

DN 6000342

**User Manual Real Time Display Setup
Command Language For Windows**

**Acroamatics, Inc.
70 South Kellogg Avenue
Goleta, CA 93117-3476**

July 17, 2006

PROPRIETARY NOTICE

Information contained in this document is disclosed in confidence and may not be duplicated in full or in part by any person without prior written approval by Acroamatics, Inc., except as provided by separate contractual agreement. Its sole purpose is to provide the user with adequately detailed documentation in order to install, operate, maintain, and order spare parts for the equipment supplied. The use of this document for any other purpose is expressly prohibited.

ACROAMATICS DOCUMENT HISTORY

The following table indicates major changes made to User Manual Real Time Display Setup Command Language For Windows, Acroamatics Document Number DN 6000342, released on April 17, 2006 and contains a record of all revisions made since that date.

DN 6000342 CHANGE HISTORY			
Rev	Date	Action	Name
	01-01-00	Original Issue	TR
A	11-05-01	Added data source descriptions.	TR
B	10-28-03	Added new display data type (unsigned short).	TR
C	12-10-04	Added broadcast function on numeric pages	TR
D	09-15-06	Convert Source To Word Format	TR
E	11-27-07	Added embedded time. Also added time of day output format for numeric display pages.	TR
F	12-19-08	Added a description of the numeric display page update rate commands.	TR

TABLE OF CONTENTS

Chapter 1	3
INTRODUCTION	3
1.1 PROGRAMMING LANGUAGE STRUCTURE	3
Chapter 2	5
DISPLAY OPERATION - SETUP MODE.....	5
2.1 GENERAL	5
2.2 NUMERIC DISPLAY PAGESETUP	6
2.3 NUMERIC DISPLAY SLOTSETUP.....	6
2.3.1 Data Source and ID selection.....	7
2.3.2 Data Conversion Algorithm.....	9
2.3.3 Output Format Selection.....	10
2.3.4 Converting Data To Engineering Units	10
2.3.5 Setting Slot Alarms	11
2.3.6 Processing to be Applied to the Data (Expression Command)	11
2.3.7 Measurement Annotation.....	15
2.3.8 Broadcasting Data.....	15
2.3.9 Out Of Limits Display Page Setup.....	17
Chapter 3	18
BAR CHART DISPLAY.....	18
3.1 BARCHART DISPLAY SETUP	18
3.2 BARCHART DISPLAY SLOTSETUP	18
3.2.1 Data Source and ID selection.....	18
3.2.2 Data Conversion Algorithm.....	19
3.2.3 Maximum and Minimum Values	19
3.2.4 Origin And Limit	20
3.2.5 Measurement Annotation.....	20
3.2.6 Arithmetic Processing.....	20
Chapter 4	23
CROSS PLOT DISPLAY.....	23
4.1 CROSS PLOT DISPLAY SETUP	23
4.2 CROSS PLOT DISPLAY SLOTSETUP	23
4.2.1 Vector Plots And Point Plots	24
4.2.2 Axis Selection	24
4.2.3 Data Source and ID selection.....	24
4.2.4 Time Axis Function	25
4.2.5 Data Conversion Algorithm.....	25
4.2.6 Maximum And Minimum Values	25
4.2.7 Origin and Limit	26
4.2.8 Measurement Labels	26
4.2.9 Axis Annotation (Numeric)	26
4.2.10 Arithmetic Processing.....	27
Chapter 5	29
EMBEDDED TIME	29
5.1 DESCRIPTION OF EMBEDDED TIME	29
5.2 EMBEDDED TIME COMMANDS.....	30
5.2.1 Time Command	30
5.2.2 Length Command	31
5.2.3 Data Source Command	31
5.2.4 Data Justification Command.....	31
5.2.5 High Order Time Word Command	31
5.2.6 Low Order Time Word Command.....	32
5.2.7 Microsecond Time Word Command	32
5.2.8 End Command	32

Chapter 633
SETUP AND LAYOUT FILES33
 6.1 SETUP FILES33
 6.2 LAYOUT FILES.....33
Chapter 735
PROGRAMMING EXAMPLES35
 7.1 INTRODUCTION.....35

USERS MANUAL

REAL TIME DISPLAYSETUP COMMAND LANGUAGE

FOR WINDOWS

CHAPTER 1 INTRODUCTION

The Real Time Display Program provides a way for you to view data from the Acroamatics TDP data stream. You can see the processed or unprocessed data as cross plots, bar charts, or in numerical form. The program uses setup files to control the data display. These setup files can be generated by the Acroamatics TDP compiler, or they can be produced and modified in the Real Time Display Program by using its setup menu system.

This manual describes the TDP Compiler Real Time Display command language. There is on line help in the Real Time Display program (display.exe) which describes how to use the display program and its setup menu. Command line arguments can be given when starting up display.exe that allow you to select a display setup file, and to select any page for initial display. The command line arguments can be given alone or in any order. The page number command must be a number between 1 and 16 concatenated with the hyphen (-) or slash (/) prefix. Any other command line argument will be interpreted as the name of the display setup file. If no command line arguments are used the program will look for the setup file named display.rtd and initiate the display on page 1 by default.

Chapter two of this manual describes setup for the Numeric and Out of Limits Display, but contains command descriptions that are common to the other display formats as well. Detailed descriptions of the common setup commands are contained in this first setup section. Chapters three and four of this manual discuss Display Setup commands for barchart and crossplots.

1.1 PROGRAMMING LANGUAGE STRUCTURE

The command language is a stream of text which is used to define the setup of the Realtime Display processor.

A Realtime Display setup program is made up of a series of lines of text containing program statements and comments. Comments begin with a vertical bar symbol (|); all characters on a line following a vertical bar are treated as commentary and ignored by the Compiler. Program statements are strings of alphanumeric text, and are constructed according to rules described in this manual. Certain conventions are used to display the lines of text when program statements are described in the chapters that follow, and statements are described in two different ways; (1) Schematic representations of a statement define the rules for construction of a real statement, and (2) Example statements show arbitrary typical statements constructed from these rules.

The following conventions apply to schematic statements:

The upper case strings in any schematic representation are keywords and must appear literally as shown. Keywords are reserved strings which define the action taken by the Compiler.

The punctuation characters: () { } are literals which must appear as shown in the examples.

- The lower case strings
- n A digit
- nn A number represent variables which must conform to the following formation rules: Numbers are strings of numeric characters. A number preceded by a "/" is interpreted as hexadecimal. In the text line schematics, "n" means a single digit and "nn" means any number less than 32768. For some purposes a number may be a real number

expressed as $\pm i.f$, where I represents the integer part and f the fractional part.

- All other lower case strings in a schematic represent parameters for which there is more than one choice. These lower case strings describe the class of the parameter.

CHAPTER 2

DISPLAY OPERATION - SETUP MODE

2.1 GENERAL

The Acroamatics Telemetry Data Processor Compiler (tdpc.exe) provides setup for the Real Time Display Program (display.exe) by translating Real Time Display commands into a binary setup file. The default name for the setup file is "DISPLAY.RTD". The default setup file name can be changed by using the environment variable named "DSPMEM" in the TDP configuration file named c:\tdpsys\tdpc.cfg. (Complete instructions for setting the environment variables in this file can be found in Acroamatics document DN 6000363 - TDP SOFTWARE CONFIGURATION PARAMETERS). Also the binary display setup file can be named dynamically using display setup commands. This method is described below.

The Display is organized into units called pages. There are sixteen pages of display in the Display System, any of which may be programmed to display the data in one of the following formats:

1. Numeric Display - Up to forty measurements may be displayed in the form of binary, octal, decimal, hexadecimal, or in engineering units. The form of the display may be selected individually for each measurement.
2. Bar Chart - The bar chart display may be used to display up to eight measurements simultaneously.
3. Cross Plots - Up to four plots of one measurement versus another will be displayed in the form of a X-Y plot with annotated axes.
4. Out of Limits - Up to forty measurements may be monitored. When any of these measurements exceeds the programmed alarm value, the data may be displayed in the form of binary, octal, decimal, hexadecimal, or in engineering units.

A typical Display setup specification has the schematic:

```
DSP [filename.rtd]
PAGE=nn
.
Page Setup
.
Other Pages
.
.
END
```

The optional parameter on the DSP command line can be used to dynamically name the setup file.

The display setup can be given interactively from the system keyboard or in batch mode from a TDP Compiler input file. Begin Real Time Display setup by typing the command:

```
DSP
```

on the Video Display Terminal (VDT) or in a TDP Compiler input file. After entering the Display System Setup Mode, any of the sixteen pages of the Real Time Display may be initialized or modified. To begin page setup while in Setup Mode, use the PAGE command. The syntax for this command is:

```
PAGE=nn [type]
```

where nn is the desired page number (between one and sixteen). The optional parameter type is used to select the format of the real time data display. The different formats available are:

FORMAT	COMMAND
Numeric Display	ND
Bar Chart	BC
Cross Plots	CP
Out of Limits	OL

To select one of these display formats, the parameter type must be ND, BC, CP, or OL. For example, to set up page five for the Numeric Display format, enter:

```
PAGE=5 ND
```

If upon completion of the setup for this page, it is determined that additional setup is required, subsequent entries to the page can be made with:

```
PAGE=5
```

since no new type was specified, the Numeric Display format will remain in effect. The parameter type is used when the page format has not been previously set, or if the page format must be changed. If type has already been set, it is not necessary to re-specify it when additional setup is entered for that page. If type is not specified, and no previous display setup has been done, the pro-gram defaults to Numeric Display. To exit the Display setup mode use the command:

```
END
```

2.2 NUMERIC DISPLAY PAGESETUP

To set up a NUMERIC DISPLAY page enter the command:

```
PAGE=nn [ND]
```

where nn is the page number (1 through 16 are valid). The ND is unnecessary if the display format has not been previously setup as Numeric Display is the default display format. If the page numbered was programmed as a different nn display format than Numeric Display, this page declaration will cause the page Setup Memory to be reset. Any time the PAGE command includes the optional parameter type, and the parameter indicates a display format other than that currently programmed for that page, the Display Setup Memory is reset for the page. This disables all previous page setup, and establishes the new display for-mat for the page.

Once the page setup mode of the Display Setup Program is entered you may optionally give the page an alphanumeric name and/or set the display update rate. Or you may immediately start programming the individual data slots with slot setup commands. (A slot corresponds to one of the forty measurements that may be setup on a Numeric Display Page.)

You select the rate at which the display program will update the numeric display of data by using one of the update rate commands. You can select update rates of 10 milliseconds, ½ second, 1 second, or 2 seconds by using the 10MS, HS, 1S, or 2S commands. You must give this command on a separate command line. This rate will apply to all slots on the numeric display page. If you do not use this command, by default the real time display program will use an update rate of ½ second.

To give a page an alphanumeric name by which the page may be referenced, you must use the NAME command. The syntax is:

```
NAME pagename
```

where pagename is any alphanumeric string of ten characters or less. This page name will be displayed when the display program runs and is useful for completing the data annotation. It should be noted that the PAGE and NAME commands can be truncated to PAG and NAM. This is legal because the TDP Compiler recognizes commands by their first three characters only. Any Compiler processed command may be invoked with the first three or more of its characters.

2.3 NUMERIC DISPLAY SLOTSETUP

To select a measurement and describe its format for display, the SLOT command must be given. The slot number must be given with the SLOT command as follows:

SLOT=nn

where nn may be a number from one to forty. In the numeric display, the measurements (slots) are displayed in the program window, two slots to a line on line numbers for twenty lines. For example, slot one measurements are displayed in the left column of the first display line, slot two, in the right column of the same line. Slot forty measurements are displayed in the right column of the twentieth line.

If a slot has been previously setup, the SCL command (Slot CLear) will reset the display parameters currently set for the Slot. This command is useful in situations where an operator wishes to use a predefined display page for the display type for which it is currently programmed, but wishes to reprogram one or more of the slots currently set up.

After the Slot command, further commands may be given allowing specification of the measurement and the format of its display. These commands permit the programmer to:

1. Select the data source and numeric ID of the measurement.
2. Select a data conversion algorithm to apply to the data.
3. Select the format for the data display.
4. Convert data to engineering units.
5. Set limits on numeric display parameters so that any occurrence of data outside those limits will cause the data to be displayed in a red font.
6. Select arithmetic processes to be applied to the data.
7. Select a ten character name to identify the measurement on the display.
8. Select up to ten characters of alphanumeric units annotation to be displayed with the measurement data.
9. Select the update rate for data display
10. Broadcast the data on the page to a multicast socket or a comport.

When the slot setup has been fully specified the programmer may proceed to set up another slot on the same page by issuing another SLOT command. Or the programmer may choose to begin setup on another page by giving a PAGE command. Another option upon completing a slot setup specification is to exit Real Time Display Setup Mode by giving the END command. Following is an explanation of how to perform the slot setup operations listed above.

2.3.1 Data Source and ID selection

The following is a summary of the data source selection commands. To select the measurement data source and ID you must use the data source selection command:

COMMAND	FUNCTION
\$FI nn	Input FIFO Bus on models 504 or 4504 DIST
\$OP nn	Output Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DA nn	DAC Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DVn nn	CVT partition on models 504 or 4504 DIST, or 1605 PDSP
\$SP nn	Special offset on FIFO bus
\$PCMn nn	Model 1502 or 1602 FSYN Current Value Table
\$FB nn	Feedback port on model 1605 PDSP

The single n appended to \$PCM selects a particular model 1502 or 1602 FSYN card in your system and can be in the range of 1 through 8. It is impossible to have both models 1502 and

1602 FSYN in the same system, so these tokens can be shared between them. Likewise the single n appended to the \$DV selects a different device on the models 504 or 4504 DIST card or the model 1605 PDSP card. You cannot have more than one of any of these three cards in your system, so some tokens are shared between the three.

The nn in the command is the numeric ID. For example, to select FIFO ID number one hundred, give the command:

```
$FI 100
```

If only the ID number is given, the program uses the currently selected default data source. The default data source is initialized as the input FIFO bus. This selection may be changed at any time while operating in Display Setup Mode. To change the Default Source selection, select the data source with no ID specified. For example:

```
$DA
```

changes the default data source to the DAC bus. This selection remains in effect until it is changed with another default data source selection command. Data source selection commands given with ID parameters do not change the default data source selection. A subsequent ID given as:

```
/c0
```

selects as the slot measurement the data defined by DAC ID (hexadecimal) c0.

An environmental variable can be used to mark an arbitrary base offset on the FIFO bus for symbolic reference in the Real-Time Display setup program, and in the Real-Time Display setup menu. To use this feature, set the environmental variable in the autoexec.bat file, the tdpc.cfg configuration file or at the DOS window prompt with the command:

```
set SPECIAL_OFFSET=nn
```

where nn is the numeric representation of the desired offset. The number can be given in standard decimal notation. If hexadecimal notation is preferred, it must be prepended with the slash character. For example, /200 is equivalent to 512. Refer to this offset symbolically in Real-Time Display setup, or in the Real-Time Display Menu by using the \$\$SP token. If the SPECIAL_OFFSET variable is not defined, use of the \$\$SP token yields a zero base offset on the FIFO bus.

This feature can be useful for accessing a block of IDs that have been assigned to a common data source. For example a large block of video data could be assigned sequential IDs starting at FIFO offset 8192. Using the SPECIAL_OFFSET mechanism these IDs could be referenced starting with \$\$SP 0.

The ID parameter for the Data Selection command may also be given as the symbolic ID which was used to define the data word in the de-commutation program. The TDP compiler program (tdpc.exe) maintains a symbol table that saves the association between the TM data IDs and their symbolic names. The command to save the symbols is SSF [file]. The command to restore the file to the Compiler is RSF [file]. The name of the symbol table file defaults to "symbols". You can use a different file name by using a filename token with the SSF or RSF command, or by setting the configuration/environment variable called SYMBOLS.

The Real Time Display Program (display.exe) can use a symbol table file for data source selection in it's setup menu. You can read that program's on-line help about the details of this use. The symbol table and symbol table file features are documented Acroamatics Documents 6000265 and 6000343.

Another option for data source for a numeric display slot is "time of day" data. You can select this option by using the TIME command, instead of any of the data source options discussed above. The default output format for time of day shows days, hours, minutes and seconds as:

```
325: 10: 18: 23
```

If you want to also see fractions of seconds in the data display, use the FL1, FL2, or FL3 output format selection commands (see section 2.3.3) to see time resolved to tenths, hundredths, or thousandths of seconds. Any other format selection command combined with the choice of TIME as your data source will produce a display resolved to whole seconds.

2.3.2 Data Conversion Algorithm

Data is converted so that the arithmetic processing can be performed on a uniform, twos complement data type. When data is digitized the digital result may represent the data in several different ways, depending on the type of converter used and the range of the data. unipolar converters, commonly called “magnitude” converters, represent the most negative data as 0 and the most positive data as 11111.... Bipolar twos complement converters represent the most negative data as 1000000..., the zero value as 0 and the most positive value as 011111... (Bipolar ones complement converters differ from twos complement by one count on the negative side only.) Bipolar sign and magnitude represent the most negative value as 11111, the zero value a 0, and the most positive value as 011111.

Frequently, unipolar converters are used to measure bipolar signals and bipolar converters are used to measure unipolar signals. This is accomplished by off-setting the analog signal so that it lies in the optimum conversion range. When this data is processed, it is generally desirable to convert it to properly ranged twos complement data. Twos complement representation is used for computations by the Real Time Display Processor which processes the data output by the TDP.

The available data conversions are:

COMMAND	DESCRIPTION
2CM	Twos Complement
1CM	Ones Complement
OBN	Offset Binary
SMG	Sign and Magnitude
UNS	Unsigned Short
LONG	32 Bit Long Word
FLOAT	32 Bit Floating Point

- a. Twos Complement
Twos complement data is passed to the arithmetic processors unchanged. The twos complement mode should also be used to leave non-analog data unmodified. This is a 16 bit data type.
- b. Ones Complement
Ones complement data is converted by incrementing negative values by one count. This is a 16 bit data type.
- c. Offset Binary
Offset binary data is generated when unipolar measuring devices measure bipolar signals. The conversion to twos complement is accomplished by inverting the sign bit, which effectively removes the half of full scale offset which was applied to the data when it was measured. Offset binary conversion is also appropriate when a bipolar converter measures a unipolar signal. The result of this is a magnitude datum, which requires special treatment, discussed below. This is a 16 bit data type.
- d. Sign and Magnitude
Sign and Magnitude data is converted by twos complementing all except the sign bit if the sign bit is a one. This is a 16 bit data type.
- e. Unsigned Short

Unsigned short data is 16 bit unsigned data. The default maximum is 65536, and the default minimum is 0.

f. 32 bit long word

Use this conversion for 32 bit twos complement data. The default maximum is 2,147,483,647.5 and the default minimum is -2,147,483,647.5 for this data conversion.

g. 32 bit floating point

Use this conversion when the data is in 32 bit IEEE floating point format. The data scaling commands do not apply to this data conversion type.

For data that is not in twos complement form (it is ones complement, offset binary, or sign magnitude), conversion to twos complement is achieved with either the "1CM" (apply to ones complement data), "OBN" (apply to offset binary data), or SMG (apply to sign magnitude data) commands. Handling 16 bit unsigned data is done with the UNS command. At initialization, the data conversion algorithm default selection is twos complement. If the data is known to be in 16 bit twos complement form, no conversion algorithm selection need be made, although the 2CM command is available. If for some reason it becomes necessary to change the data conversion selection for a slot to 16 bit twos complement from some other earlier selection, the 2CM command is required.

2.3.3 Output Format Selection

The slot measurement may be displayed numerically as a binary, octal, decimal, hexadecimal, degree, or floating point number. To select one of these numeric formats, an Output Format Selection command must be given. This command must be one of:

```
BIN OCT DEC HEX
FL0 FL1 FL2 FL3
FL4 FL5 FL6 DEG SCI
```

To choose binary format use the BIN command. The OCT, DEC, and HEX commands should be used to select octal, decimal or hexadecimal format respectively. When choosing the floating point display format you may select the number of digits displayed to the right of the decimal point. The command FL6 selects floating point format, with six digits displayed to the right of the decimal point. Other choices are FL0, FL1, FL2, FL3, FL4, and FL5. DEG programs the computer to display the data measured in radians as degrees, minutes, seconds format (DD.MM'SS"). SCI displays the data in scientific notation format. For Numeric display slots that are programmed as simple IDs (as opposed to expressions) any of the above selections may be chosen to display the data. If no selection is made the system defaults to HEX. When a slot is programmed as an expression (See Section 2.3.6) the output must be floating point format or hexadecimal. When a slot is programmed as an expression and no selection, or a non-floating point selection is made, the hexadecimal format will be selected at display time by the Display Processor.

In addition to these numeric formats there is the message format. With this format messages that reflect information about the data can be displayed instead of the actual data. One of two different messages will be displayed depending on the data value and the alarm values you have chosen. These messages can be up to 32 characters in length. The schematic for this commands is:

```
MSG=Normal Data Message; Alarm Data Message
```

As you can see the parameters for the message command are two semi-colon separated strings. You simply substitute meaningful strings for the "Normal Data Message" and "Alarm Data Message" given in this example.

2.3.4 Converting Data To Engineering Units

It is possible to convert data to engineering units, such as degrees, feet, etc. In other words, the raw data can be said to represent certain maximum and minimum values. To convert the data, you must use the MAXIMUM and MINIMUM commands. The schematic for these commands is:

```
MAX = nn
MIN=nn
```

where nn is the real max or min. For example, if the data in question represents measurements between +5 volts and -5 volts, you could scale this data for display with the commands:

```
MAX=5 . 0
MIN=-5 . 0
```

The Display Setup software translates this minimum and maximum into a scale and offset, which when applied to the data, scales the largest possible data value to five (from the MAX command), and the smallest possible data value to minus five (from the MIN command). If these commands are not given in the slot specification, the default values represent the full range possible in the data word. That is, the default maximum is 32,767.5 and the default minimum is -32,767.5 for sixteen bit data types. The default maximum is 2,147,483,647.5 and the default minimum is -2,147,483,647.5 for the 32 bit LONG data type. If the data conversion selection is FLOAT or if the slot measurement is defined with an expression (See Section 2.3.6), the data scaling command and setup are not used.

2.3.5 Setting Slot Alarms

To set limits on numeric display parameters expressed in scaled physical units, such that any occurrence of data outside those limits cause the data to be displayed in reverse video use the Alarm commands. These commands are:

```
HI A=nn
LOA=nn
```

where nn represents the High or Low Alarm value. For example, if you wanted an alarm display (reverse video) for any data exceeding +4.5 volts, or below -4.0 volts you must use:

```
HI A=4. 5
LOA=-4. 0
```

The default alarm values are 32,768 and -32,769, which will not allow a reverse video display of any 16 bit raw data, unless the programmer makes other specifications.

The alarm commands for discrete data are the PATTERN and MASK commands. The schematic for these commands:

```
PAT=nn
MAS=nn
```

where nn is the decimal or hexadecimal representation of the pattern or mask. To determine if the discrete is alarm data, the Display Processor will mask the data (with the Mask) and compare the result with the Pattern. If they are equivalent, the data is displayed in reverse video. Masking means compare only the desired bits by denoting these bits with ones. For Example, if you have an eight-bit word and an alarm condition occurs when the high four-bits are all on, a the mask and pattern would be set thus:

```
MAS=/00ff
PAT=/00f0
```

The mask is set to only the last eight bits and the pattern is set the first four of those bits

2.3.6 Processing to be Applied to the Data (Expression Command)

It is possible to display derived data rather than unprocessed data. The processes available include:

Function	Symbol
Addition	+
Subtraction	-
Multiplication	*
Division	/
Sine Function	SIN(x)
Cosine Function	COS(x)
Tangent Function	TAN(x)
Arcsine Function	ASIN(x)
Arctangent Function	ATAN(x)
Logarithm Function	LN(x)
Square Root Function	SQR(x)
Scale ID-Tagged Data Function	SC(m1,m2,ID)
Convert Ones Complement Data	OCM(ID)
Convert Offset Binary Data	OBN(ID)
Convert Sign Magnitude Data	SMG(ID)
Process 16 Bit Unsigned Data	UNS (ID)
Convert Two Data Words to Long	ELONG(ID1, ID2)
Convert Two Data Words to Float	EFLOAT(ID1, ID2)
Convert Two Data Words to MIL STANDARD Float	EMFLT(ID1, ID2)
Convert 4 Data Words to Double	DOUBLE (ID1, ID2, ID3, ID4)
Extract a Bit Field From the Data	BITS(MASK, SHIFTCOUNT, ID) BITS(MASK, >SHIFTCOUNT, ID) BITS(MASK, <SHIFTCOUNT, ID)

One or more of these processes may be applied to one or more data IDs in describing the measurement definition for a slot. These processes are selected with the Expression Command.

The trigonometric, as well as the square root and logarithm functions accept one parameter. The parameter can be the ID of the desired data or any other Real Time Display expression. The trigonometric function parameters should be expressed as radians and must be in the range 2147483647 to -2147483648.

The NATURAL LOG(LN) and SQUARE ROOT(SQR) functions have the same range as the trigonometric functions. The SQR function returns a zero (0) when its parameter is negative.

Data Conversion data and scaling in expressions must be done with the functions numbered 12 through 19. The MIN and MAX commands as well as Data Conversion Commands are ignored in slots that are described with expression commands. The reason for this is to allow unique scaling and conversion of each datum referred to in an expression. Observe that the parameters m1 and m2 (min and max) of the scaling function must be in floating point format (with a decimal point). The order of these first two parameters is arbitrary. If they do not contain decimal points they will be mistaken for IDs.

The ELONG, EFLOAT and EMFLT functions allow the user to extract two sixteen bit words from the data stream, and combine them to form a thirty-two bit word. The EFLOAT function interprets the data as thirty-two bit IEEE format floating point. The EMFLT function interprets the data as thirty-two bit MIL STD format floating point. In all functions, the first parameter (ID 1) is the most significant word while the second parameter (ID2) is the least significant word of the thirty-two bit long word.

The BITS function lets you extract a bit field from the data. A mask and shift can be applied to

the data. The greater or less than characters (“<” or “>”) can be used to indicate the direction of the shift. If no shift character is given the SHIFTCOUNT will indicate a right shift. The MASK and SHIFTCOUNT parameters can be given in either decimal or hexadecimal notation. If decimal notation is used you must not use a decimal point in the number. For an example, say you are interested in the data for ID 0; in particular bit fifteen of that data. To show that bit in the least significant bit position on the Display, you can give the expression:

```
$EX BITS (/8000, >15, 0)
```

The Expression Command allows the Real Time Display programmer to combine these functions with IDs and output from other slots to form new display out-put. The syntax for this command is:

```
$EX expression
```

where \$EX indicates to the TDP compiler that a slot expression follows. The technique for defining the data source in an expression is identical to that used in the data source command. For example, to produce a measurement for display which is the addition of DAC ID 2 and input FIFO ID 4, give the command:

```
$EX $DA 2 + $FI 4
```

The default data source may also be used to shorten the expression statement. For example, if the default data source is currently the Output bus, the command

```
$EX 100 - 150
```

chooses as the slot output measurement, data defined by Output Bus ID 150, subtracted from the data defined by Output Bus ID 100. This previous command (given that the default data source is Output Bus), is identical to:

```
$EX $OP 100 - $OP 150
```

but requires six fewer characters. Obviously, slots that require an Expression command to define the display output, do not use the Data ID command; the Data Sources and IDs are defined in the Expression command.

Note in the previous examples that decimal (also hexadecimal) numbers have been taken to be IDs. To specify a number as a constant in an expression the number must include a decimal point, hence must be given in decimal. Whole numbers may be enumerated as “nn.” rather than “nn.0”. To illustrate, to display three times the value of Output Bus ID 50, give the command:

```
$EX 3.0 * ($OP 50) | or $EX 3. * ($OP 50)
```

Without the decimal point the command:

```
$EX 3 * ($OP 50)
```

actually serves to define the data value as Output Bus ID 50, multiplied by the default data source ID 3.

The parentheses used in the previous two examples are not required, but may be included to improve readability or correctly define arithmetic operations. The arithmetic operators ‘*’ and ‘/’ share the highest precedence while the operators ‘+’ and ‘-’ share the lowest. If there is any doubt concerning the precedence’s effect on the evaluation of an expression, the use of parentheses to clarify the programmer’s intent is advised.

Since each data source in an expression may define data that is not in twos complement form, slots containing an Expression should specify data conversions within the expression command rather than using the data conversion command, as in:

```
$EX SMG($FI 1) / 10.
```

If no Data Conversion Algorithm is specified for an ID in an expression, the data is assumed to be in twos complement form. In the prior example then, the out-put value is determined by converting the data defined by input FIFO ID 1, from sign magnitude form into twos complement form, then dividing this number by ten. The parentheses are required to delimit the data word when using Data Conversion functions in Expression commands.

In addition to the arithmetic operations already described, (addition, subtraction, multiplication, and division), other functions, may be applied to the data. These are the sine, cosine, tangent, logarithmic, and square root function. The syntax for all of these math functions is:

functionname (parameter)

where functionname is one of the previously given math functions, and parameter is given by an ID or an arithmetic expression. For example legal functions might be:

\$EX LN(2)

or

\$EX SIN(COS(2)*SQR(3))

To illustrate and review the Real Time Display setup commands explained thus far, the following example is given:

DSP	Invoke Display Processor
PAG=1 ND	Page 1 is Numeric Display
SLO=4	Slot four
\$DA 4	DAC 4 data source
HEX	Hexadecimal output format
SLO=5	Slot five
FL2	Floating point output format
\$EX SIN(1)*COS(2)	Expression: Sine of ID 1 times the cosine of ID 2
SLO=6	Slot six
FL3	Floating point output format
\$EX SQR(\$FI 50)	Expression: Square root FIFO ID 50
SLO=9	Slot nine
\$EX LN(\$OP /f0)	Expression: Natural log of output bus ID /f0
SLO=10	Slot ten
FL2	Floating point output format
\$EX 1. / TAN(/100)	Expression: 1 divided by the tangent of ID /100
END	End display setup

In this programming example, the Display Setup Mode is first entered with the DSP command. Page one is then selected for setup, and declared to be a Numeric Display. The default data source is then declared to be the DAC bus. Without this declaration the default data selection would remain as the input FIFO bus. The last command given before slot setup is started is the refresh rate command (IS) which declares the Page refresh rate to be one second.

The first slot chosen for setup is slot five. Notice again, that the first three characters in a command word are sufficient for invoking the command. Equally correct is SLOT=5. The display measurement for slot five is defined with an expression. This expression specifies the sine of DAC bus ID 1, multiplied by the cosine of DAC ID 2 as the slot measurement. The parameters of the sine and cosine functions are known to be DAC bus IDs because:

- Decimal numbers with no decimal point, given in Expression commands are understood to be data source identifiers.
- Since no data source is given with the identifiers, the default data source is used, and in this program example, the default data source has been selected as the DAC bus.

Then slot six, nine and ten are set up. After slot ten setup is completed, the program exits Display Setup Mode with the END command.

Scaling the data defined by an expression is NOT done with MIN and MAX commands described earlier. Each ID contained in an expression must be scaled within the expression itself, if scaling is desired. The syntax for scaling data within an expression is:

\$EX SC(m1 ,m2 , ID)

where m1 and m2 are the minimum and maximum desired value. The order of the min and max does not matter as long as one is greater than the other. The parameter ID is the value to be

scaled. m1 and m2 must be decimal numbers (must include decimal points). For example:

```
$EX SIN(SC(0.0, 3.1415, 3))
```

The above command says take the sine of ID 3 between the values of 0 and PI.

It is apparent from the discussions about the Display Setup Expression command, that every slot, even those whose specification requires only a simple ID, can be setup with the expression command. This is not advised, because there is a limited amount of Display Setup Memory dedicated to the storage of the expressions, and the alternate method (previously described, using Data Source and Data Selection commands) for expressing this setup does not use this Expression Table Memory.

2.3.7 Measurement Annotation

To annotate each slot on the Numeric Display provisions are made to allow a measurement name and units string (used to denote the unit of the measurement). When choosing the ten character name to identify the measurement you must use the LABEL command. The syntax for this command is

```
LAB label
```

where label may be any ten character alphanumeric string. To select the ten character 'units' annotation you must use the UNITS command. The syntax for this command is:

```
UNI units
```

where units represents the desired alphanumeric annotation.

2.3.8 Broadcasting Data

You can broadcast the numeric page data to either a multicast socket or to a comport. The command to select the broadcast function can be given within the page setup commands. The syntax for the command to select broadcasting to a socket can be described as:

```
BROADCAST SOCKET ip_address port_number
```

where ip_address represents the IP address of the multicast socket, and where port_number represents the socket port number. A legal command to broadcast the numeric page data to a multicast socket could be given as:

```
BROADCAST SOCKET 224.1.1.1 7301.
```

Since IP multicast addresses must be in the class D range from 224.0.0.0 to 239.255.255.255, an address outside that range will result in a command error.

The syntax for the command to select broadcasting to a comport can be described as:

```
BROADCAST comport [rate] [parity] [word_length] [stop_bits]
```

In this example comport represents the required parameter that specifies the comport to be used. You can use COM, COM1, COM2, COM3, or COM4. COM and COM 1 select comport 1. The remaining parameters are optional (as indicated by the braces) and default values are used if they are omitted from the command line. The default values for each parameter are:

- rate = 9600
- parity = ODD
- word length = 7
- stop bits = 1

To explicitly select a parity you can use OFF, ODD, EVEN, MARK, or SPACE as the parameter. To explicitly set the word length you can use 7 or 8 as the parameter. To explicitly set the stop bits you use 1, 1.5, or 2 as the parameter. To explicitly set the transfer rate you give the number as the parameter. The optional parameters can be given in any order on the command line. Some actual commands to broadcast the data to a comport could look like:

```
BROADCAST COM1 8 38400 EVEN
BROADCAST COM2 OFF 7 2 19200
BROADCAST COM3
```

When a display is activated which has one or more numeric pages defined as broadcast data sources, broadcasting will start immediately, whether or not the defining pages are currently visible. Data will only be broadcast when the TDP is in "run" mode, actively acquiring data. Broadcast messages will be out-put ten times per second.

A broadcast message will begin with a header. The first ten characters are the page title. The next twelve characters represent the IRIG time the data was gathered from the Current Value Tables. Then the header is terminated by a newline character (0xA). The remainder of the message will be made up of one to forty variable length data messages, each of which is terminated by a newline character (0xA).

Each slot that has been programmed on the display page will generate a data message. The messages will follow the header in slot-number order. It is the user's responsibility to not define more messages than can be carried away by the broadcast method. This is particularly important for RS232 broadcasts.

The data messages consist of as many as three tab-separated fields: the label, the data, and the units. If the label or units field are un-programmed for a slot, that field and its tab separator will be omitted.

To illustrate the format of the broadcast data, assume a real time display setup described as:

```
RES DSP
DSP
| Setup display page one, make the display type "Numeric
Data".
PAG=1 ND
| Name the page "TestData".
NAME TestData
| Broadcast the page data to a multicast socket.
BROADCAST SOCKET 224.1.1.2 7302
| Setup for slot one.
SLO=1
$PCM1 1
LAB Pcm 1
UNITS Hex Data
| Setup for slot three.
SLO=3
$PCM1 3
LAB Delta RPM
FL2
| Setups for slot four.
SLO=4
$PCM1 4
FL3
LAB Motor Temp UNITS Deg C
END | End of display setup
```

An excerpt from the data broadcast by the Realtime Display program running this setup could look like:

```
TestData 045140540662 Pcm 1 /4440 Hex Data
Delta RPM 140.04
Motor Temp      116.601 Deg C TestData 045140540755
Pcm 1 /CCC0 Hex Data
Delta RPM 140.04
Motor Temp      129.980 Deg C TestData 045140540849
Pcm 1      /4440 Hex Data
Delta RPM 140.04
Motor Temp      116.601 Deg C TestData 045140540943
Pcm 1 /CCC0 Hex Data
```

```
Delta RPM 500.39
Motor Temp      129.980 Deg C TestData 045140541037
Pcm 1 /CCC0 Hex Data
Delta RPM 500.39
Motor Temp      129.980 Deg C TestData 045140541130
Pcm 1 /CCC0 Hex Data
Delta RPM 140.04
Motor Temp      116.601 Deg C
```

2.3.9 Out Of Limits Display Page Setup

Another Display type provided is called the Out Of Limits Display. The setup procedures for this display type are identical to those used for Numeric Display. The difference between the two page types is apparent only when the Display Program is running. In the Numeric Display the data is presented in normal video unless the alarm values have been exceeded, which causes the data to be displayed in reverse video mode. In the Out of Limits Display the data is not displayed at all unless the slot alarm values have been exceeded. If the data value returns to within the alarm values, the slot data display disappears again. To set up an Out of Limits Display page enter the command:

```
PAGE=nn OL
```

where nn is the page number (1 through 16 are valid). Then select measurements to monitor, and the formats for display as described in the preceding section titled NUMERIC DISPLAY SLOT SETUP .

CHAPTER 3 BAR CHART DISPLAY

3.1 BARCHART DISPLAY SETUP

When the Real Time Display System is in setup mode, it may be set up for Bar Chart Display with the PAGE command

```
PAG=nn BC
```

where nn is a page number between 1 and 16. This will set the display format to "Bar Chart format" until another format is specified.

Bar Chart format has many of the same options Numeric Display has. For example, the NAME command assigns an alphanumeric name to the page, and the SLOT command designates a slot is to be set up. A slot corresponds to one of the 8 bars that may be displayed.

3.2 BARCHART DISPLAY SLOTSETUP

To select a measurement and set its display parameters while in the Bar Chart Display Setup Mode, the SLOT command must be given. The slot number must be specified with the SLOT command as follows:

```
SLOT=nn
```

where nn may be a number from one to eight. In the Bar Chart display, the eight bars used to show the measurement value are displayed on the terminal vertically, spaced evenly on the screen between from left to right. After the Slot command, further commands may be given allowing specification of the measurement and the parametric bounds of its display. These commands permit the programmer to:

- Select the data source and numeric ID of the measurement.
- Select a data conversion algorithm to apply to the data.
- Select arithmetic processes to be applied to the data.
- Select a ten character name with which to identify the measurement on the display.

When the slot setup has been fully specified, the programmer may proceed to set up another slot on the same page by issuing another SLOT command. If the user wishes to clear a slot of previous setup he may do so with the SCL command. Or the programmer may instead choose to begin setup on another page by giving a PAGE command. Upon completion of a slot setup you may exit the Real Time Display Setup Mode by giving the END command. Following is an explanation of the display setup commands that are used to perform the slot setup operations.

3.2.1 Data Source and ID selection

The following is a summary of the data source selection commands. To select the measurement data source and ID you must use the data source selection command:

COMMAND	FUNCTION
\$FI nn	Input FIFO Bus on models 504 or 4504 DIST
\$OP nn	Output Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DA nn	DAC Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DVn nn	CVT partition on models 504 or 4504 DIST, or 1605 PDSP
\$SP nn	Special offset on FIFO bus
\$PCMn nn	Model 1502 or 1602 FSYN Current Value Table
\$FB nn	Feedback port on model 1605 PDSP

The single n appended to \$PCM selects a particular model 1502 or 1602 FSYN card in your system and can be in the range of 1 through 8. It is impossible to have both models 1502 and 1602 FSYN in the same system, so these tokens can be shared between them. Likewise the single n appended to the \$DV selects a different device on the models 504 or 4504 DIST card or the model 1605 PDSP card. You cannot have more than one of any of these three cards in your system, so some tokens are shared between the three.

The nn in the above commands is the numeric ID. The data source and ID may also be selected by giving only the numeric ID. In this case the program uses the currently selected default data source. The default data source is selected when one of the above sources is given without a specified ID.

3.2.2 Data Conversion Algorithm

As described in Section 2, data is converted so that the arithmetic processing can be performed on a uniform, twos complement data type.

The available data conversions are:

COMMAND	DESCRIPTION
2CM	Twos Complement
1CM	Ones Complement
OBN	Offset Binary
SMG	Sign and Magnitude
UNS	Unsigned Short
LONG	32 Bit Long Word
FLOAT	32 Bit Floating Point

3.2.3 Maximum and Minimum Values

Use the MAX and MIN commands to scale data. The syntax for these commands is:

MAX = nn
 MI N=nn

where nn is the real max or min. For example, if the data in question represents measurements between +5 volts and -5 volts, you could scale this data for display with the commands:

MAX=5 . 0
 MI N=-5 . 0

The Display Setup software translates this minimum and maximum into a scale and offset, which when applied to the data, scales the largest possible data value to five (from the MAX command), and the smallest possible data value to minus five (from the MIN command). If these commands are not given in the slot specification, the default values represent the full range possible in the data word. That is, the default maximum is 32,767.5 and the default minimum is -32,767.5 for sixteen bit data types. The default maximum is 2,147,483,647.5 and the default minimum is -2,147,483,647.5 for the 32 bit LONG data type. If the data conversion selection is FLOAT or if the slot measurement is defined with an expression (See Section 2.3.6), the data scaling command and setup are not used.

3.2.4 Origin And Limit

Both the origin and limit of each bar may be specified. The origin is the numeric starting point of the bar, the limit is the ending point. To set the origin, use the command

ORI =nn

where nn is the desired point of origin. To set the limit, use the command

LIM=nn

where nn is the desired limit point.

3.2.5 Measurement Annotation

To annotate each slot on the Bar Chart, provisions are made to allow a measurement name. When choosing the ten character name to identify the measurement you must use the LABEL command. The syntax for this command is

LAB label

where label may be any ten character alphanumeric string.

3.2.6 Arithmetic Processing

All arithmetic functions described in Numeric Display setup mode may be used in Bar Chart setup mode. These include:

Addition

Subtraction

Multiplication

Division

Sine Function

Cosine Function

Tangent Function

Logarithm Function

Square Root Function

Scaling Function

Data Conversion Functions

Longword Function

Floating Point Conversion Function

Mil Standard Floating Point Conversion Function

Double Word Function

Bit Field Extraction Function

One or more of these processes may be applied to one or more data IDs in describing the measurement definition for a slot. These processes are selected with the Expression Command.

All trigonometric functions should be expressed as radians and must be in the range 2147483647 to -2147483648.

The NATURAL LOG(LN) and SQUARE ROOT(SQR) functions have the same range as the trigonometric functions. The SQR function returns a zero (0) when its parameter is negative.

The Expression Command allows the Real Time Display programmer to combine these functions with data IDs and other values to form new display output. The syntax for this command is:

`$EX expression`

where \$EX indicates to the TDP compiler that a slot expression follows. The technique for defining the data source in an expression is identical to that used in the Data Source command. For example, to produce a measurement for display which is the addition of DAC ID 2 and input FIFO ID 4, give the command:

`$EX $DA 2 + $FI 4`

Note in the previous examples that decimal (also hexadecimal) numbers have been taken to be Data Source identifiers. To specify a number as a constant in an expression the number must include a decimal point, hence must be given in decimal. Whole numbers may be enumerated as "nn." rather than "nn.0". For example, to display three times the value of Output Bus ID 50, give the command:

`$EX 3.0 * ($OP 50) | or $EX 3. * ($OP 50)`

writing the above command without the decimal point:

`$EX 3 * ($OP 50)`

actually serves to define the data value as Output Bus ID 50, multiplied by the Default Data Source ID 3.

The parentheses used in the previous two examples are not required, but may be included to improve readability or correctly define arithmetic operations. The arithmetic operators '*' and '/' share the highest precedence while the operators '+' and '-' share the lowest. If there is any doubt concerning the precedence's effect on the evaluation of an expression, the use of parentheses to clarify the programmer's intent is advised.

Since each data source in an expression may define data that is not in twos complement form, slots requiring an Expression specify data conversions in the expression command rather than using the data conversion command, as in:

`$EX SMG($FI 1) / 10.`

If no Data Conversion Algorithm is specified for a data word in an expression, the data word is assumed to be in twos complement form. In the prior example then, the output value is determined by converting the data defined by input FIFO ID 1, from sign magnitude form into twos complement form, then dividing this number by ten. The parentheses are required to delimit the data word when using Data Conversion functions in Expression commands.

In addition to the arithmetic operations already described, (addition, subtraction, multiplication, and division), other functions, as listed earlier may be applied to the data. These are the sine, cosine, tangent, logarithmic, and square root function. The syntax for all of the math functions is:

`functionname(parameter)`

where functionname is one of the previously given math functions, and parameter is an ID or an expression (which may include functions). To illustrate their application and review the Real Time Memory setup commands explained thus far, the Display setup program is given:

<code>SLO=2</code>		Slot two
<code>\$EX SQR(\$FI 50)</code>		Expression: Square root of FIFO ID 50
<code>SLO=4</code>		Slot four
<code>\$OP</code>		New default - Output Bus is data source
<code>1CM</code>		New default - Ones complement data type

```

/f0          | Data source is Output Bus, ID /f0
SLO=8       | Setup slot eight
2           | Data Source is Output Bus, ID 2 |
END         | End display setup

```

In this programming example, the Display Setup Mode is first entered with the DSP command. Page one is then selected for setup, and declared to be a Bar Chart. The default data source is then declared to be the DAC bus. Without this declaration the default data selection would remain as the input FIFO bus.

The first slot chosen for setup is slot five, given with the SLO=5 command. Notice again, that the first three characters in a command word are sufficient for invoking the command. Equally correct is SLOT=5. The display measurement for slot five is defined to be ID 12.

Slot six's display measurement is then declared to be the square root of the data defined by input FIFO ID 50. Slot nine and ten are also setup for display. After slot ten setup is completed, the program exits Display Setup Mode with the END command.

Scaling the data defined by an expression is NOT done with MIN and MAX commands described earlier. Each ID contained in an expression must be scaled within the expression itself, if scaling is desired. The syntax for scaling data within an expression is:

```
$EX SC(m1, m2, ID)
```

where m1 and m2 are the minimum and maximum desired value. The order of the min and max does not matter as long as one is greater than the other. The parameter ID is the value to be scaled. m1 and m2 must be decimal numbers (must include decimal points). For example:

```
$EX SIN(SC(0.0, 3.1415, 3))
```

The above command says take the sine of ID 3 between the values of 0 and 3.1415.

It is apparent from the discussions about the Display Setup Expression command, that every slot, even those whose specification requires only a simple ID, can be setup with the expression command. This is not advised, because there is a limited (though adequate) amount of Display Setup Memory dedicated to the storage of the expressions, and the alternate method, (previously described, using Data Source and Data Selection commands) for expressing this setup does not use this Expression Table Memory.

CHAPTER 4 CROSS PLOT DISPLAY

4.1 CROSS PLOT DISPLAY SETUP

The Real Time Display System may be setup for Cross Plot Display with the PAGE command

```
PAG=nn CP
```

where nn is a page number between 1 and 16. This will set that page's display-type default to "Cross Plot format", until another format is specified.

In Cross Plot up to four functions may be displayed either simultaneously on a single graph, or individually on multiple graphs. Each graph is displayed in standard X-Y Cartesian coordinate format. The SLOT command corresponds to one of the 4 functions that may be displayed. Each SLOT command must have two data sources, one representing the X parameter, the other the Y. A maximum of four slots may be specified. If the command

```
SI NG
```

is given, a single X-Y graph is drawn, and the functions specified are drawn on that graph. If the command

```
MULT
```

is given, up to four individual graphs are drawn, each representing a single function. The program defaults to MULT.

The X-axis may be expressed as Time in a cross plot. When the X-axis is time, the display system will automatically produce a strip chart by scrolling the display when the data reaches the limit of the X-axis. Once again, the NAME command gives an alphanumeric name to the page.

4.2 CROSS PLOT DISPLAY SLOTSETUP

The SLOT command must be given to select a measurement and describe its for-mat for display while in the Cross Plot Display Setup Mode. The slot number must be given with the SLOT command as follows:

```
SLOT=n
```

where n may be a number from one to four.

The command SCL clears any previous axis setup. To clear the entire slot, each axis must be cleared individually. After the SLOT command, further commands may be given, allowing specification of the measurement and the format of its display. These commands permit the programmer to:

1. Select Point plot or Vector plot.
2. Declare an axis for setup
3. Select the data source and numeric ID of the measurement, for both X and Y variables.
4. Select data conversion expressions to apply to the data.
5. Select defined mathematical functions to be applied to the data.
6. Set limits on Cross Plot display parameters.
7. Select a ten character name with which to identify the measurement on the display for each axis.
8. Select annotation intervals of the X and Y axis.

When the slot setup has been fully specified the programmer may proceed to set up another slot on the same page by issuing another SLOT command. Or the programmer may choose to begin setup on another page by giving a PAGE command. Another option upon completing a slot setup

specification is to exit the Real Time Display Setup Mode by giving the END command.

4.2.1 Vector Plots And Point Plots

There are two types of cross plots: vector plots, and point plots. A vector plot connects the data sample points with lines when it has been determined that both the x and y coordinates of the data point have changed since the first sampled point. Until that time, the plot is drawn as a point plot. The point plot leaves the data sample points distinct. The program defaults to vector plot. To change the plot to a point plot use the command:

POI

To change back to a vector plot the command

VEC

must be entered. Since the slots always default to vector plots, the POI command must be specified for each slot that is to be a Point Plot.

4.2.2 Axis Selection

After the SLOT command has been entered it is important to select which axis will be defined first. The commands to specify the axes are:

XAX

and

YAX

for X-axis and Y-axis respectively. The program defaults to the X-axis, so if no axis is specified, the first ID source will be plotted on the X-axis.

4.2.3 Data Source and ID selection

After the SLOT and axis commands have been issued you may select the measurement data source and ID by using the data source selection command. The following is a summary of the data source selection commands. To select the measurement data source and ID you must use the data source selection command:

COMMAND	FUNCTION
\$FI nn	Input FIFO Bus on models 504 or 4504 DIST
\$OP nn	Output Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DA nn	DAC Bus on models 504 or 4504 DIST, or 1605 PDSP
\$DVn nn	CVT partition on models 504 or 4504 DIST, or 1605 PDSP
\$SP nn	Special offset on FIFO bus
\$PCMn nn	Model 1502 or 1602 FSYN Current Value Table
\$FB nn	Feedback port on model 1605 PDSP

The single n appended to \$PCM selects a particular model 1502 or 1602 FSYN card in your system and can be in the range of 1 through 8. It is impossible to have both models 1502 and 1602 FSYN in the same system, so these tokens can be shared between them. Likewise the single n appended to the \$DV selects a different device on the models 504 or 4504 DIST card or the model 1605 PDSP card. You cannot have more than one of any of these three cards in your

system, so some tokens are shared between the three.

The nn is the desired numeric ID. The data source and ID may also be selected by giving only the numeric ID. In this case the program uses the currently selected Default Data Source.

4.2.4 Time Axis Function

Either axis may be expressed as a function of time. The command

TIME

will specify that the axis currently being programmed is the time function. Time is expressed in milliseconds, and a limit should be given for the time function (e.g. LIM=30000 sets the page limit to graph 30 seconds of data).

4.2.5 Data Conversion Algorithm

Data is converted so that the arithmetic processing can be performed on a uniform twos complement data type.

The following commands may be used for data conversion (for more detail see Section 2):

TO CONVERT	USE
Twos Complement	2CM
One's Complement	1CM
Offset Binary	OBN
Sign Magnitude	SMG
Unsigned Short	UNS
32 Bit Long Word	LONG
32 Bit Floating Point	FLOAT

If for some reason it becomes necessary to change the Data Conversion selection for a slot to twos complement from some other earlier selection, the 2CM command is required, even if data is known to be twos complement.

4.2.6 Maximum And Minimum Values

The correct syntax for the Maximum and Minimum commands is:

MAX=nn

or

MIN=nn

where nn is the real maximum or minimum. The Maximum value represents the 100% position, the Minimum is the 0 point. For example, if the data in question ranges from positive full scale to negative full scale, but in actuality represents measurements between 0 and 1000 meters, it would be possible to display this data with the commands:

MAX=1000 . 0
MIN=0 . 0

The Display Setup software translates this minimum and maximum into a scale and offset, which when applied to the data, scales the largest possible data value to 1000 (from the MAX

ommand), and the smallest possible data value to zero (from the MIN command).

On the other hand, if the data has been scaled prior to display you do not need to scale with the MIN and MAX commands, instead, look at the desired data range by setting the origins and limits of the X and Y axes (See next Section). If these commands are not given in the slot specification, the default values represent the full range possible in the data word. That is, the default maximum is 32,767.5 and the default minimum is -32,767.5 for sixteen bit data types. The default maximum is 2,147,483,647.5 and the default minimum is -2,147,483,647.5 for the 32 bit LONG data type. If the data conversion selection is FLOAT or if the slot measurement is defined with an expression (See Section 2.3.6), the data scaling command and setup are not used.

4.2.7 Origin and Limit

Both the origin and limit of each axis may be specified. The origin is the numeric starting point of the axis, the limit is the ending point. To set the origin, use the command

```
ORI =nn
```

where nn is the desired point of origin. To set the limit, use the command

```
LIM=nn
```

where nn is the desired limit point. So if the Y-axis data lies between 0 and 10, and has been scaled prior to display, the commands:

```
YAX          | Begin Y-axis setup
2            | ID 2
ORI =-1      | Origin is -1
LIM=11       | Limit is +11
```

will set the screen boundaries for the Y-axis.

When the X-axis data source is TIME, you should set the LIM of the X-axis in milliseconds. Therefore the commands

```
XAX
TIME
LIM=25000
```

will set the time limit to be 25 seconds. The origin need not be set since it automatically is 0 when an axis is TIME.

4.2.8 Measurement Labels

To label each function on the Cross Plot Display a measurement name of up to 10 characters may be specified for each axis. When choosing the ten character name to identify the measurement use the LABEL command. The syntax for this command is:

```
LAB label
```

where label is the desired 32 character name. The label appears underneath the graph it describes, in the form

```
"Y-axis label" vs "X-axis label"
```

4.2.9 Axis Annotation (Numeric)

Annotation of the X and Y axis may be accomplished by utilizing the command

```
ANN=interval [n]
```

where interval represents a number within the Range of the axis endpoints (ORI < interval < LIM). The axis will then be hash-marked at every interval. (e.g. if interval is ten, every ten units will be hash-marked.)

If the optional parameter n is specified, every nth interval will be numerically annotated. For example, if n is 1, every hash-mark is annotated, if n is 2, every other hash-mark is annotated, etc. Choose logical intervals and numeric annotation, because intervals that are too long will place the

a

notation off the page, and intervals that are too short will overlap. If the single axes command is used to plot several slots on one set of axes, slot one's annotation is used, although the other slots may have different origins and limits.

4.2.10 Arithmetic Processing

All arithmetic functions in Numeric display setup may be utilized in Cross Plot format. These are:

- Addition
- Subtraction
- Multiplication
- Division
- Sine Function
- Cosine Function
- Tangent Function
- Logarithm Function Square Root Function Scaling Function
- Data Conversion Functions Longword Function
- Floating Point Conversion Function
- Mil Standard Floating Point Conversion Function
- Double Word Function
- Bit Field Extraction Function

One or more of these processes may be applied to one or more data words in describing the measurement definition for a slot. These processes are selected with the Expression Command.

`$EX expression`

where \$EX indicates to the TDP compiler that a slot expression follows. Example:

`$EX 1234 + 4111`

this means ID 1234 + ID 4111. The arithmetic operators '*' and '/' share the highest precedence while the operators '+' and '-' share the lowest. If there is any doubt concerning the precedence's effect on the evaluation of an expression, the use of parentheses to clarify the programmer's intent is advised.

Slots requiring an Expression may specify data conversions in the expression command as in:

`$EX SMG($FI 1) / 10.`

If no Data Conversion Algorithm is specified for a data word in an expression, the data word is assumed to be in twos complement form. The parentheses are required to delimit the data word when using Data Conversion functions in Expression commands.

The command schematic for sine, cosine, tangent, logarithmic, and square root is:

`functionname (parameter)`

where functionname is one of the previously given math functions, and parameter is given by an ID or an expression.

Scaling the data defined by an expression is NOT done with MIN and MAX commands described earlier. Each ID contained in an expression must be scaled within the expression itself, if scaling is desired. The syntax for scaling data within an expression is:

`$EX SC(m1, m2, ID)`

where m1 and m2 are the minimum and maximum desired value. The order of the min and max does not matter as long as one is greater than the other. The parameter ID is the value to be scaled. m1 and m2 must be decimal numbers (must include decimal points). For example:

\$EX SIN(SC(0.0, 3.1415, 3))

The above command says take the sine of ID 3 between the values of 0 and PI.

To illustrate their application and review the Real Time Memory setup commands explained thus far, the Display Setup Program is given:

DSP		Begin Display Setup
PAG=2 CP		Page two is Cross Plot
\$DA		Default data source: DAC
SLO=1		Begin Slot One setup
POI		Point plot
XAX		Begin X-axis setup
TIME		X-axis is time
LIM=30000		30 second plot
ANN=5000 2		Every 5 seconds mark the axis
LAB time		numerically annotate every 2nd mark.
		Label: time
YAX		Begin Y-axis setup
\$EX SIN(1) * COS(2)		Expression: sine of ID 1 multiplied
		by the cosine of ID 2.
ORI=0		Origin is zero.
LIM=1000		Limit is 1000.
ANN=100 1		Every 100 units mark the axis,
		numerically annotate every mark.
LAB waveform		Label: waveform
END		Exit Display setup

CHAPTER 5 EMBEDDED TIME

5.1 DESCRIPTION OF EMBEDDED TIME

The Real Time Display Program gets its “time of year” data from an IRIG time code signal, or from the computer’s system time if the IRIG time code signal is not available. This time data is used to annotate the display and to provide the time base for the graph displays.

In addition to these time sources you can define time words that are embedded in your PCM stream. You use this embedded time data to drive the Real Time Display Program graph displays, or as a data source in the numeric type of display page. This embedded time data must be in a 16-bit standardized time word format, which is described the following text.

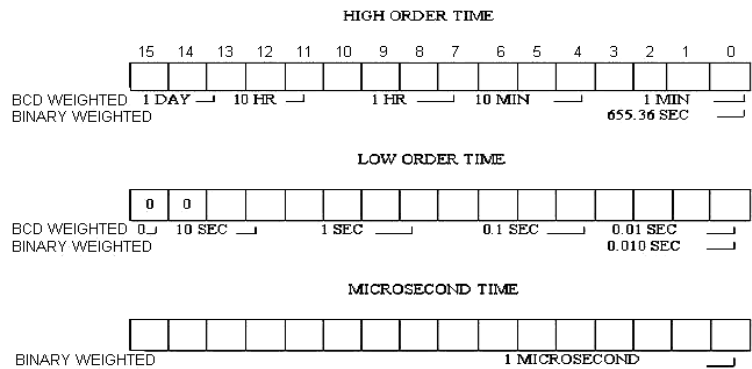
In the 16-bit standardized time word format, there shall be three words dedicated to providing timing information. These words are designated high order time, low order time, and microsecond time. High order and low order time words shall be binary or binary coded decimal (BCD) weighted, and microsecond words shall be binary weighted.

The microsecond time word shall have a resolution of 1 microsecond; that is, the least significant bit has a value of 0.000001 second. This word shall increment until it attains a value of 10 milliseconds at which time it will be reset to zero. The maximum value of the counter is 9999 (decimal).

The low order time word shall have a resolution of 10 milliseconds; that is, the least significant bit of the low order time word shall have a value of 0.010 second.

The high order time word shall have a resolution of 655.36 seconds when binary weighted. When BCD weighted, the least significant bit of the high order time word shall have a value of 1 minute.

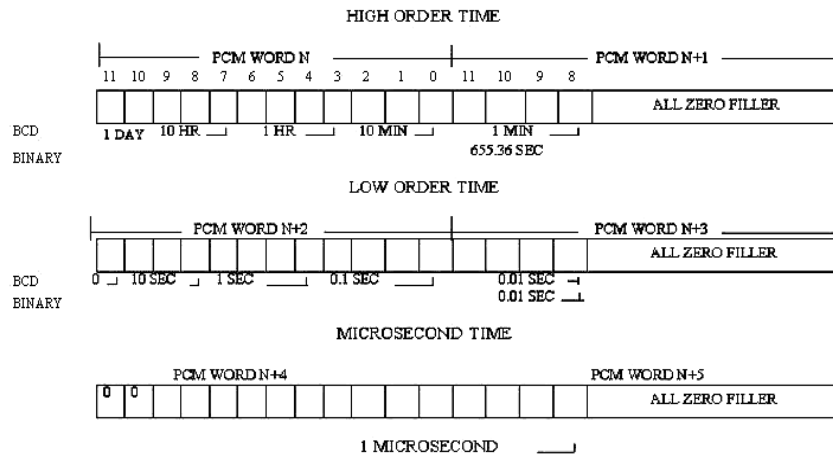
The following is a diagram of the time words.



Embedded Time Words

Time word insertion into PCM word sizes other than 16 bits shall be as follows: high order, low order, and microseconds shall be inserted into PCM words with time word bits occupying contiguous bits locations in the PCM word. The time word shall occupy contiguous PCM data words until the time word is contained in the PCM stream. If the time word size is not an integer multiple of the PCM word size and there are unused bits in the PCM word, the remaining unused bits in the last PCM word that contains the time word shall be fill bits with value 0.

The following is a diagram of time words as they would appear in a PCM data stream of 12-bit words.



Time Words in a 12-Bit Data Stream

You must use the embedded time setup commands to define and enable embedded time in your PCM data stream. The time setup commands must be given within a display setup block and they must precede any page and slot setups. The time setup commands must start with the “TIME” command, and must be terminated with an “END” command. In addition to these two commands there are six other time commands, each of which must be given to complete a valid embedded time setup. None of the commands may be omitted in favor of default settings.

To illustrate, a display setup block that included only embedded time setup, without any page or slot setups might look like this:

```
DSP
TIME IRIG BIN
LEN=16
$PCM1
RJ
HIG=0
LOW=1
MIC=2
END
END
```

If any page or slot setups were to be added to this setup block, they must be placed between the two “END” commands. (The first “END” completes the embedded time setup, while the second terminates the display setup block.)

5.2 EMBEDDED TIME COMMANDS

A description of each of the embedded time setup commands follows.

5.2.1 Time Command

The TIME command is given to initiate the description of the embedded time data, and to select which of the two available embedded formats will be used. You can choose either binary time data, or binary coded data (BCD) for the embedded time format. The command to select binary data format is given as:

```
TIME IRIG BIN
```

The command to select BCD data format is given as:

```
TIME IRIG BCD
```

After giving either of these commands, the other seven time commands (including the END

ommand) must be given to successfully complete the embedded time setup.

A variation of the TIME command is available for selecting internal time. Internal time is the standard “time of year” data normally used by the Real Time Display Program. This is generated from an IRIG time code signal, or from the computer’s system time if the IRIG time code signal is not available. When this command is used it must be given with an END command, but naturally the other embedded time commands are not used. The syntax for using the TIME command to select internal time is:

```
TIME INTERNAL
END
```

5.2.2 Length Command

The length command (uses the LEN token) is used to declare the size of the words in the PCM stream that contain the embedded time data. The length can be a value ranging from 4 through 32. To declare the size of the words in the PCM stream that contain the embedded time data to be 8 bits, use the following command:

```
LEN=8
```

5.2.3 Data Source Command

The embedded time data in the PCM data stream is gathered by the Real Time Display Program from the current value table (CVT) of one of your Acroamatics telemetry boards, like one of your model 1602 PCI Frame Synchronizer cards or your model 1605 PCI Data Distribution card. Use the data source command to let the display program know from which device CVT to get the data. Use one of the data source tokens described in section 2.3.1 of this document to indicate the CVT to be used. To indicate that your model 1602 PCI FSYN card, unit 1 is the source of the embedded time data, give the command:

```
$PCM1
```

5.2.4 Data Justification Command

The embedded time data in the PCM data stream is gathered by the Real Time Display Program from the current value table (CVT) of one of your Acroamatics telemetry boards, like one of your model 1602 PCI Frame Synchronizer cards or your model 1605 PCI Data Distribution card. The CVT is comprised of 32-bit words. When the data words in the PCM stream are less than 32 bits, the data can be either left or right justified in the CVT. More accurately, if the data words from the PCM data stream are between 16 and 31 bits, the data can be left or right justified within the device CVT 32-bit word. If the data is less than 16 bits it can be left or right justified within the least significant 16 bits of the device CVT 32-bit word.

Use the RJ or LJ command to indicate to the Real Time Display Program which data justification is being used.

5.2.5 High Order Time Word Command

The time data for each of the high, low and microsecond time words is gathered from one or more locations in the CVT, depending on the word size being used in the PCM stream. The time word commands are used to tell which identifiers are used for accessing the various time words. Use the “high order time word command” to indicate the identifier used for accessing the high order time word data in the CVT. The parameter given with the HIG= command can be the literal number of the ID value (hexadecimal values require a leading forward-slash character). Also, the ID parameter given with the HIG= command can be the symbol associated with the high order time word. For PCM streams in which the word size is less than 16 bits, the ID given with this command identifies the first data word in the stream that contains high order time word data. An example of setting the ID for the high order embedded time word using a literal decimal number is:

```
HIG=20
```

5.2.6 Low Order Time Word Command

The time data for each of the high, low and microsecond time words is gathered from one or more locations in the CVT, depending on the word size being used in the PCM stream. The time word commands are used to tell which identifiers are used for accessing the various time words. Use the “low order time word command” to indicate the identifier used for accessing the low order time word data in the CVT. The parameter given with the LOW= command can be the literal number of the ID value (hexadecimal values require a leading forward-slash character). Also, the ID parameter given with the LOW= command can be the symbol associated with the high order time word. For PCM streams in which the word size is less than 16 bits, the ID given with this command identifies the first data word in the stream that contains low order time word data. An example of setting the ID for the low order embedded time word using a literal hexadecimal number is:

```
LOW=/20
```

5.2.7 Microsecond Time Word Command

The time data for each of the high, low and microsecond time words is gathered from one or more locations in the CVT, depending on the word size being used in the PCM stream. The time word commands are used to tell which identifiers are used for accessing the various time words. Use the “microsecond time word command” to indicate the identifier used for accessing the microsecond time word data in the CVT. The parameter given with the MIC= command can be the literal number of the ID value (hexadecimal values require a leading forward-slash character). Also, the ID parameter given with the MIC= command can be the symbol associated with the microsecond time word. For PCM streams in which the word size is less than 16 bits, the ID given with this command identifies the first data word in the stream that contains microsecond time word data. An example of setting the ID for the low order embedded time word using a symbol is:

```
MI C=MI CROWD
```

This example assumes that it is used in a setup script in which the string “MICROWD” has been defined as a symbol that identifies the microsecond time word in the PCM data stream. Refer to Acroamatics document number 6000345, in chapter 5 to learn about defining and using symbols in your Acroamatics telemetry setup scripts.

5.2.8 End Command

When all of the required commands have been given to complete your embedded time setup, you must give the END command, as shown in the example in section 5.1.

You can follow this NED command with commands to set up your display pages and slots as described in chapters 2-4 in this manual.

CHAPTER 6 SETUP AND LAYOUT FILES

6.1 SETUP FILES

The Real Time Display Program presents multiple display views within the program's main window. Each view window within the main window can display 16 different pages of Real Time Display. Each view window is associated with a display setup file (which has a .rtd extension). Each view window can be associated with a unique setup file, or multiple view windows can be associated with a common setup file.

The setup files (.rtd extension) can be created in two different ways. You can use the TDP Compiler (tdpc.exe) to process Real Time Display Setup script, as described in this document. When you finish your setup by giving the "END" command a display setup file will be written to the disk. You can let the compiler choose the setup file name by default, or you can give the name explicitly.

To let the compiler name the setup file by default start your setup script with this command:

```
DSP
```

When the name is chosen by default the environment variable "DSPMEM" in the TDP configuration file (C:\tdpc.cfg) is used to give the name. The tdpc.cfg file shipped with Acroamatics installations have this variable set:

```
DSPMEM=C:\tdpsys\DI SPLAY.RTD
```

If you remove this line from your configuration file the compiler will save the setup file under the name "DISPLAY.RTD" in the program's working (current) directory.

As described in chapter 2, to name the setup file explicitly you begin your display setup script the the command:

```
DSP Filename.RTD
```

The second way to create a display setup file is to use the Real Time Display Program (display.exe) setup menu. Directions on how to use the menu to generate pages of display views, and how to save those setups into a display setup file are given in the display.exe help. Click on "Index" in the Help drop-down menu to view the display.exe help.

While running the display.exe program you can open new view windows associated with a display setup file by clicking "Open setup file" in the "File" drop-down menu. You can open a new display view window which is unassociated with any display setup file for the purpose of creating a new setup file with subsequent menu operations by clicking "New setup file" in the "File" drop-down menu. These operations are described in the display.exe help.

6.2 LAYOUT FILES

When you exit the Real Time Display Program (display.exe) all the information that is needed to restore the same view windows within the program main window is stored automatically in a display layout file (these have a .dlo extension) named c:\tdpsys\MdiDisplayAutoSave.dlo. Each time you start the display program by clicking on display.exe in Windows Explorer the program uses this file to restore the same screen display present at the last program exit. This file stores the location and setup file association for each view window within the display.exe main window. It also stores the screen location of the display.exe main window.

If the c:\tdpsys\MdiDisplayAutoSave.dlo file is not present when display.exe starts up, the program will use the default name display setup file (.rtd extension) to create the display view. If there is no default named setup file the display.exe program will create and use a new, uninitialized setup file called "Untitled.rtd".

You can save the current display.exe layout into a layout file (.dlo extension) that you name by clicking on "Save Screen Layout" on the display.exe "File" drop-down menu. You can restore

this screen layout at a future time by clicking on “Open Screen Layout” in the display.exe “File” drop-down menu. These operations are also described in the display.exe help.

If you start display.exe from a command prompt you can give the name of any layout file as a command argument (eg “display MyLayout.dlo”) to cause the display program to use this layout file rather than c:\tdpsys\MdiDisplayAutoSave.dlo. Or more simply you can double click on a layout file (.dlo extension) in Windows Explorer.

Additionally you can double click on any display setup file (.rtd extension) in Windows Explorer, or name any display setup file in a command prompt display.exe command argument and the program will start up using only a view window for that file.

CHAPTER 7 PROGRAMMING EXAMPLES

7.1 INTRODUCTION

The following is a programming example of the Real-Time Telemetry Display System. Characters that appear after a | are comments, and will not affect the program. The Display setup program is entered with the DSP command.

Page 1 is a Bar Chart display with slots 4, 6, 10, 12, 16 being used for data display. Slot 6 and 10 are expressions while the other slots are data IDs.

Page 2, whose name is NUMRC DISP, is setup to be a numeric display, with slots 1 5, 6, and 7 being used for data display. Notice that slot 7 is used for an expression, and that slot 6 is binary displayed discrete data.

Page 3 is a multiple Cross Plot each function will be displayed in its own X-Y coordinate plane.

DSP is ended with the END command.

The following diagrams represent the display output of the example program.

DSP PAG=1 BC	Invoke Display processor First page: Bar Chart
SLO=2 \$FI 50 MIN= 10 MAX= 184 LAB RADAR ALT	Second slot FIFO data ID 50 Minimum is 10 Maximum is 184 Label: RADAR ALT
SLO=6 \$DA \$EX (SQR(2.0)/2.0)*16 MIN=0 MAX=2 00000 LAB RAN TO DES	Sixth slot DAC data Expression: square root of divided by 2, times ID 16 Minimum is 0 Maximum is 200000 Label: RAN TO DES
SLO= 4 \$EX 200000-6.0 MIN=0 MAX=2 00000 LAB RAN FRM ZE	Fourth slot Expression: 200000 Minus 6.0 Minimum is 0 Maximum is 200000 Label: RAN FRM ZE
SLO=6 \$OP 30 LAB ATTACK MOD MIN=0 MAX= 181	Sixth Slot Output Bus Data ID 30 Label: ATTACK MOD Minimum is 0 Maximum is 181
SLO=8 \$OP 35 LAB ATTACK DEF	Setup Slot 8 Choose Output Bus as Data Source ID 35 Set The Slot Label MIN=181 MAX= 501
PAG=2 ND NAM NUMRC DISP	Setup Page Two, Numeric Display Assign a Page Name
SLO=1 \$FI 47 FL3 MIN=0 MAX=250	Start Setup for Slot 1 Select FIFO ID 47 as Data Source Select a Floating Point Format Data will be scaled from 0 to 250 Completes slot scaling

HIA=200 LOA=20		Set High Alarm for Slot Set Low Alarm
SLOT=5 1 OBN Label DELTA_V units kph maximum=1 minimum=0 FL5		Slot 5 Setup ID 1 Data is Offset binary Set Slot Label Set Slot Units String Data is Scaled From 0 to 1 Completes Scaling Decimal No. with 5 places after decimal pt
SLOT=6 2 Label TARGET units Meters FL2		Slot 6 Setup Select Slot ID Set Label Set Units String Decimal No. with 2 places after decimal pt.
SLOT=7 \$EX SIN(4) units CT Label CHNL7 BIN		Setup For Slot 7 Choose an Expression Units String Label String Binary Data Display
PAGE=3 CP SLOT=1 Xaxis \$EX SIN(2. *(SC(6.2832, 0. , 6)))		Page two is cross plot page Begin setup for slot 1 Begin X-axis setup A scaled expression: 2 times the sine of ID 6, which is scaled between 0 and 2 PI.
limit=1.5 origin=-1.5 label SIN(2X)		Set the Graph Endpoints Set Graph Origin Value Select Slot Label
Yaxis SMG \$ex SIN(3. *(SC(6.2832, 0. , 6)))		Begin Y-axis setup A scaled expression: 3 times the sine of ID 6, which is scaled between 0 and 2 PI.
origin=-1.5 limit=1.5 label SIN(3X)		Set Origin And Limit Select Label
SLOT=2 Time limit=10000 annotate=5000 2		Begin Slot 2 Setup X-axis is TIME 10 seconds Set Annotation Interval & Marker Rate
Yaxis 2 maximum=1 minimum=0 origin=0 limit=1 annotate=1 1		Begin Y-axis setup ID 2 Scale The Data From 0 to 1 Set Graph Origin And Limit Set Annotation Interval & Marker Rate
END		Ends Display Setup

