace, and full sampling is built up over several sweeps.

DPO DRIVER PACKAGE

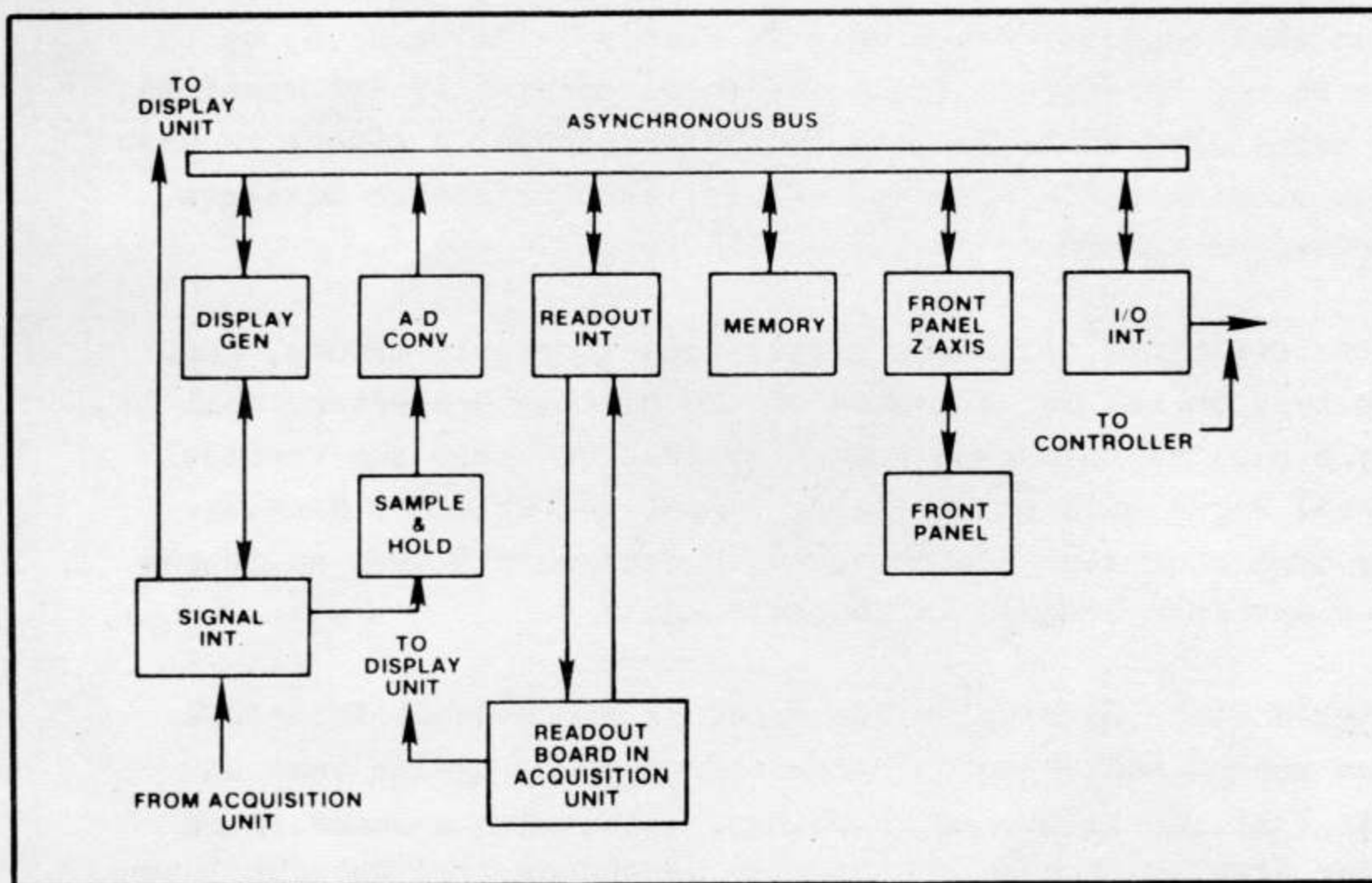


Fig. 2-2. P7001 Processor block diagram.

As each sample is taken, the Analog-to-Digital Converter digitizes the sample. The vertical analog value is converted to a 10-bit word in the range of 0 to 1777 octal. The software later converts this value to a base-10 number in the range of 0 to 1023. Point 0_{10} corresponds to one division below the bottom CRT graticule line and point 1023_{10} corresponds to one division above the top graticule line. The horizontal analog value of the sample is converted to a 9-bit word in the range of 0 to 777 octal. Again, the software converts this value to a base-10 number in the range of 0 to 511. In this case, point 0_{10} corresponds to the left-hand graticule line and point 511_{10} corresponds to the right-hand graticule line.

DPO DRIVER PACKAGE

After the horizontal and vertical values of each sample have been converted to digital words, they are sent to the Memory board of the P7001. The horizontal word will become the address (location) at which the value is stored, and the vertical word will be the data value that is stored in that address. Assuming a 4K memory is being used, up to four waveforms may be stored. These waveforms consist of 512 mantissa values and associated scale factors and units. Space in memory is also reserved for storing zero-reference values, zero-reference messages, and other messages generated by the controller.

Once the digitized data values have been stored in memory, they can be displayed on the CRT by means of the Display Generator. This is essentially a Digital-to-Analog Converter that converts the vertical and horizontal words back to an analog format suitable for display. The Display Generator also vectors between data points when an incomplete set of waveform samples is encountered.

Associated with the Display Generator is the Readout Interface. It processes analog scale-factor signals from the plug-ins into an ASCII format that can be stored in memory. Also, when a waveform is called up for display through the Display Generator, the Readout Interface converts the ASCII scale-factor values back to analog values for display on the CRT.

In most cases, it is necessary to send the digitized waveform values to the controller for processing. This transfer is accomplished by the Input/Output interface, more specifically known as the DPO/CP Bus Interface. It provides the control link between the P7001 Processor and the controller. Thus values can be sent to the controller for processing, and the processed values can be received by the DPO for display.

Data transfer between the controller and DPO can be initiated by either the terminal or the Front-Panel control of the P7001. This P7001 circuit board contains all of the pushbuttons on the P7001 front panel. It monitors which buttons are pressed, sets the DPO status accordingly, and lights the appropriate buttons to indicate the DPO status. The purpose of these P7001 front-panel buttons will be discussed shortly.

DPO DRIVER PACKAGE

All of the blocks in the P7001, except the Signal Interface and the Sample and Hold circuit are connected directly to an Asynchronous Bus. This bus allows each block to interactively work with other blocks on the bus so that various P7001 operations can be performed.

The DPO front-panel buttons allow you to store a waveform, to select the display source (either from the plug-ins or a particular memory location), to send a waveform to the controller for processing, or to receive a waveform for display on the DPO. The DPO Program Call buttons also allow you to generate interrupts which can be used to execute various programs in memory. Each of these operations will be discussed in the next section.

DPO DRIVER PACKAGE

SECTION 3

THE DPO DRIVER

Loading the DPO Driver

The first step in DPO communication is to load the DPO driver into controller memory from the peripheral device on which it's stored. This is done with the LOAD command which has the following format:

LOAD [device [drive no.]:] / $\begin{Bmatrix} F \\ R \end{Bmatrix}$, file name [,file name] ...

Normally, the driver is stored on the system device, so the following command can be used:

LOAD "DPO.SPS" or simply LOAD "DPO"

where "DPO.SPS" is the name of the DPO driver. (If no extension is specified, the LOAD command assumes a .SPS extension.) As a second example, suppose the DPO driver is stored on the right-hand drive of the TEKTRONIX CP100 CASSETTE. Then the following command should be used:

LOAD CT1:"DPO"

where CT1 is the symbol of the right-hand drive of the CP100.

When the DPO driver is no longer needed, it may be deleted from memory by executing:

RELEASE "DPO"

Notice that the .SPS extension is assumed if none is specified.

Putting Instruments On-Line

Normally, your system will have been cabled and installed on site by a Tektronix field engineer. In this case, the system should be operational as soon as the master power switch is turned on. Even so, it is necessary to understand a few facts about the strapping options which allow the controller (and software) to communicate with a particular DPO.

Since TEK SPS BASIC supports an instrument configuration with up to four DPOs, there must be some means of selecting a particular instrument via both the hardware and software. In terms of hardware, this instrument selection is facilitated by a number in the range of 0 to 31 referred to as the Hardware Unit Number (HUN). The HUN may be thought of as the name by which the controller addresses a particular DPO. Depending on interface strapping options, a DPO may be assigned any HUN in the range of 0 to 31; however, no more than one instrument (DPO or R7912) may have the same HUN. Otherwise there will be a loss of identity and resulting confusion.

The HUN is determined by two sets of straps on the CP Bus Interface system: The Group Select straps and the Device Select straps. The Group Select straps are located on the CP1100/CP Bus Interface (a circuit card inside the controller) and determine the Group number of the instrument. The Device Select straps are located on the DPO/CP Bus Interface (a circuit card inside the DPO) and determine the Device number of the instrument. The Hardware Unit Number is thus determined according to Table 3-1. (For more information on determining the Hardware Unit Number of an instrument or setting the strap options, refer to the CP1100/CP Bus Interface manual - Tektronix, Inc.)

Assuming that each instrument has been assigned to a unique HUN, there is still another step in establishing communication between the instrument and controller. Specifically, an Instrument Logical Unit Number (ILUN) must be assigned to the instrument by the ATTACH command. Until an instrument has been ATTACHED to the system, the software will not allow the controller to recognize the instrument.

DPO DRIVER PACKAGE

TABLE 3-1

Determining Hardware Unit Numbers

Group Select Strap Settings	0	0	1	2	3	4	5	6	7
	1	8	9	10	11	12	13	14	15
	2	16	17	18	19	20	21	22	23
	3	24	25	26	27	28	29	30	31
		0	1	2	3	4	5	6	7
Device Select Strap Settings									

1603-13

The format of the ATTACH command is:

ATTACH #ilun AS driver name hun:

where ilun is the Instrument Logical Unit Number and hun is the Hardware Unit Number. For example:

ATTACH #5 AS DPO0:

This command assigns Instrument Logical Unit Number 5 to the DPO whose Hardware Unit Number is 0. Hereafter, this DPO will always be referenced as #5 until it is DETACHED from the system.

The format of DETACH is simply:

DETACH #ilun

Thus the command

DETACH #5

will disassociate the instrument whose ILUN is 5. Therefore #5 could be used as the ILUN for some other instrument. When an instrument is DETACHED from the system, no more communication can occur until it is ATTACHED again.

Waveform Acquisition and Transfer

Waveform storage in the P7001 memory is accomplished with the PUT command as explained later. This operation can also be performed manually by using the STORE, MEMORY LOCATION, START, and HOLD buttons on the DPO. Assuming that the P7001 contains a 4K memory, up to four waveforms may be stored in the DPO at one time. Regardless of the method of storage, three classes of information are stored for each waveform:

- 1) 512 unscaled mantissa values.
- 2) Up to 80 characters of ASCII text corresponding to the readout for the vertical and horizontal scale-factor information.
- 3) A zero-reference value.

Before further discussing how to store waveforms under program control, let's review how to store a waveform with the P7001 pushbuttons.

Storing Waveforms with P7001 Pushbuttons. With the P7001 in the PLUG-INS mode, the first step in storing a waveform from the plug-ins is to input and display the waveform on the DPO. Once you have determined that the waveform is indeed the one you want to store, it may be stored in the P7001 by following this sequence:

- 1) Press STORE. Note that the DISPLAY SOURCE indicator lights change from PLUG-INS to BOTH, due to the hardware in the DPO.

DPO DRIVER PACKAGE

2) Press the MEMORY LOCATION button(s) corresponding to the desired P7001 storage location(s). For example: if button A is pressed, the waveform will be stored in P7001 memory location A. Allowable memory locations for multiple waveform storage are specified in the DPO Operator's Manual.

3) Press START. This begins storing values in the P7001 memory.

4) Press HOLD, after the waveform has been satisfactorily displayed on the CRT. Pressing HOLD causes a message to be displayed on the CRT asking you to provide a zero reference by grounding the probe and pressing PROGRAM CALL button 14. This zero-reference request appears for each waveform stored; it may be satisfied by performing either step 5a or 5b below.

5) a. Press PROGRAM CALL button 14 after grounding the probe. This determines the ground (zero-reference) value and stores it in the P7001 memory.

-or-

5) b. Press some other PROGRAM CALL button. This uses the previously stored zero reference as the zero reference for the newly stored waveform. It also causes the interrupt (generated by the PROGRAM CALL button) to be processed by the controller.

Step 5 must be repeated once for each waveform stored.

In the above described sequence of P7001 pushbutton operations, pressing START causes the selected memory location to be cleared of non-zero mantissa values. The existing scale-factor information is overwritten by the scale-factor information for the waveform being stored. (The previously stored zero reference is left undisturbed until later options are taken.) More importantly, the waveform is pseudorandomly sampled at a 6.5 microsecond rate (under optimum conditions), and the resulting horizontal and vertical amplitude samples are digitized.

Each horizontal sample that occurs corresponds to one of 512 possible mantissa addresses for the chosen storage location. These mantissa addresses are arranged so that the first address (P7001 array element 0) corresponds to the left-hand CRT graticule line, and the last mantissa address (P7001 array element 511) corresponds to the right-hand CRT graticule line.

As each pseudorandom sample is taken during the horizontal sweep time, the vertical amplitude of the waveform at the sample point is digitized and expressed as a 10-bit unscaled mantissa value. This mantissa value represents the waveform's amplitude value relative to a digitizing reference that exists one division below the bottom CRT graticule line. There are 1024 possible amplitude levels available for expression of the mantissa value, and the mantissa values are stored in the P7001 as unscaled integers in the range of 0 to 1023. (The stored vertical scale factor provides vertical scaling information.) A mantissa value of 0 corresponds to one division below the bottom CRT graticule line, and 1023 corresponds to one division above the top CRT graticule line. (See Fig. 3-1.)

When HOLD is pressed, the sampling and digitizing ceases, and the existing digitized samples are fixed in the selected P7001 memory location. Also, after pressing HOLD, a software generated request for zero reference appears on the DPO CRT. When this request is acknowledged by grounding the probe and pressing PROGRAM CALL button 14 (step 5a), 32 samples of the provided ground reference are sampled and digitized. The average of these 32 samples is then used to replace the previously existing zero-reference value. If step 5b is executed instead, the existing zero-reference value is retained.

When more than one trace is stored (either manually or under program control, the zero-reference level must be acquired for each memory location from which data is stored. For each trace stored, the CRT displayed zero-reference request must be followed precisely so that the correct zero-reference values are individually collected and stored with their respective waveforms.

DPO DRIVER PACKAGE

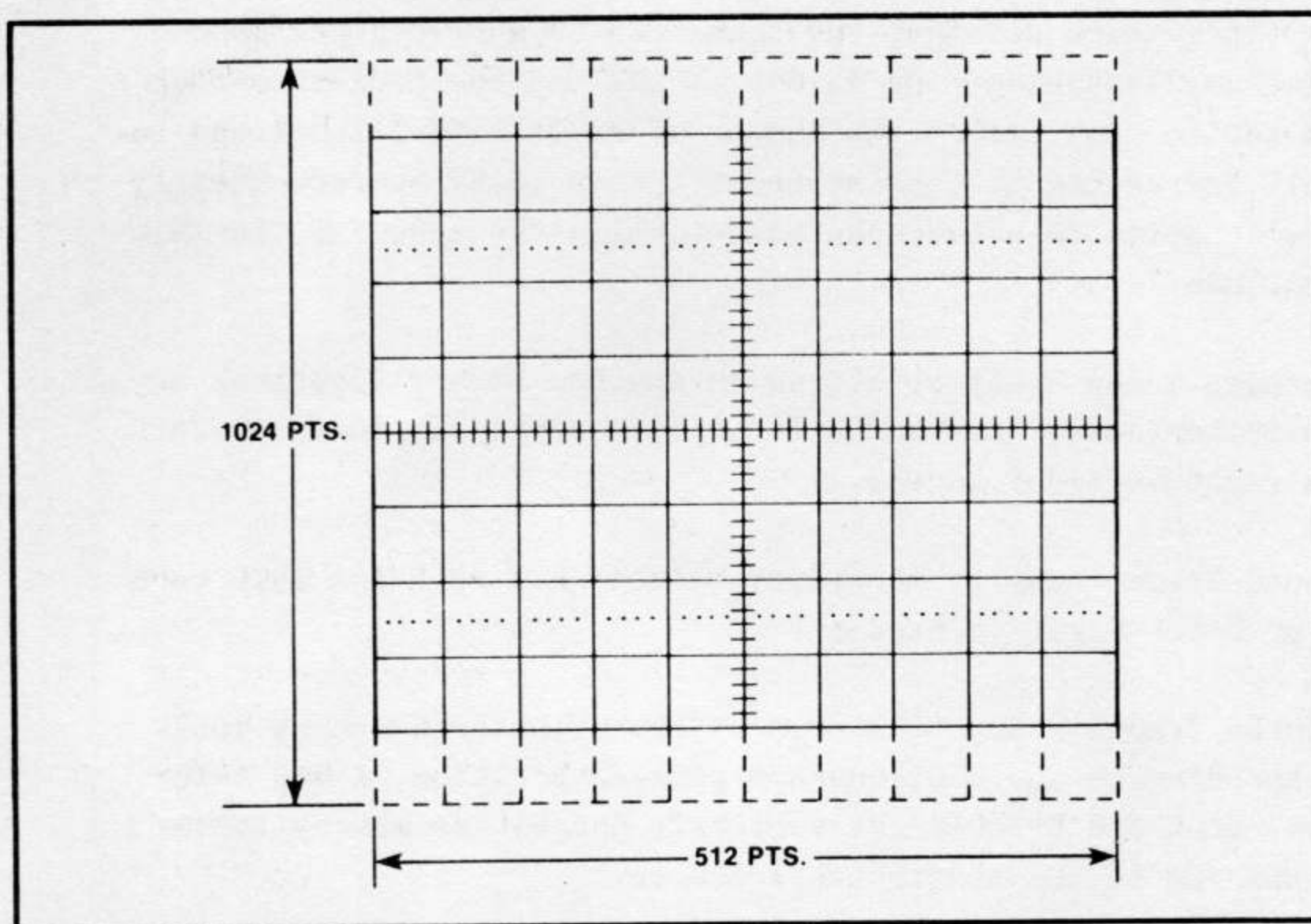


Fig. 3-1. Diagram of DPO CRT. Top and bottom divisions shown with dotted lines are not displayed.

For example, when multiple waveforms are stored with the DPO, a message of the following type will appear on the CRT.

TO STORE ZERO REF FOR MEM LOC A SELECT
PROPER CHANNEL. GROUND PROBE. PRESS 14.

For this message, you would set the DPO mainframe to LEFT vertical mode and set the left vertical plug-in to CHANNEL 1 mode if it is a dual-trace unit. Then ground the probe and press PROGRAM CALL Button 14. The zero reference for each memory location must be supplied with the mainframe VERTICAL MODE buttons (located at the top of the A7704) and plug-in DISPLAY MODE selectors (located on dual-channel plug-ins only). (They are set solely for the waveform stored in the memory location being supplied with zero-reference information.)

DPO DRIVER PACKAGE

For multiple-trace storage, the best results are obtained when the DPO mainframe switching mode is set to ALT and the plug-in switching mode is set to CHOP. Also, the number of P7001 memory locations selected should correspond to the number of traces being stored. Exactly which memory location to select can be determined by applying the following guidelines:

- 1) Single Trace - may be stored in any one memory location: may be stored simultaneously in A and B for a left vertical source or in C and D for a right vertical source.
- 2) Dual Trace - may be stored by selecting A or B for left vertical and C or D for right vertical.
- 3) Three Traces - may be stored by selecting both memory locations corresponding to the dual-channel plug-in position, A and B for left vertical or C and D for right vertical, and either memory location corresponding to the single trace source.
- 4) Four Traces - may be stored by selecting all memory locations simultaneously. Left vertical CH 1 will go to A and CH 2 to B. Right vertical CH 1 will go to C and CH 2 to D.

These rules should be followed for both multiple-trace storage of multiple-trace signal averaging (explained shortly).

Waveform Storage with the PUT Command. The just described sequence of pushbutton operations may be performed under program control with the PUT command. To initiate the STORE operation in the DPO, the following format of PUT may be used:

$$\text{PUT "STO" INTO \#ilun, } \left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \left[, \left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \right] \dots "$$

DPO DRIVER PACKAGE

This command addresses the DPO whose Instrument Logical Unit Number is *ilun* and sets all four status registers (front panel, A/D, readout, and display generator) to the STORE mode. The second argument in quotes specifies which memory location(s) (A, B, C, and/or D) are to be set. For example:

100 PUT "STO" INTO #5,"A,C"

This command sets memory locations A and C of DPO #5 to the STORE mode. The 512 waveform samples will continuously be updated until DPO memory locations A and C are set to the HOLD mode. The DPO front panel lights will indicate that waveforms are being stored in memory locations A and C.

To display a DPO memory location or to stop the storing of a waveform, set the corresponding memory location(s) to the HOLD mode. This is done with the following format of the PUT command:

PUT "HOL" INTO #*ilun*, " $\left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \left[, \left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \right] \dots$ "

This command addresses the DPO whose Instrument Logical Unit Number is *ilun* and sets all four status registers to the HOLD mode. The DPO front panel lights will indicate which memory locations are being held for display. For example:

110 PUT "HOL" INTO #5,"A,C"

This command sets memory locations A and C of DPO #5 to the HOLD mode, thereby terminating the storage process begun at line 100 above. However, the waveform will be stored without regard for the zero-reference level. (That is, the previous zero-reference level will be used.)

DPO DRIVER PACKAGE

If you want to store a waveform corrected for the new ground-reference level, execute the following command format in place of PUT "HOL":

PUT "ZHO" INTO #ilun, " $\left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \left[\begin{matrix} \left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\} \end{matrix} \right] \dots$ "

If the PUT "STO" command was previously issued and this command is executed, the storing ceases and a message requesting a zero reference is displayed on the DPO CRT (assuming the DPO is in the PLUG-INS or MEMORY display mode). To store a zero reference, ground the probe (or the plug-in) and press PROGRAM CALL button 14 on the DPO. This button should be pressed once for each memory location to be stored into. If you press any other button that can cause an interrupt, the previous zero reference is kept and the message is erased. For multiple-trace storage, the previously mentioned rules apply (see heading "Storing Waveforms with P7001 Pushbuttons").

In general, it is best to put a delay between the STORE and HOLD operations. The delay should be long enough to allow the waveform to be stored in its entirety before the HOLD process starts. The WAIT command is ideally suited for this purpose and is demonstrated in the following acquisition program.

```
200 PUT "STO" INTO #5,"C"
210 WAIT 500
220 PUT "MEM" INTO #5
230 PUT "ZHO" INTO #5,"C"
```

Line 200 puts DPO #5 into a mode which stores the input waveform into memory location C. (The PLUG-INS and STORE:C lights of the DPO are lit to indicate this condition.) Line 210 causes the controller to pause approximately 500 milliseconds before executing line 220. Line 220 sets the DPO display source to MEMORY so that the zero-reference request message (generated at line 230) can be seen. This format of PUT will be further explained later. When line 230 is reached, the

storage process ceases as indicated by the lighting of the HOLD button. Also, since a "ZHO" argument was specified, a message is displayed on the DPO CRT requesting the user to supply a ground-reference level. When PROGRAM CALL button 14 is pressed, the ground-reference level is acquired and used to adjust the waveform values.

Acquiring Waveforms into Controller Memory. After a waveform has been stored in DPO memory, it may be transferred to a controller array or waveform with the GET command. The format of GET when used for this purpose is:

$$\text{GET } \left\{ \begin{array}{c} \text{array} \\ \text{waveform} \end{array} \right\} [, \left\{ \begin{array}{c} \text{array} \\ \text{waveform} \end{array} \right\}] \dots \text{FROM } \#ilun, \left[\begin{array}{c} \left(\begin{array}{c} A \\ B \\ C \\ D \end{array} \right) \\ \left(\begin{array}{c} A \\ B \\ C \\ D \end{array} \right) \end{array} \right] \dots$$

The destination argument must either be a 512-element floating-point array or 512-element floating-point waveform. The "ilun" argument specifies the Instrument Logical Unit Number of the DPO whose data is to be acquired. The string expression following the ilun selects the memory locations whose data is to be acquired. Since there is a one-to-one correspondence between destination arrays or waveforms and the source memory locations, there must be an equal number of each.

Each of the destination arrays or waveforms are filled with "normalized" data. This means that any points not stored are interpolated or extrapolated to provide a complete set of waveform samples. Then the zero-reference is subtracted from each sample and the resulting values are multiplied by the vertical scale factor. Furthermore, if the destination argument is a waveform, the data sampling interval, horizontal units and vertical units are returned in appropriate variables.

As an example of waveform acquisition, let's suppose that memory locations A and C of DPO #5 contain waveforms. To read these waveforms into controller memory, the following routine could be used:

DPO DRIVER PACKAGE

```

300 WAVEFORM WA IS A(511),SA,HA$,VA$
310 WAVEFORM WC IS C(511),SC,HC$,VC$
320 GET WA,WC FROM #5,"A","C"
330 GRAPH WA,WC
340 END

```

Line 300 defines waveform WA to be: a 512-element array A, its data sampling interval SA, and its horizontal and vertical units HA\$ and VA\$, respectively. Similarly line 310 defines waveform WC to be: a 512-element array C, its data sampling interval SC, and its horizontal and vertical units HC\$ and VC\$. Notice that arrays A and C are dimensioned by the WAVEFORM commands. With waveforms WA and WC now defined, line 320 addresses DPO #5 and GETs the waveforms stored in memory locations A and C. The waveform in memory location A is read into array A in the controller, and the waveform in memory location C is read into array C in the controller. The units and data sampling intervals are also acquired in the process. Finally, line 330 GRAPHS waveforms WA and WC on the terminal screen, assuming you have the GRAPH command in your software system. (Since GRAPH is a non-resident command, it will be auto-loaded unless it was previously loaded into memory.) The graph will reflect the acquired units and data sampling intervals.

Sending Data to the DPO. The PUT command can be used to transfer an array or waveform to the DPO. When used for this purpose, the format of PUT is:

$$\text{PUT } \left\{ \begin{array}{l} \text{array} \\ \text{waveform} \end{array} \right\} [, \left\{ \begin{array}{l} \text{array} \\ \text{waveform} \end{array} \right\}] \dots \text{INTO } \#ilun \left[\begin{array}{c} \left(\begin{array}{c} A \\ B \\ C \\ D \end{array} \right) \left(\begin{array}{c} A \\ B \\ C \\ D \end{array} \right) \end{array} \right] \dots$$

The source array must be either a 512-element floating-point array or 512-element floating-point waveform. The "ilun" argument specifies the Instrument Logical Unit Number of the DPO whose memory location(s) will receive data. The string expression(s) following the ilun is/are the memory location(s) which serve as the destination(s) for the transferred data. For example:

DPO DRIVER PACKAGE

```
PUT WA,WC INTO #5, "B","D"
```

This statement addresses the DPO whose Instrument Logical Unit Number is 5. It then sends waveform WA to memory location B and waveform WC to memory location D. Memory locations B and D could then be displayed with the statements:

```
PUT "MEM" INTO #5 \ PUT "HOL" INTO #5,"B","D"
```

These statements are concatenated with a backslash character. The first statement sets the DPO Display Source to MEMORY and the second statement sets Memory Locations B and D to HOLD.

DPO Signal Averaging. TEK SPS BASIC provides two modes of stable averaging for DPO waveforms - a normal mode and a fast mode. The normal mode of signal averaging clears each sample from the P7001 Processor after it has been added to the average. Then it checks the next sample to ensure that it is valid (i.e., the waveform has been sampled and a nonzero sample provided) before adding it to the average. The fast averaging mode does not check the validity of each sample; hence, it executes faster than normal averaging. However, each point in fast averaging may deviate from the true signal amplitude at that point, following the averaging. Thus, the fast-average mode should be used for quick previews only.

To signal average waveforms from a DPO, a zero reference should be stored in each memory location in which data is averaged. This may be done by first storing the signal to be averaged in one of the P7001 memory locations. Then, following the message displayed on the DPO CRT at the time of waveform storage, ground the probe and press PROGRAM CALL Button 14.

Normal averaging requires that the waveform being averaged occupy a full ten centimeters horizontally on the CRT of the D7704 DISPLAY UNIT. If it does not, the averaging operation will not complete, and a restart from location 0 will be required for an exit from the averaging routine. If the real-time waveform display is not occupying the full ten centimeters of graticule, the horizontal position control on the Time-Base plug-in should be adjusted to provide the required display.

DPO DRIVER PACKAGE

With the zero-reference value(s) stored and the real-time waveform applied to selected vertical plug-in(s) of the DPO, the average operation may be begun. An averaged DPO signal is obtained with the GET command as follows:

$$\text{GET } \left\{ \begin{array}{l} \text{array} \\ \text{waveform} \end{array} \right\} \left[, \left\{ \begin{array}{l} \text{array} \\ \text{waveform} \end{array} \right\} \right] \dots \text{FROM } \# \text{ilun "AVEn[F], } \left\{ \begin{array}{l} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \end{array} \right\} \left[, \left\{ \begin{array}{l} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \end{array} \right\} \right] \dots "$$

The destination arrays are either 512-element floating-point arrays or 512-element floating-point waveforms. The data is acquired from the DPO whose Instrument Logical Unit Number is given by the "ilun" argument. The "n" in the source argument specifies the number of averages; it may range from 1 to 32767. Up to four signals can be simultaneously averaged depending on which combinations of memory locations (A through D) are specified. If the "F" is included in the string expression, a fast average is performed. This means that the driver does not wait for the DPO to store a point before averaging that point - as explained previously.

The average operation sets the DPO to store mode and returns the DPO to the mode it was in before the start of the average. The data returned is normalized. Also, if a waveform is specified as the destination, the data sampling interval, horizontal units and vertical units are returned.

The average operation uses the previously stored zero-reference value in normalizing the averaged data. Thus, you will probably want to store a zero-reference value into the memory location(s) whose data is to be averaged. For example, the following program stores a zero-reference value in memory location C and then averages data (16 times) via that memory location. This program assumes that the DPO driver has been LOAded and that a DPO is ATTACHed as Instrument Logical Unit Number 1. Here is the program:

DPO DRIVER PACKAGE

```
10 REM PROGRAM TO ACQUIRE AN AVERAGED WAVEFORM
20 PUT "PI" INTO #1
30 PRINT "INPUT WAVEFORM TO VERTICAL PLUG-IN; WAVEFORMS MUST SPAN 10 DIVS"
40 PRINT "UNGROUND PROBE AND ADJUST KNOBS FOR PROPER SENSITIVITY"
50 PRINT "PRESS 'RETURN' WHEN READY"
60 WAIT/PAGE
70 WAVEFORM WC IS C(511),SC,HC$,VC$
80 PUT "STO" INTO #1,"C"
90 PUT "MEM" INTO #1
100 PUT "ZHO" INTO #1,"C"
110 PRINT "TO ACQUIRE GROUND-REFERENCE LEVEL, GROUND PROBE OR PLUG-IN"
120 PRINT "PRESS 'PROGRAM CALL' BUTTON 14 ON DPO - PRESS 'RETURN' WHEN READY"
130 WAIT/PAGE
140 PUT "PI" INTO #1
150 PRINT "TO ACQUIRE AVERAGED WAVEFORM, UNGROUND PROBE OR PLUG-IN"
160 PRINT "PRESS 'RETURN' WHEN READY"
170 WAIT/PAGE
180 GET WC FROM #1, "AVE16,C"
190 GRAPH WC
200 END
```

The program begins by explaining the purpose of the program and setting the DPO to PLUG-INS display mode. Lines 30 through 50 instruct the user to input the waveform and set up the DPO for proper triggering and display. After the WAIT is exited by a keyboard interrupt, the screen is erased and waveform WC is defined. At line 80, the DPO is set to store in Memory Location C, and the display source is changed to MEMORY (for display of the zero-reference request message). At line 100 the storage process terminates with the zero-reference request message. This message is reiterated on the terminal screen. After you have grounded the probe, pressed PROGRAM CALL button 14, and pressed the RETURN key, the WAIT is exited and the screen is erased.

At this point the user is instructed to unground the probe and to press RETURN when ready. This erases the screen and signal averages the input waveform 16 times into waveform WC via memory location C. Finally, waveform WC is graphed on the terminal screen.

DPO DRIVER PACKAGE

The same basic program could be modified for multiple signal averaging. The important difference is that a zero-reference level must be acquired for each memory location from which data is averaged. When multiple traces are stored, the CRT displayed zero-reference request must be followed precisely so that the correct zero-reference values are individually collected and stored with their respective waveforms. Also, the previously mentioned rules for multiple-trace storage also apply to multiple-trace signal averaging.

The question sometimes arises as to how many averages should be specified. There is no one answer to this question. However, as a rule of thumb several hundred to several thousand averages may be required to recover signals nearly buried in noise. For signals with moderate noise, 128 averages may be enough. With low noise, 32 may be all that is required. In any case, the exact number of averages required is dependent upon the desired improvement in signal-to-noise ratio.

Summing a signal M times improves the signal-to-noise (S/N) ratio by \sqrt{M} . If M can be expressed as a power of 2, then the improvement in the S/N ratio will correspond to 3dB per power of 2. TEK SPS BASIC signal averaging allows a maximum of $2^{15}=32768$ averages, which corresponds $15 \times 3\text{dB}=45\text{dB}$ maximum improvement in the S/N ratio.

Reading the A/D Converter. In some applications it is particularly advantageous to read the last value processed by the DPO's Analog-to-Digital Converter. This value can be obtained with the following command format:

```
GET {variable  
    {array element}} FROM #ilun,"AD"
```

The value returned is an integer in the range of 0 to 1023. It is acquired from the DPO ATTACHED as #ilun and is entered in the specified variable or array element.

One of the more valuable applications of this format of GET is acquiring long-term records of slowly varying phenomena. This can be done by enclosing the GET command in a FOR loop. For example, to acquire a 1024 element array X, the following routine could be used to

acquire data from DPO #1.

```
10 DIM X(1023)
20 FOR I=0 TO 1023
30 GET X(I) FROM #1,"AD"
40 NEXT I
50 END
```

The loop in this routine takes about 8 seconds to execute on a TEKTRON-IX CP1151 Controller. However, the execution time (i.e., acquisition time) can be lengthened by including a WAIT statement in the loop and specifying the desired number of milliseconds. After a period of trial and error, the loop can be adjusted to match the desired acquisition time. Of course the data returned in array X is integer data; thus you may want to normalize it and adjust it for the ground-reference level before processing it.

Reading and Setting DPO Status

Reading the Status of the Single Sweep. There is only one programmable feature of the DPO plug-ins. This is the single-sweep function on the time base. To read the status of the single sweep, use a GET command of the following format:

```
GET string variable FROM #ilun, "SWE"
```

This command GETs the sweep status from the DPO whose Instrument Logical Unit Number is given by the "ilun" argument. The status is returned in the specified string variable. The status variable may then be printed to yield one of the following:

```
SSU - single sweep unarmed
SSA - single sweep armed
```

DPO DRIVER PACKAGE

Arming the Single Sweep. The single-sweep feature of the DPO time base can be armed with a PUT command of the following format:

```
PUT "SSA" INTO #ilun
```

This command arms the sweep of the DPO whose Instrument Logical Unit Number is given by the "ilun" argument. A string variable containing "SSA" can be used in place of the string expression.

As an example, suppose you want to arm the sweep of a DPO ATTACHED as #5. This could be done with the command:

```
PUT "SSA" INTO #5
```

Following this command, you could read the sweep status with the concatenated commands:

```
GET A$ FROM #5,"SWE"\PRINT A$
```

An "SSA" or "SSU" would be printed on the terminal depending on whether the sweep was armed or unarmed at the time the GET command was executed.

Selecting the DPO Display Source. As previously demonstrated, the DPO Display Source may be selected either manually or under program control. To select the Display Source remotely, use the following format of PUT:

```
PUT " { PI } " INTO #ilun  
      { BOT }  
      { MEM }
```

This command addresses the DPO ATTACHED as #ilun and selects either the PLUG-INS (PI), BOTH (BOT) or MEMORY (MEM) Display Source. For example, the following command would address DPO #5 and select the MEMORY Display Source:

```
PUT "MEM" INTO #5
```

DPO DRIVER PACKAGE

NOTE

The PUT command also allows you to select a particular Memory Location for storage (STOre) or display (HOLd). This is explained previously under the heading "Waveform Storage with the PUT Command."

Recognizing DPO Interrupts. Various DPO-generated interrupts can be scheduled on a priority basis by the WHEN command. The format of this command is:

WHEN #ilun [HAS event][AT priority] GOSUB line number

This command schedules interrupts from the instrument ATTACHed as #ilun. The expression following "AT" specifies the priority at which the interrupt routine will be executed. It may range from a high of 126 to a low of 0. If none is specified, the default priority of the driver (51) is assumed. When the specified event occurs, the corresponding interrupt routine (starting at the specified line number) is executed if its priority is higher than the system priority. (System priority ranges from a high of 126 to a low of 0; it defaults to 50 unless modified by a PRIORIY command or an interrupting WHEN routine.) For more information on interrupt processing see Section 6 of the TEK SPS BASIC Software System manual entitled "Instrument Communications".

Like the "priority" argument, the "event" argument is optional. If none is specified, any DPO interrupt will execute the associated interrupt routine whenever system priority falls below the priority specified (or assumed by default) in the WHEN command.

Any of the following key words may be used as the string expression to specify the type of interrupt that will be recognized.

"SND $\left\{ \begin{matrix} A \\ B \\ C \\ D \end{matrix} \right\}$ "

This type of WHEN command transfers program control to the specified line number when a sequence of SEND-waveform-START buttons are pressed on the DPO front panel. More than one waveform button can be pressed and recognized if the appropriate WHEN's are set.

DPO DRIVER PACKAGE

"CB1" through "CB15"

These are the DPO PROGRAM CALL buttons. "CB14" can be used for terminating a zero-reference acquisition as well as generating an interrupt.

"REC" $\left\{ \begin{array}{c} A \\ B \\ C \\ D \end{array} \right\}$ "

This works the same way as SEND except that the RECEIVE button is recognized instead of SEND.

"SSC"

This type of WHEN command transfers program control when the single-sweep-complete interrupt occurs. It is only recognized if the single sweep was armed with a PUT command. (Manual arming is not recognized.)

An IGNORE command of the following form can be used to disable any of the preceeding interrupts.

IGNORE #ilun [,string expression]

The string expression must be one of the preceeding list of keywords. If none is specified, all interrupts for the DPO are ignored. Likewise an END, STOP, or Control-P instruction can be used to disable all interrupts.

One of the more important uses of the WHEN command is recognizing interrupts from the DPO PROGRAM CALL buttons. These buttons were so named because they can be programmed to begin program execution at a particular line number. This is particularly useful where you have a series of short routines which must frequently be executed. By pressing the appropriate button you can immediately select the next routine to be executed.

DPO DRIVER PACKAGE

As an example of programming the PROGRAM CALL buttons, consider the following routine. This assumes that DPO #1 is being addressed.

```
100 WHEN #1 HAS "CB1" GOSUB 200
110 WHEN #1 HAS "CB2" GOSUB 210
120 WHEN #1 HAS "CB3" GOSUB 220
130 WHEN #1 HAS "CB15" AT 126 GOSUB 230
150 GOTO 150
200 PRINT "BUTTON - 1"\RETURN
210 PRINT "BUTTON - 2"\RETURN
220 PRINT "BUTTON - 3"\RETURN
230 PRINT "OPERATION HALTED"\RETURN
```

The program begins by setting up interrupt routines for PROGRAM CALL buttons 1 through 3. These interrupt routines have equal priority. (Since no priority was specified in the first three WHEN commands, a priority of 51 is used by default.) The associated interrupt routines are found in lines 200 through 220 which simply print which buttons were pressed. Line 130 sets up an interrupt routine for PROGRAM CALL button 15, which in this case functions as a halt button. Since it has a priority of 126 (the highest possible) it can be used to interrupt any other interrupts of lesser priority. In each case, after the interrupt routine has been completed, the program loops at 150 waiting for a further interrupt. To stop this program, press Control-P. However, Control-P disables all interrupts so that the program must be RUN again before the PROGRAM CALL interrupts can be recognized.

Another useful function of the WHEN command is in detecting interrupts from the SEND and RECEIVE buttons on the DPO. (The SEND and RECEIVE buttons do not automatically transfer waveforms in TEK SPS BASIC.) For example, the following routine defines four waveforms in controller memory. Then when a sequence of SEND, Memory Location:A, B, C, or D, and START is pressed, data is transferred from the selected memory location to the corresponding waveform array in controller memory. Also the waveform is graphed on the terminal screen. Finally, the routine enters a busy loop waiting for another interrupt to occur. This routine accesses the DPO ATTACHED as #5.

DPO DRIVER PACKAGE

```
110 WAVEFORM WA IS A(511),SA,HA$,VA$
120 WAVEFORM WB IS B(511),SB,HB$,VB$
130 WAVEFORM WC IS C(511),SC,HC$,VC$
140 WAVEFORM WD IS D(511),SD,HD$,VD$
150 WHEN #5 HAS "SNDA" GOSUB 250
160 WHEN #5 HAS "SNDB" GOSUB 260
170 WHEN #5 HAS "SNDC" GOSUB 270
180 WHEN #5 HAS "SNDD" GOSUB 280
200 GOTO 200
250 GET WA FROM #5,"A"
255 PAGE\GRAPH WA\RETURN
260 GET WB FROM #5,"B"
265 PAGE\GRAPH WB\RETURN
270 GET WC FROM #5,"C"
275 PAGE\GRAPH WC\RETURN
280 GET WD FROM #5,"D"
285 PAGE\GRAPH WD\RETURN
```

In order to interrupt this routine, Control-P must be entered on the terminal. However, since Control-P disables all WHEN commands, the program must be run again before it can be used.

A similar routine could be written which would detect DPO interrupts from a sequence of RECEIVE, Memory Location:A, B, C, or D, and START. This could be used to transfer data from controller arrays to designated memory locations in the DPO.

Display and Acquisition of ASCII Text

TEK SPS BASIC has two commands which deal with readout information displayed on the DPO CRT. The PUT command is used in displaying text on the DPO CRT and in labeling a particular DPO memory location. Conversely, the GET command is used to acquire ASCII readout information from a particular memory location. Before examining each of these operations we need to take a brief look at which characters can be stored and where they are stored in P7001 memory.

DPO DRIVER PACKAGE

The memory of the P7001 Processor consists of two major sections. The first section of memory is dedicated to storing waveforms for display on the DPO CRT by the readout interface. This latter section of memory is divided into 16 areas (four fields per waveform location) six of which are available to the user through TEK SPS BASIC.

Four of the available areas, field zero of waveforms A through D, are used for storing scale-factor messages that occur during waveform storage and transfer operations. Also, labeled messages generated by the PUT command are contained within field zero of the waveform location addressed by the PUT command. Field one of waveform A and field one of waveform B are the two remaining areas. Field one of A is used to store displayed messages generated by the PUT command. Field one of B is used to store zero-reference requests generated by the controller during waveform storage. Fig. 3-2 shows a functional diagram of the 4K P7001 memory. (Refer to the DPO Operators manual for more information, such as memory locations in 1K, 2K, and 3K memories.)

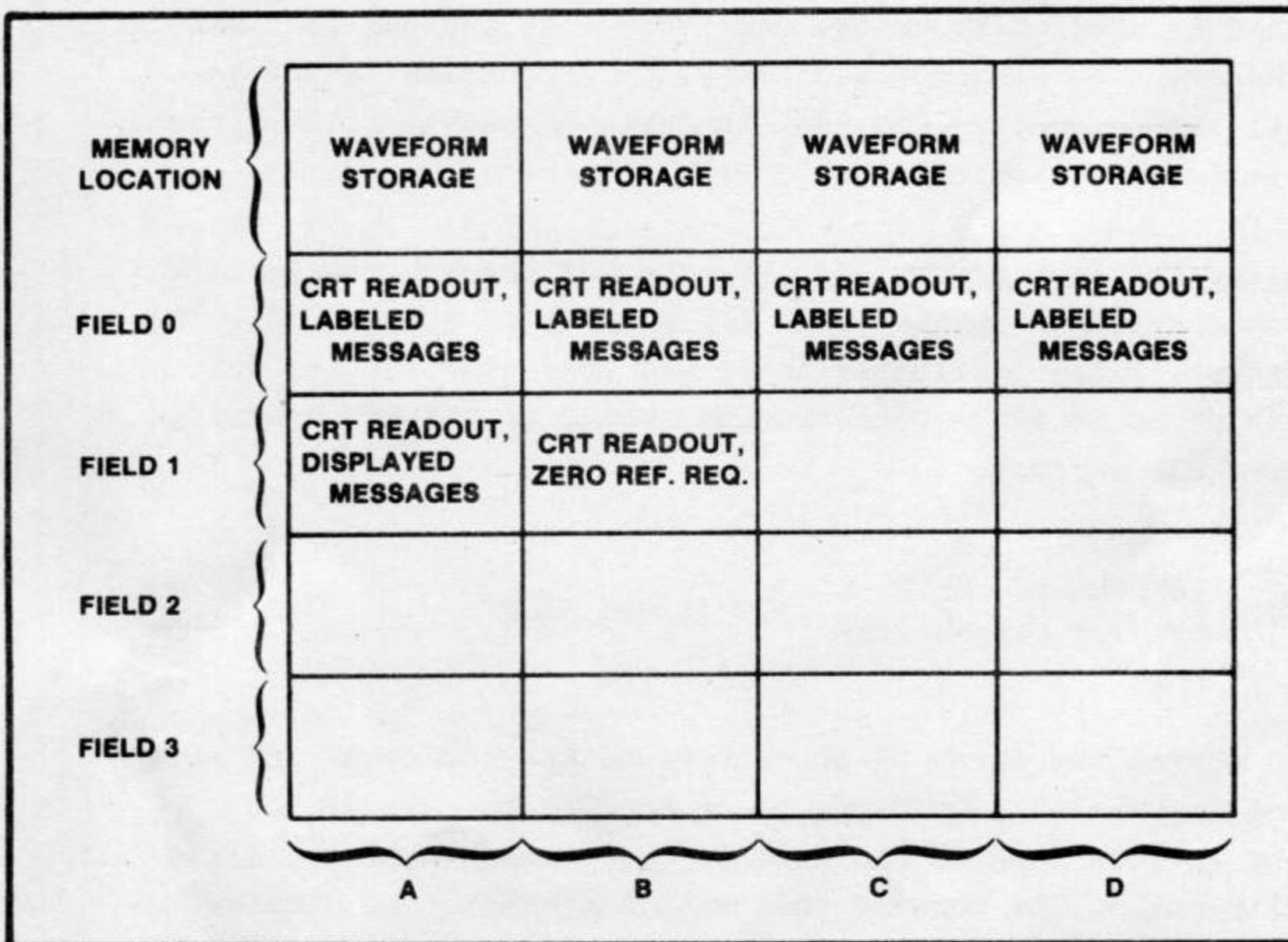


Fig. 3-2. Functional diagram of 4K P7001 memory.

DPO DRIVER PACKAGE

Displayable Characters. The set of characters displayable on the DPO CRT readout is a subset of the full 96 character ASCII code set. The characters available are: the digits 0 through 9; all upper-case letters; the lower-case letters c, d, m, n, p, u; and the six characters ., <, /, +, -, and >. There are three additional non-ASCII characters available by outputting the ASCII characters shown in Table 3-2.

TABLE 3-2

ASCII Character	Readout Character
!	†
=	Δ
@	Ω

Since a standard computer terminal has only upper case letters, a special method is used to input the six lower case letters from the keyboard. Whenever the CTRL key and R key are typed simultaneously (^R), it will not be echoed. Instead, the next character to be typed will be converted to lower case before it is stored.

Displaying Messages on the DPO CRT. The PUT command may be used to send various messages to the P7001 for display on the DPO CRT. The messages consist of the previously described character set and may have a length of up to 80 characters. The format of the PUT command, when used for this purpose, is:

```
PUT {string variable } INTO #ilun, "DIS"  
    {string expression}
```

This command stores the first 80 characters of the source string variable or string expression into field 1A of the DPO memory. If the source string is less than 80 characters, the remainder of the field is filled with nulls. The command then sets the readout to display field 1A and sets the front panel display source to MEMORY.

DPO DRIVER PACKAGE

Displayed messages appear beginning at the upper-left corner and ending at the lower-right corner of the DPO CRT; they occupy the space normally used for scale factors and zero-reference levels. However, a displayed message does not modify the stored readout (such as scale factors and units) associated with any of the four DPO memory locations. It merely displays the message in the portion of screen used by the readout. In fact, pressing any MEMORY LOCATION button on the DPO will extinguish the displayed message and redisplay the contents of the selected memory location.

As an example, the following command will display the message "LASER EXPERIMENTS" in the upper-left corner of the CRT if the DPO Display Source is set to BOTH or MEMORY:

```
PUT "LASER EXPERIMENTS" INTO #1,"DIS"
```

This assumes that the DPO driver is loaded into memory and that a DPO has been ATTACHED as Instrument Logical Unit Number 1. The message will remain on the CRT until the displayed memory location is changed either manually or under program control. Changing memory locations permanently extinguishes the displayed message; thus to redisplay the message, the PUT command must be reexecuted.

Labeling DPO Memory Locations. In some cases, you may want to attach a message to a P7001 waveform that will remain with the waveform for as long as it is stored in a particular memory location. By using a labeled message, up to 34 characters of information may be written into one of the four memory areas reserved for scale factors and readout. The following format of PUT is used in labeling a memory location:

```
PUT {string variable } INTO #1un, LAB { A }  
    {string expression}                                     { B }  
                                                         { C }  
                                                         { D }
```

DPO DRIVER PACKAGE

This command stores the first 34 characters of the string variable or string expression into one of these selected areas of memory: field 0A (for memory location A), field 0B (for memory location B), field 0C (for memory location C), and field 0D (for memory location D). If the source string is less than 34 characters, the remainder of the locations are filled with nulls. The message is displayed beginning at the lower left corner of the DPO CRT and is seen, along with the scale factors and units, whenever the associated memory location is selected for display.

As an example, the following commands will display "EXPERIMENT 1" in the lower-left corner of the CRT if Memory Location A is selected:

```
A$="EXPERIMENT 1"\PUT A$ INTO #1,"LABA"
```

Again it is assumed that the DPO driver is loaded and that a DPO is ATTACHED as Instrument Logical Unit Number 1. The labeled message appears whenever location A is selected and is not deleted by switching memory locations.

Acquiring DPO Readout Information. Occasionally a need may arise to input DPO readout information directly from the P7001 memory to controller memory. A special format of the GET command allows this. Before discussing this format of GET, however, let's take a brief look at the conventions used in numbering the CRT readout channels.

As seen in Fig. 3-3, there are eight readout channels (each with 10 characters) on the DPO CRT. These are numbered 0 through 7 and each one occupies a unique position on the CRT. For example, when a dual-trace vertical plug-in is inserted in the extreme left-hand compartment, the plug-in's CHANNEL 1 readout will appear in channel 0 on the CRT and the plug-in's CHANNEL 2 readout will appear in channel 4 on the CRT. On the other hand, when a horizontal (time base) plug-in is inserted in the extreme right-hand compartment, the sweep speed settings will appear in channel 3 on the CRT. (Nothing will appear in channel 7 on the CRT; this is reserved for other uses.)

DPO DRIVER PACKAGE

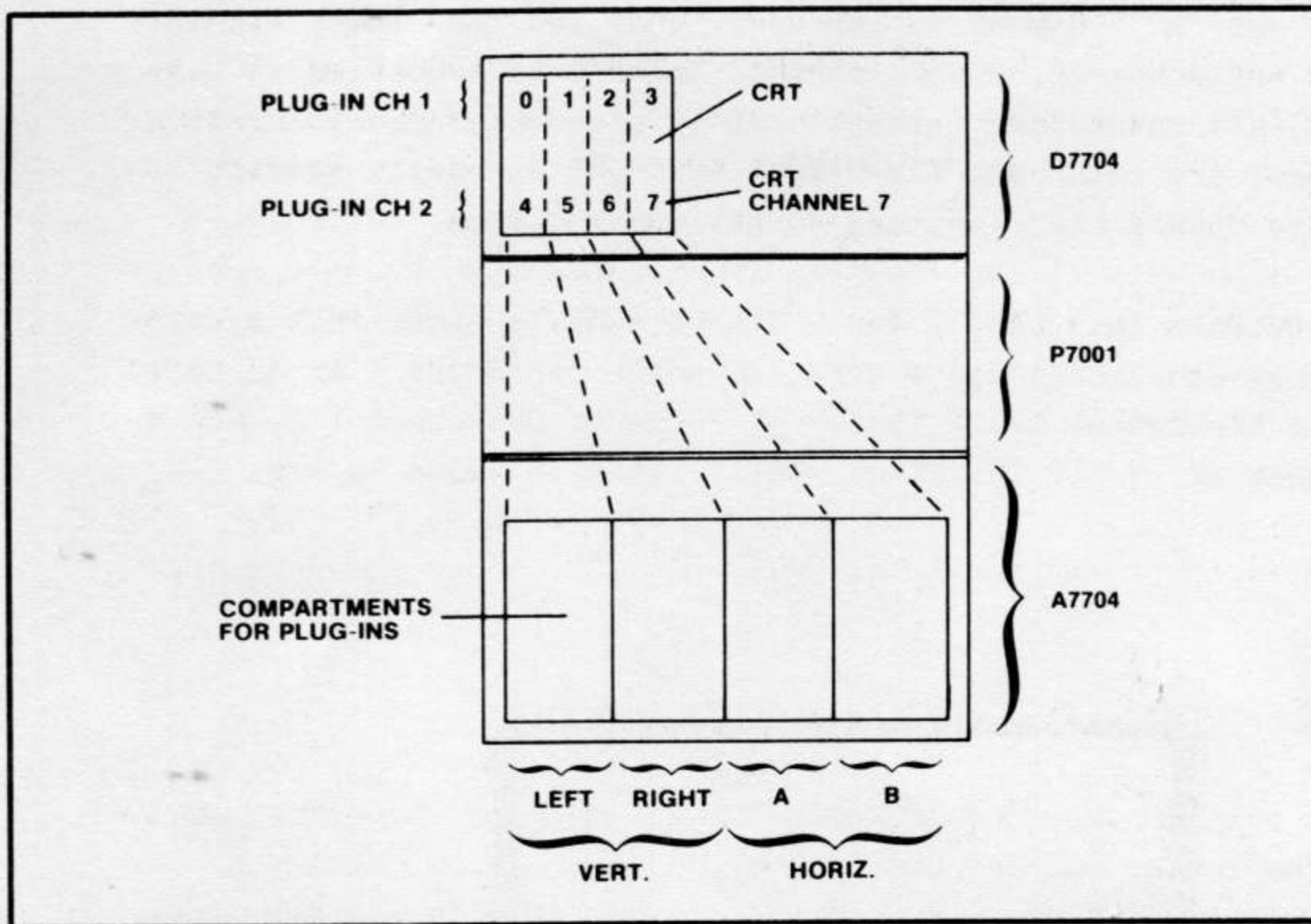


Fig. 3-3. DPO CRT readout conventions.

Each ASCII readout channel of the four DPO waveforms can be read using the GET command as follows:

GET {variable
string variable} FROM #ilun, "CH {A
B
C
D} number [,number][L]"

In the above format, the "number" argument must be an integer between 0 and 7. It specifies which CRT readout channel is to be read from. If two numbers are specified, the contents of the corresponding readout channels are concatenated. This allows long strings of readout information from digital plug-ins to be processed. When a variable is specified as the destination, the numeric portion of the readout is returned.

DPO DRIVER PACKAGE

However, if the "L" argument is included, only the four least significant digits are acquired. When a string variable is specified as the destination, all characters (except spaces) after the numeric portion of the channel are returned. The "ilun" and "CH" arguments specify which DPO and Memory Locations are to be acquired from.

To illustrate this use of the GET statement, suppose that a waveform in Memory Location A has a vertical scale factor of 5 mV in CRT-channel 0, a horizontal scale factor of 20 ms in CRT-channel 2, and a zero reference of -3 DIV in CRT-channel 7. This is shown in Fig. 3-4.

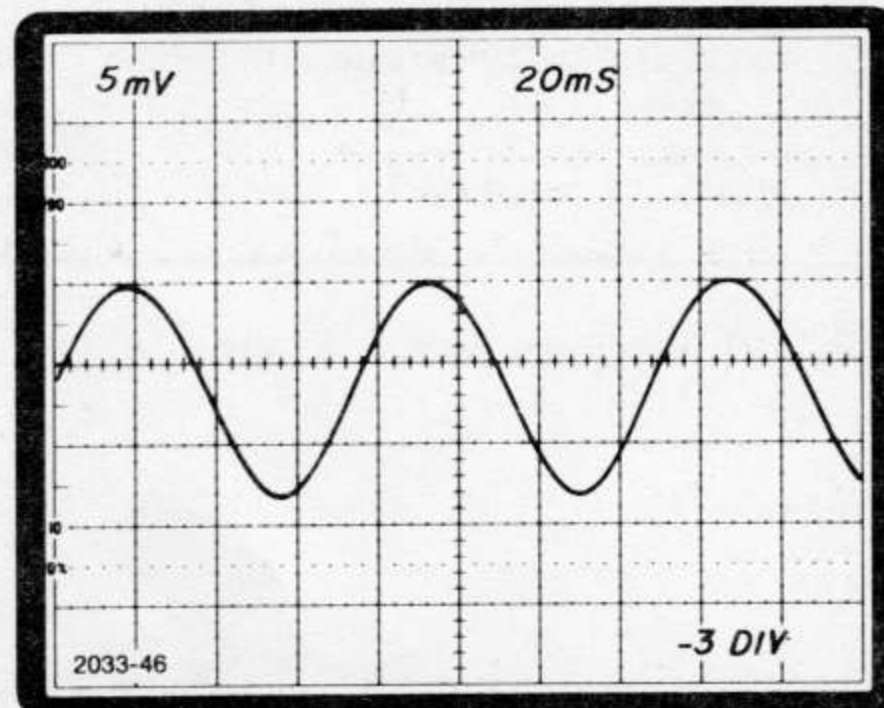


Fig. 3-4. Example of CRT readout on DPO.

Under this condition, it is possible to GET the scale-factor information into the controller memory and PRINT it on the terminal. Each of the following three examples illustrate this by showing the statements and the resulting output.

DPO DRIVER PACKAGE

- 1) GET X FROM #1,"CHA0"\GET A\$ FROM #1,"CHA0"\PRINT X,A\$
5,00000E-03 V
- 2) GET Y FROM #1,"CHA2"\GET B FROM #1,"CHA2"\PRINT Y,B\$
.02 S
- 3) GET Z FROM #1,"CHA7"\GET C\$ FROM #1,"CHA7"\PRINT Z,C\$
-3 DIV

The first example GETs the scale-factor information in CRT channel 0 of the waveform in Memory Location A. It then stores the floating-point value in the variable X and stores the units (V for "volts") in A\$. Note that the m (for micro) prefix is inherent in the floating-point value and therefore is not entered in A\$. The second example does the same as the first, except that CRT channel 2 is selected and the floating-point value is stored in Y. This time the units stored in B\$ are s (for seconds). The third example GETs the baseline information stored in CRT channel 7 and stores it in Z. Also, the units (DIV for divisions) are stored in C\$.

Consider another example where the readout from a TEKTRONIX 7D13 Digital Multimeter is stored in CRT channel 1 of Memory Location A. If the readout of the multimeter is:

10.15 Ω

then the GET statements:

GET X FROM #1,"CHA1"\GET A\$ FROM #1,"CHA1"

would store the readout from it. When the statement:

PRINT X,A\$

is entered, the following output may be seen:

DPO DRIVER PACKAGE

.015E + 4 @

Notice that the Ω symbol was changed to @ as noted in Table 3-2.

YOUR COMMENTS COUNT

The Manual Writers at Tektronix, Inc. are interested in what you think about this manual, how you use it, and changes you might like to see in future manuals. Any queries regarding this manual will be answered personally.

What did you find that was:

interesting? _____

frustrating? _____

helpful? _____

confusing? _____

Is there anything you would like to see added to or deleted from this manual? _____

What is your major application area for this product? _____

Have you found any interesting applications, operating hints, or software routines which you would like to share with us? _____

* * * * *

Name: _____ Position: _____

Company: _____ Department: _____

Street: _____

City: _____ State: _____ Zip: _____

BUSINESS REPLY MAIL

No postage necessary if mailed in the United States

Postage will be paid by

TEKTRONIX, INC.

P.O. Box 500

Beaverton, Oregon 97005

FIRST CLASS

PERMIT NO. 61

BEAVERTON, OREGON

ATTN: SPS Manuals Group 157

