LIST OF EFFECTIVE PAGES

The manual pages listed below that are affected by a current change or revision, are so identified by a revision number.

Date of Issue for original and changed pages are:

Original ...................... 0 ...................... January 1998

TOTAL NUMBER OF PAGES IN THIS MANUAL IS 1242 CONSISTING OF THE FOLLOWING

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>VOLUME I</td>
<td>VOLUME II</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Title ............................................... 0</td>
<td>Title ............................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Copyright .......................................... 0</td>
<td>Copyright .......................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A ..................................................... 0</td>
<td>i through ix ........................................ 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B Blank ............................................. 0</td>
<td>x Blank ............................................. 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>i through xviii ................................. 0</td>
<td>7-1 ................................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-1 ................................................. 0</td>
<td>7-2 Blank ........................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-2 Blank ......................................... 0</td>
<td>8-1 through 8-73 .................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2-1 through 2-34 ................................. 0</td>
<td>8-74 Blank ......................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-1 through 3-105 ............................... 0</td>
<td>9-1 through 9-405 .................................. 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-106 Blank ...................................... 0</td>
<td>9-406 Blank ....................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4-1 through 4-14 ................................. 0</td>
<td>10-1 through 10-52 ................................ 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5-1 through 5-48 ................................. 0</td>
<td>11-1 through 11-77 ................................ 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6-1 through 6-173 ............................... 0</td>
<td>11-78 Blank ....................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6-174 Blank ...................................... 0</td>
<td>12-1 through 12-163 .............................. 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Index-1 through Index-18 ...................... 0</td>
<td>12-164 Blank ...................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>......</td>
<td>......</td>
<td>......</td>
<td>......</td>
</tr>
<tr>
<td>A-1 ................................................... 0</td>
<td>A-2 Blank .......................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B-1 ................................................... 0</td>
<td>B-1 ................................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B-2 Blank ......................................... 0</td>
<td>C-1 through C-7 ..................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C-8 Blank ......................................... 0</td>
<td>D-1 ................................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D-1 ................................................... 0</td>
<td>D-2 Blank .......................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E-1 through E-3 .................................. 0</td>
<td>E-4 Blank .......................................... 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Index-1 through Index-18 ...................... 0</td>
<td>......</td>
<td>......</td>
<td>......</td>
</tr>
</tbody>
</table>

A/B Blank
PREFACE

SCOPE

This Two Volume Manual contains instructions for remotely operating the IFR-1900 CSA Communication Service Monitor and Tri-Band/Dual Mode Cellular System Analyzer. The instruction level is relatively basic and presupposes no previous experience on the part of the operator with remote operation of a communication service monitor or cellular system analyzer. A basic understanding of communication electronics and cellular system formats is helpful. It is strongly recommended that the operator be thoroughly familiar with this manual as well as the Operation Manuals of the Test Set before attempting to operate the Test Set, remotely.

ORGANIZATION

The IFR-1900 CSA TMAC Users Manual is composed of the following volumes and sections:

**VOLUME I - IFR-1900 CSA GENERAL AND HOST SPECIFIC TMAC**

**SECTION 1 - INTRODUCTION**

Provides an introduction to the TMAC language.

**SECTION 2 - TMAC (TEST MACRO LANGUAGE) OVERVIEW**

Describes and explains the various features of the TMAC language, focusing primarily on General TMAC.

**SECTION 3 - GENERAL TMAC COMMANDS**

Lists all General TMAC Commands alphabetically and provides a detailed definition of each command.

**SECTION 4 - CREATING AND LOADING TMAC PROGRAMS**

Provides step-by-step procedures and examples for creating and loading TMAC programs.

**SECTION 5 - HOST SPECIFIC TMAC QUICK REFERENCE LIST**

Briefly lists the IFR-1900 CSA HOST TMAC commands in alphabetical order.

**SECTION 6 - HOST SPECIFIC TMAC COMMANDS**

Lists and details the Specific TMAC commands for the IFR-1900 CSA HOST. Commands are arranged by Operation Mode for convenience.
SECTION 7 - INTRODUCTION TO VOLUME II

Provides an organization of Volume II and definition of the configuration of the Test Set configuration.

SECTION 8 - SPECIAL TEST SPECIFIC TMAC QUICK REFERENCE LIST

Briefly lists the IFR-1900 CSA Special Test Specific TMAC commands in alphabetical order.

SECTION 9 - SPECIAL TEST SPECIFIC TMAC COMMANDS

Lists and details the Specific TMAC commands for the IFR-1900 CSA Special Test. Commands are arranged by Operation Mode for convenience.

SECTION 10 - SPECIAL TEST PROGRAM EXAMPLES

Provides functional Special Test program examples.

SECTION 11 - IS-136 COMMAND REFERENCE

Provides tables showing the relationship between IS-136 Layer 3 Messages and associated IFR-1900 CSA Special Test TMAC commands.

SECTION 12 - SPECIAL TEST KEY WORD INDEX

Provides a permuted index of all of the Special Test commands in the IFR-1900 CSA TMAC Users Manual. **Bold** words in the center column are the particular key words being indexed. Each full command is indexed by each word in the command.

NOMENCLATURE

The IFR-1900 CSA Test Set consists of the following:

<table>
<thead>
<tr>
<th>FUNCTION</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Service Monitor</td>
<td>HOST</td>
</tr>
<tr>
<td>Tri-Band/Dual Mode Cellular System Analyzer</td>
<td>SPECIAL TEST or Sp Tst</td>
</tr>
</tbody>
</table>

The Special Test (Tri-Band/Dual Mode Cellular System Analyzer) utilizes the test equipment contained in the Communication Service Monitor portion of the IFR-1900 CSA, thus the Communication Service Monitor acts as HOST to the Special Test.

For remote communications and uploading of variables and TMAC programs via RS-232, two separate Rear Panel RS-232 Connectors are utilized on the IFR-1900 CSA. The HOST utilizes the RS-232 Connector labeled **HOST**, and the Sp Tst employs the connector labeled **OPT**.
<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Syntax</td>
<td>2-1</td>
</tr>
<tr>
<td>2-2</td>
<td>Comments</td>
<td>2-3</td>
</tr>
<tr>
<td>2-3</td>
<td>Numeric Variables and Arrays</td>
<td>2-3</td>
</tr>
<tr>
<td>2-3-1</td>
<td>General</td>
<td>2-3</td>
</tr>
<tr>
<td>2-3-2</td>
<td>Saving Variables and Arrays (HOST Only)</td>
<td>2-4</td>
</tr>
<tr>
<td>2-4</td>
<td>Constants and Data Arrays</td>
<td>2-5</td>
</tr>
<tr>
<td>2-5</td>
<td>Numeric Notation and Formats</td>
<td>2-6</td>
</tr>
<tr>
<td>2-5-1</td>
<td>Numerical Notation</td>
<td>2-6</td>
</tr>
<tr>
<td>2-5-2</td>
<td>Numeric Formats</td>
<td>2-6</td>
</tr>
<tr>
<td>2-6</td>
<td>Mathematical Operators</td>
<td>2-7</td>
</tr>
<tr>
<td>2-6-1</td>
<td>Operators</td>
<td>2-7</td>
</tr>
<tr>
<td>2-6-2</td>
<td>Order of Calculation</td>
<td>2-8</td>
</tr>
<tr>
<td>2-7</td>
<td>Mathematical Functions</td>
<td>2-8</td>
</tr>
<tr>
<td>2-8</td>
<td>String Variables and Functions</td>
<td>2-9</td>
</tr>
<tr>
<td>2-9</td>
<td>Macros</td>
<td>2-10</td>
</tr>
<tr>
<td>2-10</td>
<td>Macro Decision Points</td>
<td>2-12</td>
</tr>
<tr>
<td>2-11</td>
<td>Variables and Arrays in Macros</td>
<td>2-14</td>
</tr>
<tr>
<td>2-12</td>
<td>Multitasking Macros</td>
<td>2-15</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>2-13</td>
<td>Display and Sound Control</td>
<td>2-18</td>
</tr>
<tr>
<td>2-13-1</td>
<td>Colors</td>
<td>2-18</td>
</tr>
<tr>
<td>2-13-2</td>
<td>Color Display</td>
<td>2-19</td>
</tr>
<tr>
<td>2-13-3</td>
<td>Windows</td>
<td>2-19</td>
</tr>
<tr>
<td>2-13-4</td>
<td>Soft Function Key Displays</td>
<td>2-20</td>
</tr>
<tr>
<td>2-13-5</td>
<td>Graphics and Text</td>
<td>2-21</td>
</tr>
<tr>
<td>2-13-6</td>
<td>Video Page Control (HOST Only)</td>
<td>2-22</td>
</tr>
<tr>
<td>2-13-7</td>
<td>Audio Tones</td>
<td>2-22</td>
</tr>
<tr>
<td>2-14</td>
<td>System Commands</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-1</td>
<td>System Key Commands</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-2</td>
<td>System Error Command</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-3</td>
<td>System Defaults Commands (HOST Only)</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-4</td>
<td>System Plot Commands (HOST Only)</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-5</td>
<td>System Time and Date Commands</td>
<td>2-23</td>
</tr>
<tr>
<td>2-14-6</td>
<td>System RS-232 Configure Commands</td>
<td>2-24</td>
</tr>
<tr>
<td>2-15</td>
<td>GPIB Operation</td>
<td>2-24</td>
</tr>
<tr>
<td>2-15-1</td>
<td>Operating Test Set Using GPIB Communication (Sp Tst Only)</td>
<td>2-24</td>
</tr>
<tr>
<td>2-15-2</td>
<td>Operating Test Set Using GPIB Communication (HOST Only)</td>
<td>2-25</td>
</tr>
<tr>
<td>2-15-3</td>
<td>Test Set Operating as GPIB Controller (HOST Only)</td>
<td>2-26</td>
</tr>
<tr>
<td>2-15-4</td>
<td>Test Set Operating as Talker Only (HOST Only)</td>
<td>2-26</td>
</tr>
<tr>
<td>2-15-5</td>
<td>Test Set Operating as Listener Only (HOST Only)</td>
<td>2-26</td>
</tr>
<tr>
<td>2-15-6</td>
<td>Test Set GPIB Print Control (HOST Only)</td>
<td>2-27</td>
</tr>
<tr>
<td>2-15-7</td>
<td>Test Set Frequency Control (HOST Only)</td>
<td>2-27</td>
</tr>
</tbody>
</table>
### SECTION 3 - GENERAL TMAC COMMANDS

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-16</td>
<td>Status Subsystem (HOST Only)</td>
<td>2-28</td>
</tr>
<tr>
<td>2-16-1</td>
<td>Status Registers</td>
<td>2-28</td>
</tr>
<tr>
<td>2-16-2</td>
<td>Status Byte</td>
<td>2-30</td>
</tr>
<tr>
<td>2-16-3</td>
<td>Standard Event Status Register</td>
<td>2-31</td>
</tr>
<tr>
<td>2-16-4</td>
<td>Operation Status Register</td>
<td>2-31</td>
</tr>
<tr>
<td>2-16-5</td>
<td>Operation Instrument Register</td>
<td>2-32</td>
</tr>
<tr>
<td>2-16-6</td>
<td>Questionable Status Register</td>
<td>2-32</td>
</tr>
<tr>
<td>2-16-7</td>
<td>Instrument Status Register</td>
<td>2-32</td>
</tr>
<tr>
<td>2-16-8</td>
<td>Instrument Summary Status Register</td>
<td>2-33</td>
</tr>
<tr>
<td>2-17</td>
<td>IEEE 488.2 Compliance Commands</td>
<td>2-34</td>
</tr>
<tr>
<td>2-18</td>
<td>General and Specific TMAC Commands</td>
<td>2-34</td>
</tr>
</tbody>
</table>

### SECTION 4 - CREATING AND UPLOADING TMAC PROGRAMS

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-1</td>
<td>General</td>
<td>4-1</td>
</tr>
<tr>
<td>4-2</td>
<td>Setup</td>
<td>4-2</td>
</tr>
<tr>
<td>4-2-1</td>
<td>RS-232 Terminal Emulation Program Setup</td>
<td>4-2</td>
</tr>
<tr>
<td>4-2-2</td>
<td>HOST Setup</td>
<td>4-3</td>
</tr>
<tr>
<td>4-2-3</td>
<td>Special Test Setup</td>
<td>4-3</td>
</tr>
<tr>
<td>4-3</td>
<td>Creating a TMAC Program</td>
<td>4-3</td>
</tr>
<tr>
<td>4-4</td>
<td>Uploading TMAC Program</td>
<td>4-4</td>
</tr>
<tr>
<td>4-5</td>
<td>Executing TMAC Program</td>
<td>4-4</td>
</tr>
<tr>
<td>4-6</td>
<td>Storing Uploaded TMAC Program into Flash Memory</td>
<td>4-6</td>
</tr>
</tbody>
</table>
### SECTION 5 - HOST SPECIFIC TMAC QUICK REFERENCE LIST

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-7</td>
<td>Executing TMAC Program Stored in Flash Memory</td>
<td>4-7</td>
</tr>
<tr>
<td>4-7-1</td>
<td>From RS-232 Terminal</td>
<td>4-7</td>
</tr>
<tr>
<td>4-7-2</td>
<td>From HOST Front Panel</td>
<td>4-7</td>
</tr>
<tr>
<td>4-8</td>
<td>Example TMAC Program with Multiple Macros</td>
<td>4-8</td>
</tr>
</tbody>
</table>

### SECTION 6 - HOST SPECIFIC TMAC COMMANDS

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-1</td>
<td>General</td>
<td>6-1</td>
</tr>
<tr>
<td>6-2</td>
<td>Setup Commands</td>
<td>6-1</td>
</tr>
<tr>
<td>6-3</td>
<td>Screen Commands</td>
<td>6-2</td>
</tr>
<tr>
<td>6-4</td>
<td>Instrument Command</td>
<td>6-5</td>
</tr>
<tr>
<td>6-5</td>
<td>RF Generator</td>
<td>6-6</td>
</tr>
<tr>
<td>6-5-1</td>
<td>RF Generator Commands</td>
<td>6-6</td>
</tr>
<tr>
<td>6-5-2</td>
<td>Remote RF Generator Examples</td>
<td>6-15</td>
</tr>
<tr>
<td>6-6</td>
<td>Receiver</td>
<td>6-19</td>
</tr>
<tr>
<td>6-6-1</td>
<td>Receiver Commands</td>
<td>6-19</td>
</tr>
<tr>
<td>6-6-2</td>
<td>Remote Receiver Examples</td>
<td>6-34</td>
</tr>
<tr>
<td>6-7</td>
<td>Duplex</td>
<td>6-40</td>
</tr>
<tr>
<td>6-7-1</td>
<td>Duplex General Commands</td>
<td>6-40</td>
</tr>
<tr>
<td>6-7-2</td>
<td>Duplex Transmitter Commands</td>
<td>6-41</td>
</tr>
<tr>
<td>6-7-3</td>
<td>Duplex Receiver Commands</td>
<td>6-50</td>
</tr>
<tr>
<td>6-7-4</td>
<td>Remote Duplex Examples</td>
<td>6-55</td>
</tr>
<tr>
<td>6-8</td>
<td>AF Generator</td>
<td>6-56</td>
</tr>
<tr>
<td>6-8-1</td>
<td>AF Generator Commands</td>
<td>6-56</td>
</tr>
<tr>
<td>6-8-2</td>
<td>Remote AF Generator Examples</td>
<td>6-65</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>6-9</td>
<td>Oscilloscope..................................................</td>
<td>6-66</td>
</tr>
<tr>
<td>6-9-1</td>
<td>Oscilloscope Commands .......................................</td>
<td>6-66</td>
</tr>
<tr>
<td>6-9-2</td>
<td>Remote Oscilloscope Examples .............................</td>
<td>6-76</td>
</tr>
<tr>
<td>6-10</td>
<td>Spectrum Analyzer ...........................................</td>
<td>6-79</td>
</tr>
<tr>
<td>6-10-1</td>
<td>Spectrum Analyzer Commands ..................................</td>
<td>6-79</td>
</tr>
<tr>
<td>6-10-2</td>
<td>Remote Spectrum Analyzer Example ..........................</td>
<td>6-91</td>
</tr>
<tr>
<td>6-11</td>
<td>Meter Commands ................................................</td>
<td>6-92</td>
</tr>
<tr>
<td>6-11-1</td>
<td>AF Meter Commands ............................................</td>
<td>6-92</td>
</tr>
<tr>
<td>6-11-2</td>
<td>Frequency Error Meter Commands ............................</td>
<td>6-96</td>
</tr>
<tr>
<td>6-11-3</td>
<td>Power Meter Commands ........................................</td>
<td>6-98</td>
</tr>
<tr>
<td>6-11-4</td>
<td>Deviation Meter (Peak) Commands ...........................</td>
<td>6-101</td>
</tr>
<tr>
<td>6-11-5</td>
<td>Modulation Meter Commands ...................................</td>
<td>6-104</td>
</tr>
<tr>
<td>6-11-6</td>
<td>Distortion Meter Commands ...................................</td>
<td>6-106</td>
</tr>
<tr>
<td>6-11-7</td>
<td>SINAD Meter Commands .........................................</td>
<td>6-109</td>
</tr>
<tr>
<td>6-11-8</td>
<td>Signal Strength Meter Commands ............................</td>
<td>6-112</td>
</tr>
<tr>
<td>6-11-9</td>
<td>Bit Error Rate (BER) Meter Commands ......................</td>
<td>6-113</td>
</tr>
<tr>
<td>6-11-10</td>
<td>Digital Multimeter Commands ................................</td>
<td>6-115</td>
</tr>
<tr>
<td>6-11-11</td>
<td>Phase Meter Commands .......................................</td>
<td>6-118</td>
</tr>
<tr>
<td>6-11-12</td>
<td>Deviation Meter (RMS) Commands ............................</td>
<td>6-120</td>
</tr>
<tr>
<td>6-11-13</td>
<td>Phase Meter (RMS) Commands ..................................</td>
<td>6-122</td>
</tr>
<tr>
<td>6-11-14</td>
<td>AF Level Meter Command ......................................</td>
<td>6-123</td>
</tr>
<tr>
<td>6-12</td>
<td>Accessory Commands ..........................................</td>
<td>6-123</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>6-13</td>
<td>Cellular AMPS/NAMPS</td>
<td>6-124</td>
</tr>
<tr>
<td>6-13-1</td>
<td>AMPS/NAMPS Cell Site Monitor Commands</td>
<td>6-125</td>
</tr>
<tr>
<td>6-13-2</td>
<td>AMPS/NAMPS Cellular Simulation Commands</td>
<td>6-132</td>
</tr>
<tr>
<td>6-13-3</td>
<td>Remote AMPS/NAMPS Cell Site Monitor Example</td>
<td>6-163</td>
</tr>
<tr>
<td>6-14</td>
<td>Program Commands</td>
<td>6-164</td>
</tr>
<tr>
<td>6-15</td>
<td>Flash Memory File Directory Operation</td>
<td>6-165</td>
</tr>
<tr>
<td>6-16</td>
<td>Generic Measure Commands</td>
<td>6-170</td>
</tr>
<tr>
<td>6-17</td>
<td>Initiate and Fetch Commands</td>
<td>6-172</td>
</tr>
<tr>
<td>6-18</td>
<td>Auxiliary (Special Test) Commands</td>
<td>6-173</td>
</tr>
</tbody>
</table>

**SECTION 7 - INTRODUCTION TO VOLUME II**

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-1</td>
<td>Organization of Volume II</td>
<td>7-1</td>
</tr>
<tr>
<td>7-2</td>
<td>Nomenclature Used in Volume II</td>
<td>7-1</td>
</tr>
</tbody>
</table>

**SECTION 8 - SPECIAL TEST SPECIFIC TMAC QUICK REFERENCE LIST**

**SECTION 9 - SPECIAL TEST SPECIFIC TMAC COMMANDS**

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9-1</td>
<td>General</td>
<td>9-1</td>
</tr>
<tr>
<td>9-2</td>
<td>HOST Commands</td>
<td>9-2</td>
</tr>
<tr>
<td>9-3</td>
<td>FREQ:BAND Command</td>
<td>9-3</td>
</tr>
<tr>
<td>9-4</td>
<td>Forward Control Channel (FOCC) Monitor Commands</td>
<td>9-4</td>
</tr>
<tr>
<td>9-4-1</td>
<td>Forward Control Channel Control</td>
<td>9-4</td>
</tr>
<tr>
<td>9-4-2</td>
<td>Forward Control Channel Decode Data</td>
<td>9-10</td>
</tr>
<tr>
<td>9-4-3</td>
<td>Forward Control Channel Raw Data</td>
<td>9-16</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>9-5</td>
<td>Forward Voice Channel (FVC) Monitor Commands</td>
<td>9-20</td>
</tr>
<tr>
<td>9-5-1</td>
<td>Forward Voice Channel Control</td>
<td>9-20</td>
</tr>
<tr>
<td>9-5-2</td>
<td>Forward Voice Channel Decode Data</td>
<td>9-22</td>
</tr>
<tr>
<td>9-5-3</td>
<td>Forward Voice Channel Raw Data</td>
<td>9-25</td>
</tr>
<tr>
<td>9-6</td>
<td>Forward Digital Traffic Channel (FDTC) Monitor Commands</td>
<td>9-26</td>
</tr>
<tr>
<td>9-6-1</td>
<td>Forward Digital Traffic Channel Control</td>
<td>9-26</td>
</tr>
<tr>
<td>9-6-2</td>
<td>Forward Digital Traffic Channel Decode Data</td>
<td>9-27</td>
</tr>
<tr>
<td>9-6-3</td>
<td>Forward Digital Traffic Channel Raw Data</td>
<td>9-39</td>
</tr>
<tr>
<td>9-6-4</td>
<td>IS-54 Raw Data</td>
<td>9-40</td>
</tr>
<tr>
<td>9-7</td>
<td>Reverse Control Channel (RECC) Monitor Commands</td>
<td>9-41</td>
</tr>
<tr>
<td>9-7-1</td>
<td>Reverse Control Channel Control</td>
<td>9-41</td>
</tr>
<tr>
<td>9-7-2</td>
<td>Reverse Control Channel Decode Data</td>
<td>9-42</td>
</tr>
<tr>
<td>9-8</td>
<td>Reverse Voice Channel (RVC) Monitor Commands</td>
<td>9-45</td>
</tr>
<tr>
<td>9-8-1</td>
<td>Reverse Voice Channel Control</td>
<td>9-45</td>
</tr>
<tr>
<td>9-8-2</td>
<td>Reverse Voice Channel Decode Data</td>
<td>9-46</td>
</tr>
<tr>
<td>9-9</td>
<td>Reverse Digital Traffic Channel (RDTC) Monitor Commands</td>
<td>9-47</td>
</tr>
<tr>
<td>9-9-1</td>
<td>Reverse Digital Traffic Channel Control</td>
<td>9-47</td>
</tr>
<tr>
<td>9-9-2</td>
<td>Reverse Digital Traffic Channel Decode Data</td>
<td>9-49</td>
</tr>
<tr>
<td>9-10</td>
<td>FDCCH Data Monitor</td>
<td>9-59</td>
</tr>
<tr>
<td>9-10-1</td>
<td>Setup Commands</td>
<td>9-59</td>
</tr>
<tr>
<td>9-10-2</td>
<td>Continuous Remote Raw Timeslot Data</td>
<td>9-60</td>
</tr>
<tr>
<td>9-10-3</td>
<td>Continuous Remote Raw Data</td>
<td>9-61</td>
</tr>
<tr>
<td>9-10-4</td>
<td>Buffered Raw Data</td>
<td>9-62</td>
</tr>
<tr>
<td>9-10-5</td>
<td>Layer 2 Data Monitor</td>
<td>9-63</td>
</tr>
<tr>
<td>9-10-6</td>
<td>FDCCH Real Time Data Monitor</td>
<td>9-71</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>9-11</td>
<td>RDCCH Data Monitor</td>
<td>9-133</td>
</tr>
<tr>
<td>9-11-1</td>
<td>Setup Commands</td>
<td>9-133</td>
</tr>
<tr>
<td>9-11-2</td>
<td>Continuous Remote Raw Timeslot Data</td>
<td>9-134</td>
</tr>
<tr>
<td>9-11-3</td>
<td>Continuous Remote Raw Data</td>
<td>9-135</td>
</tr>
<tr>
<td>9-11-4</td>
<td>Buffered Raw Data</td>
<td>9-136</td>
</tr>
<tr>
<td>9-11-5</td>
<td>Layer 2 Data Monitor</td>
<td>9-137</td>
</tr>
<tr>
<td>9-11-6</td>
<td>RDCCH Real Time Data Monitor</td>
<td>9-140</td>
</tr>
<tr>
<td>9-12</td>
<td>Cell Site Simulation Commands</td>
<td>9-156</td>
</tr>
<tr>
<td>9-12-1</td>
<td>General</td>
<td>9-156</td>
</tr>
<tr>
<td>9-12-2</td>
<td>FOCC Overhead Message Parameters</td>
<td>9-157</td>
</tr>
<tr>
<td>9-12-3</td>
<td>Calling Setup</td>
<td>9-166</td>
</tr>
<tr>
<td>9-12-4</td>
<td>Process and Handoff</td>
<td>9-168</td>
</tr>
<tr>
<td>9-12-5</td>
<td>Forward Voice Channel (FVC)</td>
<td>9-170</td>
</tr>
<tr>
<td>9-12-6</td>
<td>Forward Digital Traffic Channel (FDTC)</td>
<td>9-178</td>
</tr>
<tr>
<td>9-12-7</td>
<td>Talkback</td>
<td>9-206</td>
</tr>
<tr>
<td>9-12-8</td>
<td>Global Action Overhead Messages</td>
<td>9-206</td>
</tr>
<tr>
<td>9-12-9</td>
<td>Mobile Station Control Messages</td>
<td>9-212</td>
</tr>
<tr>
<td>9-12-10</td>
<td>Overhead Enable Commands</td>
<td>9-218</td>
</tr>
<tr>
<td>9-12-11</td>
<td>Superframe Setup</td>
<td>9-218</td>
</tr>
<tr>
<td>9-12-12</td>
<td>F-BCCH Commands</td>
<td>9-224</td>
</tr>
<tr>
<td>9-12-13</td>
<td>E-BCCH Commands</td>
<td>9-248</td>
</tr>
<tr>
<td>9-12-14</td>
<td>User-Defined Optional Message Types and Information Elements for the F-BCCH and E-BCCH</td>
<td>9-283</td>
</tr>
<tr>
<td>9-12-15</td>
<td>SPACH Commands</td>
<td>9-291</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>9-13</td>
<td>DCCH Mobile Station Simulation Commands</td>
<td>9-334</td>
</tr>
<tr>
<td>9-13-1</td>
<td>TDMA Transmission Overview</td>
<td>9-334</td>
</tr>
<tr>
<td>9-13-2</td>
<td>Setup Commands</td>
<td>9-338</td>
</tr>
<tr>
<td>9-13-3</td>
<td>RDCCH Raw Generator</td>
<td>9-340</td>
</tr>
<tr>
<td>9-13-4</td>
<td>Random Access Control Channel (RACH) Message Generator</td>
<td>9-343</td>
</tr>
<tr>
<td>9-13-5</td>
<td>RACH Layer 3 Message Builder</td>
<td>9-349</td>
</tr>
<tr>
<td>9-13-6</td>
<td>Reverse Digital Traffic Channel (RDTC) Mobile Simulation</td>
<td>9-387</td>
</tr>
<tr>
<td>9-13-7</td>
<td>Reverse Voice Channel (RVC) Mobile Simulation</td>
<td>9-388</td>
</tr>
<tr>
<td>9-14</td>
<td>BER Commands</td>
<td>9-389</td>
</tr>
<tr>
<td>9-15</td>
<td>Modulation Accuracy Commands</td>
<td>9-391</td>
</tr>
<tr>
<td>9-16</td>
<td>Generic Measure Commands</td>
<td>9-392</td>
</tr>
<tr>
<td>9-17</td>
<td>Flash Memory Commands</td>
<td>9-392</td>
</tr>
<tr>
<td>9-18</td>
<td>Miscellaneous Commands</td>
<td>9-394</td>
</tr>
<tr>
<td>9-19</td>
<td>TMAC Special Editing Commands</td>
<td>9-395</td>
</tr>
</tbody>
</table>

**SECTION 10 - SPECIAL TEST PROGRAM EXAMPLES**

<table>
<thead>
<tr>
<th>Paragraph</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-1</td>
<td>General</td>
<td>10-1</td>
</tr>
<tr>
<td>10-2</td>
<td>Forward Control Channel (FOCC)</td>
<td>10-1</td>
</tr>
<tr>
<td>10-2-1</td>
<td>Monitoring Decoded Data</td>
<td>10-1</td>
</tr>
<tr>
<td>10-2-2</td>
<td>Monitoring Raw Data</td>
<td>10-2</td>
</tr>
<tr>
<td>10-3</td>
<td>Forward Voice Channel (FVC)</td>
<td>10-3</td>
</tr>
<tr>
<td>10-3-1</td>
<td>Monitoring Decoded Data</td>
<td>10-3</td>
</tr>
<tr>
<td>10-3-2</td>
<td>Monitoring Raw Data</td>
<td>10-4</td>
</tr>
<tr>
<td>Paragraph</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>10-4</td>
<td>Forward Digital Traffic Channel (FDTC)</td>
<td>10-5</td>
</tr>
<tr>
<td>10-4-1</td>
<td>Monitoring Decoded Data</td>
<td>10-5</td>
</tr>
<tr>
<td>10-4-2</td>
<td>Monitoring Raw Data</td>
<td>10-6</td>
</tr>
<tr>
<td>10-4-3</td>
<td>Monitoring IS-54 Raw Data</td>
<td>10-7</td>
</tr>
<tr>
<td>10-5</td>
<td>Reverse Control Channel (RECC)</td>
<td>10-8</td>
</tr>
<tr>
<td>10-6</td>
<td>Reverse Voice Channel (RVC)</td>
<td>10-8</td>
</tr>
<tr>
<td>10-7</td>
<td>Reverse Digital Traffic Channel (RDTC)</td>
<td>10-9</td>
</tr>
<tr>
<td>10-8</td>
<td>Cell Site Simulation (CSS)</td>
<td>10-10</td>
</tr>
<tr>
<td>10-8-1</td>
<td>Overhead Message for FOCC</td>
<td>10-10</td>
</tr>
<tr>
<td>10-8-2</td>
<td>Overhead Message Using Primary and Secondary Cycles</td>
<td>10-11</td>
</tr>
<tr>
<td>10-8-3</td>
<td>Global Action Overhead Message</td>
<td>10-13</td>
</tr>
<tr>
<td>10-8-4</td>
<td>Mobile Station Control</td>
<td>10-13</td>
</tr>
<tr>
<td>10-8-5</td>
<td>Mobile Station Initiated Call</td>
<td>10-14</td>
</tr>
<tr>
<td>10-8-6</td>
<td>Base Station Initiated Call</td>
<td>10-15</td>
</tr>
<tr>
<td>10-8-7</td>
<td>Handoff</td>
<td>10-17</td>
</tr>
<tr>
<td>10-8-8</td>
<td>Page</td>
<td>10-19</td>
</tr>
<tr>
<td>10-9</td>
<td>Bit Error Rate (BER)</td>
<td>10-21</td>
</tr>
<tr>
<td>10-10</td>
<td>Modulation Accuracy (MODACC)</td>
<td>10-22</td>
</tr>
<tr>
<td>10-11</td>
<td>GPIB</td>
<td>10-23</td>
</tr>
<tr>
<td>10-12</td>
<td>Digital Control Channel (DCCH) Cell Site Simulation</td>
<td>10-25</td>
</tr>
</tbody>
</table>

SECTION 11 - IS-136 COMMAND REFERENCE

SECTION 12 - SPECIAL TEST KEY WORD INDEX
APPENDICES

Appendix | Title | Page
---------|-------|-----
A        | Predefined Macros and Constants | A-1
B        | Front Panel Keys and Keycodes   | B-1
C        | Individual Self Test Commands (HOST Only) | C-1
D        | Screen Characters               | D-1
E        | Abbreviations                    | E-1

INDEX

INDEX

LIST OF ILLUSTRATIONS

Figure No. | Title | Page
-----------|-------|-----
2-1        | Window Example                          | 2-19
2-2        | Soft Function Key Example               | 2-20
2-3        | Graphics Examples                        | 2-21
2-4        | ICON Example                             | 2-22
2-5        | HOST Status Register Description         | 2-28
2-6        | HOST Status Register Configuration       | 2-29
2-7        | HOST RQS Bit and SRQ Generation          | 2-30
3-1        | Box Example                              | 3-23
3-2        | KEYPAD:ERASE Example                     | 3-57
3-3        | KEYPAD:LABEL Example                     | 3-58
3-4        | KEYPAD:SOFT Example                      | 3-59
3-5        | RETURN Command                           | 3-71
Figure No. | Title | Page
---|---|---
4-1 | Operating IFR-1900 CSA Via RS-232 | 4-1
4-2 | Example TMAC Program | 4-5
4-3 | Macro Name Identification | 4-6
4-4 | Flash File and Macro Name Relationship | 4-6
9-1 | Mobile Station MSID | 9-139
9-2 | Examples of Primary and Secondary Cycles | 9-159
9-3 | Superframe Data Message | 9-219
9-4 | Normal Length Message in the RDCCH | 9-335
9-5 | Abbreviated Length Message in the RDCCH | 9-336
9-6 | Contiguous and Sub Channel Transmissions | 9-337
9-7 | User Data Message Diagram | 9-342
9-8 | RDCCH Message Diagram | 9-344
9-9 | RACH Message Transmission (Contiguous) | 9-345
9-10 | RACH Message Transmission (Sub Channel) | 9-346
10-1 | OMT Examples | 10-12
10-2 | Examples of Page and VCDES Message in OMTs | 10-20

**LIST OF TABLES**

Table No. | Title | Page
---|---|---
2-1 | Mathematical Operators | 2-7
2-2 | Colors and Color Selection Numbers | 2-18
2-3 | IEEE 488.2 Compliance Commands | 2-34
3-1 | Bitwise AND Operation Results | 3-5
3-2 | Bitwise XOR Operation Results | 3-6
3-3 | Bitwise OR Operation Results | 3-7
3-4 | AND Operation Results | 3-21
<table>
<thead>
<tr>
<th>Table No.</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-5</td>
<td>Ellipse Width to Height Equations</td>
<td>3-37</td>
</tr>
<tr>
<td>3-6</td>
<td>OR Operation Results</td>
<td>3-64</td>
</tr>
<tr>
<td>3-7</td>
<td>Character Rotation for ( n ) Values</td>
<td>3-73</td>
</tr>
<tr>
<td>3-8</td>
<td>Enable Register Preset Conditions</td>
<td>3-77</td>
</tr>
<tr>
<td>3-9</td>
<td>Video Page Settings</td>
<td>3-99</td>
</tr>
<tr>
<td>6-1</td>
<td>DMM Upper and Lower Limit Ranges and Units</td>
<td>6-116</td>
</tr>
<tr>
<td>6-2</td>
<td>Flash Memory Error Messages</td>
<td>6-169</td>
</tr>
<tr>
<td>9-1</td>
<td>Speech Parameter (27 bytes)</td>
<td>9-48</td>
</tr>
<tr>
<td>9-2</td>
<td>BER Bit Definition</td>
<td>9-49</td>
</tr>
<tr>
<td>9-3</td>
<td>Shared Channel Feedback Response</td>
<td>9-223</td>
</tr>
<tr>
<td>9-4</td>
<td>Flash Memory Error Messages</td>
<td>9-393</td>
</tr>
<tr>
<td>9-5</td>
<td>Parameters for Sp Tst Editing Commands</td>
<td>9-395</td>
</tr>
<tr>
<td>11-1</td>
<td>F-DCCH - DCCH Structure</td>
<td>11-4</td>
</tr>
<tr>
<td>11-2</td>
<td>F-BCCH - Access Parameters</td>
<td>11-5</td>
</tr>
<tr>
<td>11-3</td>
<td>F-BCCH - Control Channel Selection Parameters</td>
<td>11-6</td>
</tr>
<tr>
<td>11-4</td>
<td>F-BCCH - Registration Parameters</td>
<td>11-7</td>
</tr>
<tr>
<td>11-5</td>
<td>F-BCCH - System Identity</td>
<td>11-8</td>
</tr>
<tr>
<td>11-6</td>
<td>F-BCCH - BSMC Message Delivery</td>
<td>11-9</td>
</tr>
<tr>
<td>11-7</td>
<td>F-BCCH - Mobile Assisted Channel Allocation</td>
<td>11-9</td>
</tr>
<tr>
<td>11-8</td>
<td>F-BCCH - Overload Class</td>
<td>11-10</td>
</tr>
<tr>
<td>11-9</td>
<td>F-BCCH - Service Menu</td>
<td>11-10</td>
</tr>
<tr>
<td>11-10</td>
<td>F-BCCH - SOC/BSMC Identification</td>
<td>11-11</td>
</tr>
<tr>
<td>11-11</td>
<td>F-BCCH - SOC Message Delivery</td>
<td>11-11</td>
</tr>
<tr>
<td>11-12</td>
<td>F-BCCH - Mobile Assisted Channel Allocation (Multi Hyperband)</td>
<td>11-12</td>
</tr>
<tr>
<td>Table No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>11-13</td>
<td>E-BCCH - Neighbor Cell</td>
<td>11-13</td>
</tr>
<tr>
<td>11-14</td>
<td>E-BCCH - Regulatory Configuration</td>
<td>11-16</td>
</tr>
<tr>
<td>11-15</td>
<td>E-BCCH - BSMC Message Delivery</td>
<td>11-16</td>
</tr>
<tr>
<td>11-16</td>
<td>E-BCCH - Emergency Information Broadcast</td>
<td>11-17</td>
</tr>
<tr>
<td>11-17</td>
<td>E-BCCH - Mobile Assisted Channel Allocation</td>
<td>11-17</td>
</tr>
<tr>
<td>11-18</td>
<td>E-BCCH - Service Menu</td>
<td>11-18</td>
</tr>
<tr>
<td>11-19</td>
<td>E-BCCH - SOC/BSMC Identification</td>
<td>11-18</td>
</tr>
<tr>
<td>11-20</td>
<td>E-BCCH - SOC Message Delivery</td>
<td>11-19</td>
</tr>
<tr>
<td>11-21</td>
<td>E-BCCH - Time and Date</td>
<td>11-19</td>
</tr>
<tr>
<td>11-22</td>
<td>E-BCCH - Neighbor Service Info</td>
<td>11-20</td>
</tr>
<tr>
<td>11-23</td>
<td>E-BCCH - Alternate RCI Info</td>
<td>11-20</td>
</tr>
<tr>
<td>11-24</td>
<td>E-BCCH - Neighbor Cell (Multi Hyperband)</td>
<td>11-21</td>
</tr>
<tr>
<td>11-25</td>
<td>E-BCCH - Neighbor Service Info (Multi Hyperband)</td>
<td>11-25</td>
</tr>
<tr>
<td>11-26</td>
<td>E-BCCH - Mobile Assisted Channel Allocation (Multi Hyperband)</td>
<td>11-26</td>
</tr>
<tr>
<td>11-27</td>
<td>SPACH - Analog Voice Channel Designation</td>
<td>11-27</td>
</tr>
<tr>
<td>11-28</td>
<td>SPACH - Audit Order</td>
<td>11-28</td>
</tr>
<tr>
<td>11-29</td>
<td>SPACH - Base Station Challenge Order Confirmation</td>
<td>11-29</td>
</tr>
<tr>
<td>11-30</td>
<td>SPACH - BSMC Message Delivery</td>
<td>11-30</td>
</tr>
<tr>
<td>11-31</td>
<td>SPACH - Capability Request</td>
<td>11-31</td>
</tr>
<tr>
<td>11-32</td>
<td>SPACH - Digital Traffic Channel Designation</td>
<td>11-32</td>
</tr>
<tr>
<td>11-33</td>
<td>SPACH - Directed Retry</td>
<td>11-34</td>
</tr>
<tr>
<td>11-34</td>
<td>SPACH - Message Waiting</td>
<td>11-35</td>
</tr>
<tr>
<td>11-35</td>
<td>SPACH - Page</td>
<td>11-36</td>
</tr>
<tr>
<td>11-36</td>
<td>SPACH - Parameter Update</td>
<td>11-39</td>
</tr>
<tr>
<td>Table No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>11-37</td>
<td>SPACH - R-DATA</td>
<td>11-40</td>
</tr>
<tr>
<td>11-38</td>
<td>SPACH - R-DATA ACCEPT</td>
<td>11-43</td>
</tr>
<tr>
<td>11-39</td>
<td>SPACH - R-DATA REJECT</td>
<td>11-44</td>
</tr>
<tr>
<td>11-40</td>
<td>SPACH - Registration Accept</td>
<td>11-45</td>
</tr>
<tr>
<td>11-41</td>
<td>SPACH - Registration Reject</td>
<td>11-48</td>
</tr>
<tr>
<td>11-42</td>
<td>SPACH - Release</td>
<td>11-49</td>
</tr>
<tr>
<td>11-43</td>
<td>SPACH - Reorder/Intercept</td>
<td>11-50</td>
</tr>
<tr>
<td>11-44</td>
<td>SPACH - SOC Message Delivery</td>
<td>11-51</td>
</tr>
<tr>
<td>11-45</td>
<td>SPACH - SPACH Notification</td>
<td>11-52</td>
</tr>
<tr>
<td>11-46</td>
<td>SPACH - SSD Update Order</td>
<td>11-53</td>
</tr>
<tr>
<td>11-47</td>
<td>SPACH - Test Registration Response</td>
<td>11-54</td>
</tr>
<tr>
<td>11-48</td>
<td>SPACH - Unique Challenge Order</td>
<td>11-55</td>
</tr>
<tr>
<td>11-49</td>
<td>SPACH - User Alert</td>
<td>11-56</td>
</tr>
<tr>
<td>11-50</td>
<td>SPACH - Queue Disconnect Ack</td>
<td>11-57</td>
</tr>
<tr>
<td>11-51</td>
<td>SPACH - Queue Update</td>
<td>11-58</td>
</tr>
<tr>
<td>11-52</td>
<td>RACH - Audit Confirmation</td>
<td>11-59</td>
</tr>
<tr>
<td>11-53</td>
<td>RACH - Authentication</td>
<td>11-60</td>
</tr>
<tr>
<td>11-54</td>
<td>RACH - Base Station Challenge Order</td>
<td>11-60</td>
</tr>
<tr>
<td>11-55</td>
<td>RACH - BSMC Message Delivery</td>
<td>11-61</td>
</tr>
<tr>
<td>11-56</td>
<td>RACH - Capability Report</td>
<td>11-62</td>
</tr>
<tr>
<td>11-57</td>
<td>RACH - MACA Report</td>
<td>11-64</td>
</tr>
<tr>
<td>11-58</td>
<td>RACH - Origination</td>
<td>11-65</td>
</tr>
<tr>
<td>11-59</td>
<td>RACH - Page Response</td>
<td>11-68</td>
</tr>
<tr>
<td>11-60</td>
<td>RACH - R-DATA</td>
<td>11-70</td>
</tr>
<tr>
<td>Table No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>11-61</td>
<td>RACH - R-DATA ACCEPT</td>
<td>11-72</td>
</tr>
<tr>
<td>11-62</td>
<td>RACH - R-DATA REJECT</td>
<td>11-72</td>
</tr>
<tr>
<td>11-63</td>
<td>RACH - Registration</td>
<td>11-73</td>
</tr>
<tr>
<td>11-64</td>
<td>RACH - Serial Number</td>
<td>11-74</td>
</tr>
<tr>
<td>11-65</td>
<td>RACH - SOC Message Delivery</td>
<td>11-75</td>
</tr>
<tr>
<td>11-66</td>
<td>RACH - SPACH Confirmation</td>
<td>11-75</td>
</tr>
<tr>
<td>11-67</td>
<td>RACH - SSD Update Order Confirmation</td>
<td>11-76</td>
</tr>
<tr>
<td>11-68</td>
<td>RACH - Test Registration</td>
<td>11-76</td>
</tr>
<tr>
<td>11-69</td>
<td>RACH - Unique Challenge Order Confirmation</td>
<td>11-77</td>
</tr>
<tr>
<td>11-70</td>
<td>RACH - Queue Disconnect</td>
<td>11-77</td>
</tr>
</tbody>
</table>
SECTION 1 - INTRODUCTION

TMAC (Test Macro Language) was developed by IFR Systems Inc. to operate IFR test equipment in a remote configuration and allow internal versatile software to be created. TMAC is written so not just single measurements but entire batteries of tests may be performed. Combining this capability with the test functions available with the IFR-1900 CSA Test Set gives the user a great deal of flexibility in testing communication devices in an ATE environment. Additionally this language also provides a format to store and perform user defined test sequences. TMAC is based on the IEEE-488.2 format and conforms to the SCPI Standard. Some commands and operators, created before and not conforming to the SCPI Standard, are so noted.

TMAC is written on several layers of complexity. The first layer consists of the Machine Specific TMAC Commands. These commands are what actually set up and control the Test Set and take the measurements. The Specific TMAC commands for the IFR-1900 CSA encompass the complete operation of the Test Set. Every function of the IFR-1900 CSA can be performed remotely with the exception of power up of the Test Set.

The second level of TMAC is the framework that is used to combine the machine specific commands into a coherent and fluid process. Decision making commands allow different procedures to be performed under differing circumstances. These decisions may be based on complex expressions as well as simple boolean answers being returned from the Test Set. Loopy commands allow the continual monitoring of a parameter while performing tasks. Math and logic function commands together with bit manipulating commands allow the use and evaluation of complex functions. TMAC also provides a variety of data structures to assist in developing test procedures. Graphic commands allow the development of user-defined screens and menus to aid the user in performing test sequences. The Status System provided allows the monitoring of the Status Registers called for in the IEEE-488.2 Standard.

The third layer and one of the main strengths of TMAC is the capability to define macros with procedures built of machine specific commands as organized by the framework commands. Macros allow step by step test procedures to be developed, stored and locally or remotely implemented. Using macros is easy. Entering the name of the macro along with any variable parameters, executes the macro. Macros can be executed within other macros, allowing complicated procedures to be divided into smaller tasks. Macros can be initiated remotely, from the Front Panel Keyboard, or automatically upon power up (see Program Commands in Section 6) of the Test Set.

The final layer of TMAC is the ability to Multitask Macros. The Multitasking feature of TMAC allows several macros to alternate command execution at once giving the appearance of being executed simultaneously. Multitasking in TMAC allows the user to activate a task, pass to the next task, put a task to sleep, wake a sleeping task and stop a task.
SECTION 2 - TMAC (TEST MACRO LANGUAGE) OVERVIEW

2-1 SYNTAX

When working with TMAC, punctuation marks are used to establish a format of how commands and parameters are entered. The syntax needed to use TMAC is detailed below.

- Command Forms

Many commands can be entered using a short form or a long form. In this manual, the short form is shown in upper case, while the remainder of the long form is shown in lower case. Upper and lower case letters are used only in this manual to differentiate between the long and short form of commands. TMAC executes any valid command whether in upper and lower case letters or a combination of upper and lower case letters.

Given the command, STATUS:QUESTIONable:CONDition?, entries can be any of the following forms:

Example: status:questionable:condition? or
STATUS:QUESTIONABLE:CONDITION? or
STAT:QUES:COND? or
stat:ques:condition? or
STATUS:questionable:condition?

The following forms of the command are not allowed:

Example: STATU?:QUESTION:CON? or
status:questionable:condition? or
status:QUESTIONABLE:condition?

Each word in each command may have two acceptable forms (full or abbreviated). Entering a command using other than the two acceptable forms causes a syntax error.

- Colon (:

TMAC commands are arranged in hierarchical levels using the colon to separate the different levels. A colon before the beginning of a command (:FGEN) signifies the command starts at the first hierarchical level.

Example: FGEN:GEN1:SHAPE:SIN

FGEN precedes all Function Generator commands. The FGEN:GEN1 precedes all Function Generator commands dealing with Generator 1. FGEN:GEN1:SHAPE precedes all Function Generator commands selecting the waveform of Generator 1. FGEN:GEN1:SHAPE:SIN selects a sine wave for the waveform of Generator 1 of the Function Generator.

A new line character marks the end of a command and places the next command at the first command level.
Semicolon (;)

A semicolon signifies the end of a command and starts the next command at the last level of the previous command.

Example: \texttt{FGEN:GEN1:FREQ 2000;SHAPE:SIN;} :SCREEN:FUNC

The first command, \texttt{FGEN:GEN1:FREQ 2000}, leaves the TMAC Compiler at the \texttt{FGEN:GEN1} level. Preceded by a semicolon, the next command, \texttt{SHAPE:SIN}, starts from this level. The next semicolon leaves the TMAC Compiler at the \texttt{FGEN:GEN1:SHAPE} level. The following colon resets the Compiler to the first hierarchical level to start the next command, \texttt{SCREEN:FUNC}.

Example: \texttt{DUP:INP:FREQ?;MOD:USER:MOD:DATA;FILTER 30;:SCREEN:DUP}

This command line performs the following commands:

\texttt{DUP:INP:FREQ?}
\texttt{DUP:INP:MOD:USER:MOD:DATA}
\texttt{DUP:INP:MOD:USER:FILTER 30}
\texttt{SCREEN:DUP}

Whitespace

At least one space is needed between commands and parameters. Other spaces or lack of them between items do not affect command execution. Spaces inside command words or between command word and ending punctuation (\texttt{ME AS:FREQ?} or \texttt{MEAS:FREQ ?}) are not allowed.

Comma (,)

Commas are used between parameters when more than one is listed and to continue printing on same line.

Example: \texttt{MACRO 1,2,3,4}

\begin{verbatim}
PRINT "This gets printed ". // Comma causes next printing to begin at end of line, not the next line down.
PRINT "on one line."
\end{verbatim}

Back slash (\)

A back slash allows a command to be continued on the next line. The back slash cannot break a command word.

Example: \texttt{FGEN:GEN1:\nFREQ 2000}

This is the same command as \texttt{FGEN:GEN1:FREQ 2000}.

Example: \texttt{FGEN:GE\nN1:FREQ 2000}

This command fails to execute and produces an error message because command word \texttt{GEN1} is broken.
2-2 COMMENTS
Comments provide clarity in macros and are ignored by the TMAC compiler/interpreter. There are two types of comments.

Characters /* cause the rest of line to be a comment.
Example: PRINT COS(freq*SQR(2))  // PRINTS COS OF FREQ X SQUARE ROOT OF 2  
  The message PRINTS COS OF FREQ X SQUARE ROOT OF 2 is not executed by the TMAC compiler/interpreter.  
Characters */ start and end the second type of comment. It can contain several lines.
Example: PRINT COS(freq*SQR(2))  /* prints cosine of the quantity RF Generator  
  frequency multiplied by the square root of 2 */  
  The TMAC compiler/interpreter ignores everything between the comment characters.

2-3 NUMERIC VARIABLES AND ARRAYS
2-3-1 GENERAL
Variables and arrays are used to hold numeric values and can be local or global. If declared inside a macro, a variable or array is local and is used inside that macro only. If declared outside a macro, a variable or array is global and is used outside or inside any macro (unless a local variable has the same name as the global variable). Locality takes precedence.

The number of variables and arrays allowed is limited by the amount of memory available. A maximum of 998 memory locations are available for global variables. A global variable takes one location, global arrays take a location for each array element and global string variables take 9 locations each. Local variables are also limited by memory space which varies depending on many factors including call nesting (see Macros in 2-9). One to two hundred local variables are usually available with arrays taking the space of a variable for each array element and string variables taking nine times the space needed for a variable.

The first character of a variable or array name must be a letter while the remaining characters can be letters, digits 0 through 9 and the underscore. The maximum length is 31 characters.

Variables A through Z are pre-declared and global and cannot be declared as a string variable. Variables A through Z are also referred to as Free variables.

To declare a variable, use the VAR command:
Example:  VAR Freq, Rad, Mod

Freq, Rad and Mod are now declared variables.

Variables can be initialized when they are declared by including an equal sign and the value. The value of numeric variables are equal to 0 when declared unless initialized by the user.
Example:  VAR Rad = 4.32876, Mod

Rad is declared and assigned a value of 4.32876. Mod is declared and contains a value of 0.

2-3
Arrays are also declared using the VAR command. The highest index is included in brackets. The index count starts with 0, making index one less than the total number of variables in the array.

Example: \texttt{VAR BB[9]}

This command declares array BB with 10 variables: BB[0] through BB[9].

Braces are used to initialize arrays. More than one can be initialized at a time.

Examples:
\begin{verbatim}
VAR BUNCH[10] = (1,2,3,4,5,6,7,8,9,10,11)
\end{verbatim}

A variable and an array cannot have the same name at the same time. Accessing an array without specifying an index yields erroneous results.

\section*{2-3-2 SAVING VARIABLES AND ARRAYS (HOST ONLY)}

Refer to Section 9-2 to run HOST commands when operating as the Sp Tst.

Global variable and array values are lost when the Test Set power is turned off. To save variable or array values in non-volatile memory, use the \texttt{NVSAV} command. 283 non-volatile memory locations are available. To recall a variable, use the \texttt{NVRCL} command:

Each variable uses one memory location and arrays use a memory location for each value. Free variables, local variables and local arrays cannot be saved in non-volatile memory.

Example:
\begin{verbatim}
VAR DD
NVSAV DD,10
\end{verbatim}

DD is now saved in non-volatile memory location 10. To recall DD, use the NVRCL command.

Example:
\begin{verbatim}
NVRCL DD,10
\end{verbatim}

The content of memory location 10 is loaded into variable DD.

When an array is saved using the \texttt{NVSAV} command, the whole array is saved in consecutive memory locations.

Example:
\begin{verbatim}
VAR list[5]
NVSAV list,1
\end{verbatim}

The values of list are saved in memory locations 1 through 6 (6 values starting at 1). Arrays are recalled the same as variables.

Example:
\begin{verbatim}
NVRCL list,1
\end{verbatim}

The contents of memory locations 1 through 6 are loaded into list[0] through list[5].

\textbf{Saving to previously used memory locations overwrites previous entries.}
Example: \[ \text{VAR List}[4] = \{2, 4, 6, 8, 10\} \] // Declares an array named List.

\[ \text{VAR Group}[2] = \{23, 46, 69\} \] // Declares an array named Group.

\[ \text{NVSAV List, 10} \] // Saves List in memory locations 10 through 14.

\[ \text{NVSAV Group, 12} \] // Saves Group in memory locations 12 through 14.

\[ \text{NVRCL List, 10} \] // Recalls memory locations 10 through 14 and assigns the values to List.

Three of the saved values of List were overwritten when the same memory locations were used to save the array Group.

2-4 CONSTANTS AND DATA ARRAYS

Constants are declared using the `CONST` command. Constants are global only and cannot be declared inside a macro. Constants can only be assigned numeric data.

Example: \[ \text{CONST PI, 3.1415, FREQ, 3000} \]

An expression can be assigned to a constant. Expressions are evaluated prior to being assigned.

Example: \[ \text{CONST AA, (4+6)/2} \] // Assigns 5 to AA.

After being declared, a constant can be changed only with another `CONST` command.

Example: \[ \text{CONST AA, 36} \]

\[ \text{AA} = 20 \]

/* The second command results in an error message and is not performed, leaving AA equal to 36. */

Constant data arrays are declared using the `DATA` command.

Example: \[ \text{DATA offsets_1 = \{2, 3, 4, 5, 6, 7, 8, 9\}} \]

Like a constant, data arrays can only be assigned numeric data and an expression can be assigned to a constant. Expressions are evaluated prior to being assigned. After being declared, a data array can be changed only by another `DATA` command. Contents of data arrays are accessed using the index enclosed in brackets, where the first element of the data array has an index value of zero.

Example: \[ \text{DATA Points = \{11.6, 17.3, 22.4, 27.1\}} \]

\[ \text{Y} = \text{Points}[2] \]

Variable Y is set equal to 22.4.

The `IDATA` command is the same as `DATA` except uses integer values only.
2-5 NUMERIC NOTATION AND FORMATS

2-5-1 NUMERICAL NOTATION

Numerals are expressed as fixed point decimal, scientific notation or as an integer. Numerals are specified as negative with a – character as shown:

Example:  -51

This denotes a negative 51.

The maximum fixed point decimal numeral that can be entered is ±2147483647. Scientific notation uses the letter e to denote an exponent of 10.

Example:  54e7

This number signifies 54 multiplied by 10 to the 7th power or 540,000,000.

100 is the maximum power of 10 that can be entered, although the maximum number in scientific form allowed is ±1.797693e308. Numbers with a larger power of 10 than 100 are assigned to a variable by multiplying powers of 10.

Example:  Large = 54e7*1e100*1e100

This example assigns 54e207 or 54 multiplied by 10 to the 207th power to variable Large.

For execution reasons, the Sp Tst uses single precision floating point numbers (e.g., 1e±38).

2-5-2 NUMERIC FORMATS

To enter data in a different format (base 2, 8 or 16), precede data with one of the following characters:

#B   Binary
#Q   Octal
#H   Hexadecimal

Data format for the returned data for queries is changed to base 2, 8, 10 or 16 using the FORMat command. The default format is base 10.

Binary, octal or hexadecimal data received from the Test Set (HOST only) is preceded by #B, #Q or #H notation. Data in scientific notation is unaffected by any of the FORMat commands.
2-6 MATHEMATICAL OPERATORS

TMAC provides a variety of mathematical operators and bit manipulating functions to perform mathematical operations. A 32 bit word is used for binary and bitwise operations.

2-6-1 OPERATORS

Table 2-1 lists the mathematical operators used in TMAC.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Assignment operator</td>
</tr>
<tr>
<td>**</td>
<td>Exponential operator (does not conform to SCPI Standard)</td>
</tr>
<tr>
<td>*</td>
<td>Multiplication operator</td>
</tr>
<tr>
<td>/</td>
<td>Division operator</td>
</tr>
<tr>
<td>+</td>
<td>Addition operator and positive unary operator</td>
</tr>
<tr>
<td>-</td>
<td>Subtraction operator and negative unary operator</td>
</tr>
<tr>
<td>%</td>
<td>Modulo operator (does not conform to SCPI Standard)</td>
</tr>
<tr>
<td>++</td>
<td>Increment operator</td>
</tr>
<tr>
<td>--</td>
<td>Decrement operator</td>
</tr>
<tr>
<td>~</td>
<td>Bitwise complement</td>
</tr>
<tr>
<td>!</td>
<td>Logical negation (NOT) unary operator</td>
</tr>
<tr>
<td>;</td>
<td>Bitwise OR</td>
</tr>
<tr>
<td>&amp;</td>
<td>Bitwise AND</td>
</tr>
<tr>
<td>^</td>
<td>Bitwise XOR (conflicts with SCPI Standard for Exponential operator)</td>
</tr>
<tr>
<td>&lt;&lt;=</td>
<td>Shift left operator</td>
</tr>
<tr>
<td>&gt;&gt;=</td>
<td>Shift right operator</td>
</tr>
</tbody>
</table>

Table 2-1 Mathematical Operators
2-6-2 ORDER OF CALCULATION

When more than one operator is contained in an expression, the operators are calculated in the following order:

- Contents of parenthesis, calculating from inner sets to outer sets of parenthesis.
- Positive (+), negative (-), bitwise complement (~) and logical negation (!).
- Exponentiation (**).
- Multiplication (*), division (/) and modulus (%).
- Addition and subtraction.
- Shift left (<<) and right (>>) operations.
- Bitwise AND (\&).
- Bitwise XOR (^).
- Bitwise OR (|).
- Logical AND.
- Logical OR.
- Condition? true:false. (Refer to IF ELSE (Shorthand) command in section 3.)

2-7 MATHEMATICAL FUNCTIONS

TMAC provides the following mathematical functions to assist in mathematical calculations:

<table>
<thead>
<tr>
<th>TRUE,ON</th>
<th>FALSE,OFF</th>
<th>RND</th>
<th>RAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOG</td>
<td>LN</td>
<td>ABS</td>
<td>FLOOR</td>
</tr>
<tr>
<td>SQR</td>
<td>EXP</td>
<td>SIN</td>
<td>COS</td>
</tr>
<tr>
<td>SIGN</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See Section 3 for a description of each command.
2-8 STRING VARIABLES AND FUNCTIONS

A string is a segment of text consisting of ASCII characters terminated by a null character. String variables are declared using the STRING command. There is a free string variable, represented by the $ character that is pre-declared and global. ($) cannot be declared a numeric variable.) String variables are either local (declared inside a macro) and are used only inside that macro or are global (declared outside a macro) and are used outside or inside any macro, depending on scope. 128 bytes (1 byte per ASCII character) of system memory is allocated for each string variable declared.

A literal string (actual ASCII text segment included in macro) is enclosed in quotation marks or apostrophes. To signify a quotation mark inside a literal string enclosed with quotation marks, a pair of quotation marks are used. When quotation marks are used to signify a literal string, apostrophes are treated as normal characters. To signify an apostrophe inside a literal string enclosed in apostrophes, a pair of apostrophes are used. When apostrophes are used to signify a literal string, quotation marks are treated as normal characters.

String variables cannot be initialized when declared. After being declared literal strings and/or other declared string variables can be assigned to a string variable.

Example: STRING WARNING, Note1, Message_no_2
          Note1 = "This is a string variable"

Strings and string variables are combined (concatenated) by using + character as follows:

Example: STRING Note_1, Note_2, Note_3
          Note_1 = "A STRING"
          Note_2 = "COMBINING"
          Note_3 = Note_1 + " " + Note_2 + " EXAMPLE"

Note_3 now equals A STRING COMBINING EXAMPLE.

Segments are extracted from strings using brackets in the following manner:

    name1 = name2[start][end]

name1 is set equal to the segment in name2 that begins with the start number and ends with the end number. Count of string name2 begins with 0. String name2 is not affected unless name1 and name2 are the same name.

Example: STRING Test
          $ = "Test1"
          Test = $[4][4]

          Test now equals 1, the fourth (count begins with 0) character of string $.

Example: STRING MEMO1
          MEMO1 = 'message .example'
          MEMO1 = MEMO1[10][15]+MEMO1[0][3]

          MEMO1 now contains the following string:

          ample mess

Word of caution: when accessing data in a string, invalid data beyond the terminating null character is accessible, because no bounds checking is performed.
String Arrays are also declared using the **STRING** command by including the highest index in brackets.

The index count starts with 0, making index one less than the actual number of variables in the array. Like string variables, string arrays cannot be initialized when declared. Only one string array element is assigned a value at a time.

**Example:**

```
STRING MESSAGE[5]
```

This command declares a string array named message containing six elements.

**Example:**

```
MESSAGE[0] = "see"
MESSAGE[1] = "spot"
MESSAGE[2] = "run"
```

This example assigns values to the first three elements of string array MESSAGE.

128 bytes (1 byte per ASCII character) of system memory is allocated for each element of a string array declared. (Sp Tst only: A maximum length for each element of the string array may be specified when the string array is declared. See **STRING** command in Section 3 for details.)

Substringing on string arrays allows segment selection from one or more elements in a string array. Adding a **start** and **end** number selects the segment in the **element** of the declared and initialized string array (**arrayname**). Segments are selected using the following format:

```
arrayname[element][start][end]
```

**Example:**

```
$ = MESSAGE[0][1][2]
```

This example assigns the string "ee" to $. **ee** is the segment in the first element [0], starting at character [1] and going through character [2].

The following string functions allow manipulations of strings:

<table>
<thead>
<tr>
<th>CHR</th>
<th>TAB</th>
<th>ASC</th>
<th>LEN</th>
<th>STRPOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>STR</td>
<td>VAL</td>
<td>PIXLEN</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See Section 3 for a description of each command.

**2-9 MACROS**

Macros are groups of commands arranged to accomplish one or more tasks and may be considered programs; however, several macros may be grouped together (along with constants and global variables) into a single program. Macros are written and transferred to the Test Set by a Host System (PC with terminal emulating software). Once the macro is transferred to the Test Set, the macro is stored in memory (non-volatile for HOST) until deleted by a **PMC** or **FORGET** command or power is removed (Sp Tst only). Once in memory, macros can be transferred to Flash Memory (see Section 4).

Macros are defined using the **DMC** command (complying with IEEE-488.2) and have the form:

```
*DMC "name", command,... , command
```

**Example:**

```
*DMC "Square", Y=X**2;PRINT "Y holds X squared"
```
Once loaded into memory, a macro is executed by entering the macro name. If the above macro is loaded, entering Square executes the macro Square. Macros can be called out and executed from other macros. However, a macro can only be transferred into memory if all macros called out are defined macros, unless the name of any undefined macro is in an `INTERP` command.

Example: 
```
*DMC 'Test_1',BEGIN
  Average
END
```

Example: 
```
*DMC 'Test_2',BEGIN
  INTERP 'Total'
END
```

The macro `Test_1` can be transferred into memory only if the macro `Average` is already defined. Macro `Test_2` can be loaded into memory because the string `Total` is evaluated after execution of the `INTERP` command. The macro `Total` has to be defined prior to `Test_2` execution, otherwise an execution error occurs.

Call nesting is the extended process of calling out other macros that call out other macros that call out other macros, etc. Call nesting uses a limited stack to contain all the internal variables and arrays for each macro called out. Reaching the stack limit initiates a memory error.

With TMAC, as opposed to IEEE-488.2, macros are not limited to one line. Using the `BEGIN` and `END` commands lets the macro contain multiple lines, allowing large involved macros, while also providing clarity lacking with one line macros.

Example: 
```
*DMC 'Square_x',BEGIN
  Y=X**2
END
```

A macro is deleted from memory using the `FORGET` command. All macros and variables declared after the deleted macro was declared, are also deleted (because they may contain references to the deleted macro). The `PMC` command deletes all macros and declared variables in memory regardless of location. Predefined macros, listed in Appendix A, are permanently in memory and are unaffected by the `PMC` or `FORGET` commands. The `LMC?` query (HOST only) returns a list of the macros and defined variables contained in memory (e.g., `PRINT *LMC?`).

The address of a macro (location of macro in memory) is assigned to a variable by using an ampersand in the following manner:

\[ X=\&name \]

The `EXEC` command executes the macro at the address following the command.

Example: 
```
ADD = &Square_x
EXEC ADD
```

The first command loads the address of `Square_x` into `ADD`. The `EXEC` command then executes `Square_x`.

The `EXEC` command can be used to call out undefined macros by simply declaring a variable and using the variable as the address.

The `WAI` command pauses command execution until all previous operations are complete. `WAI` is used following `SCREEN` commands and commands involving routing changes or long execution times.
The amount of available memory is the only limit to the number of macros that can be loaded. The ROOM command provides the amount of currently available memory space in bytes.

Example: PRINT ROOM
          Prints the number of available bytes left in memory on the color display (HOST) or
          on the RS-232 terminal (Sp Tst).

Refer to 6-13 and 6-14 for HOST specific PROGram and MMEM commands used in executing
HOST specific macros from the Front Panel.

Refer to Section 4 for loading Sp Tst macros into memory for Front Panel execution.

2-10 MACRO DECISION POINTS

TMAC allows decision points to be placed inside a macro to allow different courses of action.
For instance, if a condition is true, then one course of action is taken. If the condition is false,
then another course of action is taken. See CASE, IF, IF ELSE and IF ELIF (see Section 3).
Generally, a condition has one of the following relational operators:

=         Equal          !=        Not equal
<         Less than      >         Greater than
<=        Less than or equal  >=       Greater than or equal

Example: IF x=6
          PRINT y
          ENDF

Condition x equals 6 must be true for the next command (PRINT y) to execute;
otherwise, command execution passes to the command following the IF ENDF
command, ignoring the print command.

Example: IF x!=z
          PRINT y
          ENDF

Condition x != z (x not equal to z) must be true for commands contained in the
IF ENDF command to execute.
Conditions may also contain logical operators **AND** and **OR**. These operators connect two conditions. An expression with an **OR** is true if one or both conditions are true. An expression with an **AND** is true only if both conditions connected by the operator are true. Many logical operators are possible in a condition.

Example:  
```
IF x=6 OR x!=z
  PRINT y
ENDIF
```

The above condition is true if either x is equal to 6 or if x is not equal to z or if both are true.

Example:  
```
IF x=6 AND x!=z
  PRINT y
ENDIF
```

This condition is true if both x is equal to 6 and x is not equal to z.

Example:  
```
IF W=3 OR U< 8 AND V=4
  PRINT y
ENDIF
```

This condition is true if either W is equal to 3 or both U is less than 8 and V is equal to 4.

Mathematical operators can be used in conditions.

Example:  
```
IF W> 2*X**3
```

The quantity $2 * \text{x}^3$ is calculated before compared to W.

Example:  
```
IF T = U&++V
```

In this example, V is incremented and bitwise AND is performed before comparison with T is performed.

Mathematical functions can also be used in conditions.

Example:  
```
IF Y = SQR(COS(2*Freq))
```

In this example, the square root of the cosine of the quantity of 2 multiplied by Freq is calculated before the quantity is compared to Y.

Conditional expressions can be used without **IF** or other decision commands. Conditional expressions alone return a 1 if true and a 0 if false.

Example:  
```
X=W<2*Y**2
```

If W is less than $2y^2$, X is set to 1. If W is greater than $2y^2$, X is set to 0.

Example:  
```
X=W=ABS(W)
```

If W is positive, W equals ABS(W) and X is set to 1. If W is negative, W does not equal ABS(W) and X is set to 0.
2-11 VARIABLES AND ARRAYS IN MACROS

Variables and arrays are passed to a macro when executed by entering variables and arrays (parameters) after the name of the macro. Inside the macro, characters $1, $2, $3, ..., $9 are placeholders for the passed parameters. Parameters may be passed to macros during execution.

Example: 
*DMC "Compute",Y=$1**$2;PRINT $1," TO THE POWER OF","$2," IS ",Y

Entering Compute 4,3 executes the macro Compute and sets $1 equal to 4 and $2 equal to 3. The displayed result: 4 TO THE POWER OF 3 IS 64.

Example: 
VAR LIST[2]=(1,3,5)

*DMC "Average",BEGIN
VAR SUM=0
FOR N=0 TO $1-1
SUM=SUM + $2[N]
NEXT N
PRINT "THE AVERAGE IS ",SUM/$1
END
Average 3, LIST

The macro Average reads an array and finds the average of the elements of the array. The length and name of the array is passed upon macro execution. When Average 3, LIST is entered, array LIST is passed using a FOR loop. The macro then prints the average of the 3 array values.

Example: 
Add=&Compute
EXEC Add, 4, 3

A comma is placed before the variables when a macro is executed using the EXEC command. This example executes the macro Compute and has the same result as the first example.

The INPUT command allows data entry during macro execution. Data is entered using the Test Set DATA ENTRY Keypad (HOST) or the RS-232 terminal keyboard (Sp Tst). For the HOST, the INPUT command causes a blinking cursor to be displayed. The EDIT:WIDTH command limits the width of the blinking cursor to n number of pixels.

Example: 
*DMC "Compute",BEGIN
EDIT:COLOR:MENU 4 // Sets Input background color to Red.
EDIT:WIDTH 32 // Sets cursor width to 32 pixels.
INPUT X // Displays a blinking cursor and sets X to entered value.

Y=$1**X
PRINT $1," TO THE POWER OF ",X," IS ",Y
END

If Compute 5 is entered to execute the macro and 3 is entered using the DATA ENTRY Keypad (HOST), 5 TO THE POWER OF 3 IS 125 is printed.
## 2-12 MULTITASKING MACROS

Multitasking allows command execution to alternate between different macros prior to their completion. This allows complex macros to be separated into more manageable macros. Before sharing execution time with other macros, a macro must: 1) Be declared a task using the `TASK` command; and 2) Be put into the schedule queue using the `ACTIVATE` command.

Multitasking requires the first macro in the schedule queue also be the last macro to finish execution.

See Section 3 for a description of each command.

The following commands provide multitasking control:

<table>
<thead>
<tr>
<th>TASK</th>
<th>ACTIVATE</th>
<th>TPAUSE</th>
<th>TSTOP</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLEEP</td>
<td>KILL</td>
<td>WAKE</td>
<td></td>
</tr>
</tbody>
</table>

Example:

```plaintext
*DMC "Main",BEGIN
ACTIVATE "First"
ACTIVATE "Second"
A = 0
B = 0
WHILE A = 0 OR B = 0
  TPAUSE
WEND
END

*DMC "First",BEGIN
FOR x = 1 to 10
  PRINT "FIRST"
  IF x = 3
    A = 1
    TSTOP
  ENDIF
  TPAUSE
NEXT x
A = 1
END

*DMC "Second",BEGIN
n = 1
FOR m = 1 to 10
  PRINT "SECOND"
  IF n = 2
    TPAUSE
    n = 0
  ENDIF
  n = n + 1
NEXT m
B = 1
END

TASK "Second"  // These TASK commands are placed after macros
TASK "First"   // First and Second, because each macro must be
               // defined before being declared a task.
```
As the above list of commands are uploaded into memory (see Section 4), macros Main, First and Second are defined, followed by the two TASK commands. The TASK command declares a defined macro as a task. If a TASK command is encountered that declares an undefined macro, an error occurs.

Executing the macro Main places Main in the schedule queue and, then using the ACTIVATE command, loads tasks First and Second into the schedule queue with Main. A WHILE loop assures Main executes until First and Second have finished execution. The following sequence occurs:

- Macro execution enters WHILE loop, because conditions are met. The TPAUSE command is encountered which causes the execution of Main to pause and pass execution on to the next macro in the schedule queue. The next macro in the schedule queue is First.

- First begins execution, x is set to 1 and FIRST is printed. The IF ENDIF command is skipped (x ≠ 3), the TPAUSE command executes and command execution passes to Second.

- Second begins execution, n and m are set to 1 and SECOND is printed. The IF ENDIF command is skipped (n ≠ 2) and n is incremented to 2. m is incremented to 2 and command execution loops to top of the FOR loop. SECOND is printed and the TPAUSE command inside the IF ENDIF command executes passing command execution to Main.

- Command execution begins again after the TPAUSE command and returns to the beginning of the WHILE loop. A and B are compared and the TPAUSE command is executed, passing command execution to First.

- In First, command execution begins again after the TPAUSE command. x is incremented to 2, command execution loops to the top of the FOR loop and FIRST is printed. The IF ENDIF command is skipped (x ≠ 3) and the TPAUSE command passes command execution to Second.

- In Second, command execution begins again after the TPAUSE command. n is set to 0 and then incremented to 1. m is incremented to 3 and command execution loops to the top of the FOR loop. SECOND is printed, the IF ENDIF command is skipped (n ≠ 2) and n is incremented to 2. m is incremented to 4 and command execution loops to the top of the FOR loop. SECOND is printed and the TPAUSE command inside the FOR loop executes passing command execution to Main.

- Command execution begins again after the TPAUSE command and returns to the beginning of the WHILE loop. A and B are compared and the TPAUSE command is executed, passing command execution to First.

- In First, command execution begins again after the TPAUSE command. x is incremented to 3, command execution loops to the top of the FOR loop and FIRST is printed. A is set to 1 and the TSTOP command executes taking First out of the schedule queue. Command execution passes to Second.

- In Second, command execution begins again after the TPAUSE command. n is set to 0 and then incremented to 1. m is incremented to 5, command execution loops to the top of the FOR loop and SECOND is printed. The IF ENDIF command is skipped (n ≠ 2) and n is incremented to 2. m is incremented to 6, command execution loops to the top of the FOR loop and SECOND is printed. The TPAUSE command is executed, passing command execution to Main.
• Command execution begins again after the TPAUSE command and returns to the beginning of the WHILE loop. A and B are compared and the TPAUSE command is executed, passing command execution to Second (Main and Second are the only remaining macros in the schedule queue).

• In Second, command execution begin again after the TPAUSE command. n is set to 0 and then incremented to 1. m is increment to 7, command execution loops to the top of the FOR loop and SECOND is printed. The IF ENDIF command is skipped (n ≠ 2) and n is incremented to 2. m is incremented to 8, command execution loops to the top of the FOR loop and SECOND is printed. The TPAUSE command is executed, passing command execution to Main.

• Command execution begins again after the TPAUSE command and returns to the beginning of the WHILE loop. A and B are compared and the TPAUSE command is executed, passing command execution to Second.

• In Second, command execution begin again after the TPAUSE command. n is set to 0 and then incremented to 1. m is increment to 9, command execution loops to the top of the FOR loop and SECOND is printed. The IF ENDIF command is skipped (n ≠ 2) and n is incremented to 2. m is incremented to 10, command execution loops to the top of the FOR loop and SECOND is printed. The TPAUSE command is executed, passing command execution to Main.

• Command execution begins again after the TPAUSE command and returns to the beginning of the WHILE loop. A and B are compared and the TPAUSE command is executed, passing command execution to Second.

• In Second, command execution begin again after the TPAUSE command. n is set to 0 and then incremented to 1. m is increment to 11, command execution leaves the FOR loop. B is set to 1, and macro Second finishes execution leaving only Main in schedule queue. Command execution is returned to Main.

• The conditional expression of the WHILE loop is now false, causing command execution to leave the WHILE loop. Macro Main finishes execution.
2-13 DISPLAY AND SOUND CONTROL

TMAC allows creation of new screen displays and changes to display screen configurations. Different colors can be used. Windows can be created and moved. Pixels, boxes, lines and figures may be drawn and placed anywhere on the color display. Screen page control makes applications such as animation possible. Audio cues can be implemented.

Some commands work in both the HOST or Sp Tst. Other commands work only with one or the other as indicated.

2-13-1 COLORS

Table 2-2 lists the 16 colors, according to selection number, available on the color display.

<table>
<thead>
<tr>
<th>COLOR</th>
<th>NUMBER</th>
<th>COLOR</th>
<th>NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black</td>
<td>0</td>
<td>Dark Gray</td>
<td>8</td>
</tr>
<tr>
<td>Dark Blue</td>
<td>1</td>
<td>Blue</td>
<td>9</td>
</tr>
<tr>
<td>Dark Green</td>
<td>2</td>
<td>Green</td>
<td>10</td>
</tr>
<tr>
<td>Dark Cyan</td>
<td>3</td>
<td>Cyan</td>
<td>11</td>
</tr>
<tr>
<td>Dark Red</td>
<td>4</td>
<td>Red</td>
<td>12</td>
</tr>
<tr>
<td>Dark Magenta</td>
<td>5</td>
<td>Magenta</td>
<td>13</td>
</tr>
<tr>
<td>Brown</td>
<td>6</td>
<td>Yellow</td>
<td>14</td>
</tr>
<tr>
<td>Light Gray</td>
<td>7</td>
<td>White</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 2-2 Colors and Color Selection Numbers

COLOR commands change the color selection for the color display except for Soft Function Keys and menus which are changed by EDIT:COLOR commands (HOST only).

Colors are entered using selection number or constant name (see Appendix A).

The following commands are used to control the foreground and background colors, the text and background colors of built-in HOST menus and the colors for the Soft Function Keys:

COLOR COLOR? BCOLOR

The following EDIT commands only apply to the HOST.

EDIT:COLOR:MENU     EDIT:COLOR:MENU?
EDIT:COLOR:LETTER   EDIT:COLOR:LETTER?
EDIT:COLOR:SOFT:BOX EDIT:COLOR:SOFT:BOX?
EDIT:WIDTH

See Section 3 for a description of each command.
2-13-2 COLOR DISPLAY

The display subsystem divides the color display screen into pixels: 640 in the horizontal direction and 350 in the vertical direction. Window and graphic commands use these pixels to determine screen location. Screen locations are specified using a coordinate system with point 0,0 located in the top left corner of the display screen. The positive horizontal component is to the right and the positive vertical component is in the downward direction.

2-13-3 WINDOWS

Windows, used primarily for menus, can be created and moved. There can be up to 15 windows on a screen at one time. Windows are numbered in the order they are opened and must be closed in reverse order if overlapped. The last window opened is the selected window unless selection is changed by a WSEL command. Windows can be any size or color and can be moved to any location on the screen. Following are the window commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Argument</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLS</td>
<td></td>
</tr>
<tr>
<td>WOPEN</td>
<td>SCREEN:USER</td>
</tr>
<tr>
<td>WSEL</td>
<td>USER</td>
</tr>
<tr>
<td></td>
<td>WMOVE</td>
</tr>
<tr>
<td></td>
<td>WINDOW?</td>
</tr>
<tr>
<td></td>
<td>WCLOSE</td>
</tr>
</tbody>
</table>

See Section 3 for a description of each command.

Print commands executed while a window is open prints inside the currently selected window.

To print outside the windows, select window 0 (background screen).

Example:

```
VAR Win1, Win2
CLS
WOPEN green, 100, 50, 250, 300
Win1=WINDOW?
WOPEN red, 300, 100, 550, 250
PRINT "WIN 2"
Win2=WINDOW?
WSEL Win1
PRINT "WIN 1"
WSEL 0
PRINT "THIS IS THE BACKGROUND"
```

These commands produce the following display:

![Window Example](image)

Figure 2-1 Window Example

The display control software forces the window locations and sizes to the nearest multiple of eight (byte size). Therefore, the user should determine the location values prior to opening or moving a window.
2-13-4 SOFT FUNCTION KEY DISPLAYS

Soft Function Key definitions and the Soft Function Key frame at the bottom of the display screen are added using the following commands:

**KEYPAD:SOFT**  **KEYPAD:LABel**  **KEYPAD:ERASE**

See Section 3 for a description of each command.

Example:  
```plaintext
CLS
KEYPAD:SOFT
KEYPAD:LAB 3, "Test"
```

The following is shown on the Test Set color display:

![Soft Function Key Example](image-url)

Figure 2-2  Soft Function Key Example
2-13-5  GRAPHICS AND TEXT

The following graphics and text commands create text, pixels, lines, boxes and user-defined shapes for display:

<table>
<thead>
<tr>
<th>XY</th>
<th>PIXEL</th>
<th>DRAW</th>
<th>BOX</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEIGHT</td>
<td>XYPRINT</td>
<td>PRINT</td>
<td>ELLIPSE¹</td>
</tr>
<tr>
<td>ICON¹</td>
<td>CENTER²</td>
<td>ERASE:TEXT²</td>
<td></td>
</tr>
<tr>
<td>LJPRINT²</td>
<td>PIXLEN²</td>
<td>RJPINT²</td>
<td>HPRINT²</td>
</tr>
</tbody>
</table>

1. HOST only.
2. Sp Tst only.

For Sp Tst operation, the PRINT command prints out the OPT. RS-232 Connector and the HPRINT (host print) command prints on the color display of the Test Set.

See Section 3 for a description of each command.

Example:

```
XY 400,100
COLOR blue, black
PIXEL
DRAW 200,100,250,300, red
BOX 0,250,20,350,120, green
BOX 1,280,200,550,280, magenta
```

See Table 2-2 for available colors and color numbers.

This set of commands creates the following graphics displayed in Figure 2-3.

- Blue pixel at point 400,100 and red line from point 200,100 to point 250,300.
- Hollow green box with top left corner at point 250,20 and lower right corner at point 350,120.
- Solid magenta box with top left corner at point 280,200 and lower right corner at point 550,280.

![Figure 2-3 Graphics Examples](image-url)
Example: DATA bits={#hPPPPFF3FF,#hFFCFFPPP,#hFFFFF3FF,#hFFCFFFFF,#hFFFFF3FF,#hFFCFFPPP,#hFFFFF3FF,#hFFCFFFFF,#h00F003C0,#h00F0000F,#h00F003C0,#h00F0000F,#h00F003C0,#h00F0000F,#h00F003C0,#h00F0000F,#h00F003FF,#hFC0FPPP,#hFFFFF3FF,#h00F003FF,#hFC0FPPP,#hFFFFF3FF,#h00F003FF,#hFC0FPPP,#hFFFFF3C0,#h00F7E00,#h00F003C0,#h00F3F00,#h00F003C0,#h00F0F80,#h00F003C0,#h00F07E00,#hFFFFF3C0,#h00F01F8,#hFFFFF3C0,#h00F003C0,#h00F003E,#hFFFFF3C0,#h00F001F}

XY 300,130
ICON 64,20, bits

The example creates the following image:

![Image of IFR]

Figure 2-4 ICON Example

2-13-6 VIDEO PAGE CONTROL (HOST Only)

Two screen displays are available: one active and one static for use as pages to switch back and forth. Two video pages allow the user to construct or make changes to a screen display before actually displaying the new screen. Video Page Control allows the user to specify which video page to currently display or to copy from one to the other.

VIDEOpage:SET VIDEOpage:COPY

See Section 3 for a description of each command.

2-13-7 AUDIO TONES

Audio tones are generated using the SOUND command (see SOUND command in Section 3).

There are two predefined macros, Chirp_1 and Chirp_2, that produce a short series of tones. Chirp_1 and Chirp_2 are detailed in Appendix A.
2-14 SYSTEM COMMANDS

2-14-1 SYSTEM KEY COMMANDS

The System Key commands assign Front Panel Keys to command strings that can execute macros. Command strings are limited to 80 characters. Only 16 keys can be assigned at one time. Appendix B lists Front Panel Keys with keycodes. Appendix A lists Front Panel Keys having predefined constants.

The following are the Key commands:

- **SYSTem:KEY:DEFine**
- **SYSTem:KEY:DELeTe**
- **KEY**
- **KEYPAd:CLAIM**
- **KEYPAd:UNCCLAIM**
- **KEY?**

1. HOST only.

See Section 3 for a description of each command.

2-14-2 SYSTEM ERROR COMMAND

The **SYSTem:ERRor** command reports system errors. See Section 3 for a description of this command.

2-14-3 SYSTEM DEFAULTS COMMANDS (HOST Only)

The System Defaults commands restore settings in the HOST to default states.

- **SYSTem:DEFaults**
- **SYSTem:CURsor:DEFaults**

See Section 3 for a description of each command.

2-14-4 SYSTEM PLOT COMMANDS (HOST Only)

The System Plot commands select the GPIB or HOST RS-232 Connector for plotter output.

- **SYSTem:PLOT:GPIB**
- **SYSTem:PLOT:SERial**

See Section 3 for a description of each command.

2-14-5 SYSTEM TIME AND DATE COMMANDS

System Time and Date commands allow setting and recording the date and time, in addition to determining elapsed time.

- **SYSTem:DATE?**
- **SYSTem:DATE**
- **SYSTem:TIME?**
- **SYSTem:TIME**
- **SYSTem:MILLIsec?**
- **TICKS?**

See Section 3 for a description of each command.
2-14-6 SYSTEM RS-232 CONFIGURE COMMANDS

The following commands, applying only to the HOST, allow the Test Set to control another device using serial RS-232 communication. Commands unique to the device controlled are sent as strings. Responses are received as strings.

```
SYSTem:PTHrough:SERial
SYSTem:PTHrough:SERial:QUEue?
SYSTem:PTHrough:SERial:KEY?
```

The following commands, applying to the HOST and Sp Tst, edit the RS-232 parameters remotely and select the RS-232 Connector for printer output:

```
SYSTem:COMMunicate:SERial:BAUD
SYSTem:COMMunicate:SERial:PARity
SYSTem:COMMunicate:SERial:BIiT
SYSTem:COMMunicate:SERial:SBITs
SYSTem:COMMunicate:SERial:PACE
SYSTem:COMMunicate:SERial:ECHO
SYSTem:COMMunicate:SERial:PRINTer
```

See Section 3 for a description of each command.

2-15 GPIB OPERATION

The Sp Tst operates as a GPIB Talker/Listener device. The HOST operates as a GPIB Talker/Listener, Controller, Talk Only or Listen Only device. The GPIB Mode of Operation is selected using commands received through the RS-232 Connector, commands from a macro executed within the Test Set or from the Front Panel Auxiliary Functions Menu.

Commands are received and transmitted as strings on GPIB. Convert all commands to strings before transmitting. Precede non-printable characters with the \ character to transmit.

Both the HOST and the Sp Tst can operate on the same GPIB at the same time, using different addresses.

2-15-1 OPERATING TEST SET USING GPIB COMMUNICATION (Sp Tst Only)

The Sp Tst is a GPIB Talker/Listener. The Sp Tst provides a user-defined Status Byte to generate Service Requests. The following commands are used when operating the Sp Tst on the GPIB:

```
GPIB:ADDResS    GPIB:MASK    GPIB:SRQ
```

See Section 3 for a description of each command.
2-15-2 OPERATING TEST SET USING GPIB COMMUNICATION (HOST Only)

For the Test Set to become a GPIB Talker/Listener, Talk/Listen is selected as the GPIB Mode. This function is performed remotely by executing the following command:

**SYSTem:COMMunicate:GPIB:ADDRESS**

The following suggested procedures operate the HOST Test Set using GPIB communication:

A. Initialization of the Test Set
   - Prepare Controller for adding the HOST Test Set to the GPIB.
   - Clear the HOST Test Set (see *CLS in Section 3).
   - Send "*ESE 1" command to Test Set. This command sets the event register of the Standard Event Status Register to 1 to allow reading of the operation complete bit.

B. Sending Commands to Test Set
   - Send "*SRE 36" command to the Test Set. This command sets the Service Request enable register to generate an RQS upon the occurrence of an OPC or Error.
   - Send desired command with "*OPC" command following.
   - Wait for RQS bit to be set (bit 6 of Status Byte) or for a time out.
   - Retrieve Status Byte of Test Set and store response to value variable.
   - If (value AND 4) is true, send error has occurred.
   - If (value AND 4) is false and (value AND 32) is true, command executed.

C. Sending Queries to Test Set
   - Send "*SRE 20" command to Test Set. This command sets the Service Request enable register to report the Message Available (MAV) bit and error bit.
   - Send query command to Test Set.
   - Wait for RQS bit to be set (bit 6 of Status Byte) or for a time out.
   - Retrieve Status Byte of Test Set and store response to value variable.
   - If (value AND 4) is true, send error has occurred.
   - If (value AND 4) is false and (value AND 16) is true, send read command to return results of query.

D. Sending Macros to Test Set
   Perform the following procedure until all command lines of macro are sent to Test Set.
   - Send command line to Test Set.
   - Wait for CMPL (completed I/O) or for a time out.
   - If time out, stop sending of macro. Restart sending of macro (see *PMC in Section 3).
2-15-3  TEST SET OPERATING AS GPIB CONTROLLER (HOST Only)

For the Test Set to become a GPIB controller, the address of the Test Set must be set to an address that does not conflict with a slave device address (30 is advised) and the GPIB Operation Mode must be set to Controller. The commands used to set these parameters are:

SYSTem:COMMunicate:GPIB:ADDRESS
SYSTem:COMMunicate:GPIB:CONTroller

Once the Test Set is made a GPIB Controller, the talker/listener status is returned by repeating the SYST:COMM:GPIB:ADDR command. The following commands are used when operating the Test Set as a GPIB Controller:

SYSTem:COMMunicate:GPIB:SLAVe
SYSTem:PTHRough:GPIB
SYSTem:PTHRough:GPIB?
SYSTem:COMMunicate:GPIB:DCL
SYSTem:COMMunicate:GPIB:GET
SYSTem:COMMunicate:GPIB:CMD
SYSTem:COMMunicate:GPIB:SRQ?
SYSTem:COMMunicate:GPIB:SPOLL?

See Section 3 for a description of each command.

2-15-4  TEST SET OPERATING AS TALKER ONLY (HOST Only)

When the Test Set becomes a GPIB Talk Only device, addressing is unnecessary. The Test Set must be the only talking device on the GPIB bus and the remaining devices must be compatible listeners. The following command selects Talk Only as the GPIB Mode of Operation.

SYSTem:COMMunicate:GPIB:TONly

See Section 3 for a description of this command.

To select the Talk/Listen GPIB Mode of Operation again, reset the Test Set address using the SYST:COMM:GPIB:ADDR command.

2-15-5  TEST SET OPERATING AS LISTENER ONLY (HOST Only)

When the Test Set becomes a GPIB Listen Only device, the address is set first using the SYST:COMM:GPIB:ADDR command. After setting the address, the following command selects Listen Only as the GPIB Mode of Operation.

SYSTem:COMMunicate:GPIB:LONly

See Section 3 for a description of this command.

To select the Talk/Listen GPIB Mode of Operation again, reset the Test Set address using the SYST:COMM:GPIB:ADDR command.
2-15-6  TEST SET GPIB PRINT CONTROL (HOST Only)
The following command allows the user to direct the output of the printer to the GPIB Connector:

```
SYSTem:COMMunicate:GPIB:PRINTer
```

See Section 3 for a description of this command.

2-15-7  TEST SET FREQUENCY CONTROL (HOST Only)
The following commands provides the user the following remote capabilities:

- To slave the RF Generator, Receiver and Spectrum Analyzer frequency, together. When a user enters a frequency through either one of the three operations, the remaining two operations automatically change to the identical frequency.
- To configure, manipulate or query settings in the Frequency List.

The following TMAC commands provide frequency controls for the HOST in that affect multiple operation modes:

```
SYSTem:FREQuency:LOCK           SYSTem:FREQuency:LOCK?
SYSTem:FREQList:GENERator       SYSTem:FREQList:GENERator?
SYSTem:FREQList:OFFset          SYSTem:FREQList:OFFset?
SYSTem:FREQList:RECeiver        SYSTem:FREQList:RECeiver?
SYSTem:FREQList:SCAN            SYSTem:FREQList:SCAN?
```

See Section 3 for a description of each command.
2-16 STATUS SUBSYSTEM (HOST Only)

TMAC provides an implementation of the Event Status register structure as designated by IEEE-488.2 and SCPI Standards. Standard Event, Operation, Operation Instrument, Questionable, Instrument and Instrument Summary Status Registers and their relationship to one another are shown in Figure 2-6.

2-16-1 STATUS REGISTERS

Each Status Register shown in Figure 2-6 consists of a condition, event and enable register shown in Figure 2-5. The condition register continually receives data coming into the Status Register. When a bit is changed from 0 to 1 for the first time since being cleared, the same bit in the event register is set to 1. The event register is only changed by the first 0 to 1 transitions that occur in the condition register. The condition or event registers are cleared when read.

The 16 bits of the event register and the 16 bits of the enable register undergo a bitwise AND operation. In order for an event register bit to be reported, the corresponding bit in the enable register must be set to 1. The 16 bits resulting from this bitwise AND operation undergo an OR operation with each other. The result is 1 if one or more bits were 1.

---

Figure 2-5 HOST Status Register Description
Figure 2-6 HOST Status Register Configuration
STATUs:PRESet
Sets all bits in enable registers to an initialized condition. Presets the enable registers of the following status registers to the associated condition:

<table>
<thead>
<tr>
<th>STATUS REGISTER</th>
<th>CONDITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Status</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Operation Instrument</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Questionable Status</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Instrument Status</td>
<td>ON (#h7FFF)</td>
</tr>
<tr>
<td>Instrument Summary</td>
<td>ON (#h7FFF)</td>
</tr>
</tbody>
</table>

2-16-2 STATUS BYTE
The Status Byte reports an occurrence in any of the Status Registers, and is used to activate a SRQ message and the RQS bit (bit 6 of the Status Byte) (see Figure 2-6). A Status Byte bit determines an SRQ and RQS if the corresponding Service Request enable register bit is set to 1. Bits 0 to 5 and bit 7 of the Status Byte and the same bits of the Service Request enable register undergo a bitwise AND operation. The resulting bits of the AND operation undergo an OR operation. If any of the enabled Status Byte bits are 1, an SRQ is generated and the RQS Status Byte bit (bit 6) is set to 1.

![Diagram of Status Byte and Service Request Enable Register](image)

Figure 2-7 HOST RQS Bit and SRQ Generation
The following commands pertain to the Status Byte and Service Request enable register:

*SRE             *SRE?    *STB?

See Section 3 for a description of each command.
2-16-3 STANDARD EVENT STATUS REGISTER

The Standard Event Status Register contains an operation complete bit, several error bits and a power on bit. Figure 2-6 shows the Standard Event Status Register bits. The power on bit is set to 1 following power up of the Test Set. The following commands control this register:

*ESE *ESE? *ESR? *OPC *OPC?

See Section 3 for a description of each command.

Example:  *DMC "ERROR_CHECK",BEGIN
          *ESE 61
          IF (*ESR & 1)= 0
          IF (*ESR & 4)=1
          PRINT "QUERY ERROR"
          ELIF (*ESR & 8)=1
          PRINT "DEVICE SPECIFIC ERROR"
          ELIF (*ESR & 16)=1
          PRINT "EXECUTION ERROR"
          ELIF (*ESR & 32)=1
          PRINT "COMMAND ERROR"
          ELSE
          PRINT "NO ERROR DETECTED"

2-16-4 OPERATION STATUS REGISTER

The Operation Status Register contains the Waiting For Arm, Waiting For Trigger and Instrument Summary bits. The Operation Instrument Register sets the Instrument Summary bit (see 2-16-5). The following commands control the Operation Status Register:

STATus:OPERation:CONDition?  STATus:OPERation:ENABLE
STATus:OPERation:ENABLE?  STATus:OPERation:EVENT?

See Section 3 for a description of each command.
2-16-5 OPERATION INSTRUMENT REGISTER

The Operation Instrument Register is used with the INITiate and FETCH commands. Once an INITiate command is completed and the meter is ready to be read, the appropriate meter ready bit is set to 1. This indicates the meter is ready for the FETCH command. The following STATUS commands control the Operation Instrument Register:

```
STATus:OPERation:INSTRument:CONDition?
STATus:OPERation:INSTRument:ENABLE
STATus:OPERation:INSTRument:ENABLE?
STATus:OPERation:INSTRument:EVENT?
```

See Section 3 for a description of each command.

2-16-6 QUESTIONABLE STATUS REGISTER

The Questionable Status Register reports events that could bring Test Set results into question. The Instrument Status Register provides the input. Figure 2-6 shows events reported by both registers. The following STATUS commands control the Questionable Status Register:

```
STATus:QUEStionable:CONDition?     STATus:QUEStionable:ENABLE
STATus:QUEStionable:ENABLE?        STATus:QUEStionable:EVENT?
```

See Section 3 for a description of each command.

2-16-7 INSTRUMENT STATUS REGISTER

The Instrument Status Register reports meter readings exceeding the upper and lower limit of the AF, RF Power, FM Deviation, AM Modulation and Distortion Meters. The results of the Instrument Summary Status Register (an extension of the Instrument Status Register) are also reported. The following STATUS commands control the Instrument Status Register.

```
STATus:QUEStionable:INSTRument:CONDition?
STATus:QUEStionable:INSTRument:ENABLE
STATus:QUEStionable:INSTRument:ENABLE?
STATus:QUEStionable:INSTRument:EVENT?
```
2-16-8 INSTRUMENT SUMMARY STATUS REGISTER

The Instrument Summary Status Register acts as an extension of the Instrument Status Register. The Instrument Summary Status Register reports the exceeding of the upper and lower limit of the SINAD and Phase Meter and Digital Multimeter. Following are STATUS commands controlling the Instrument Summary Status Register.

STATus:QUEStionable:INSTRument:ISummary:CONDi tion?
STATus:QUEStionable:INSTRument:ISummary:ENABLe
STATus:QUEStionable:INSTRument:ISummary:ENABLe?
STATus:QUEStionable:INSTRument:ISummary:EVENT?

The following example performs a DMM reading and checks the Status Byte for the reporting of the DMM Upper or Lower Limit being exceeded. The enable registers of the Instrument Summary, Instrument and Questionable Status Registers are set to pass the information to the Status Byte.

Example: *DMC "DMM_READ",BEGIN
STAT:QUE:INSTR:ISUM:ENABLE 12 // Define macro named DMM_READ.
STAT:QUE:INSTR:ENABLE 2 // Set enable register to read DMM
STAT:QUE:ENABLE 8192 // Upper and Lower Limit exceeded.
*CLS // Set enable register to pass the
SCREEN:DMW // Instrument Summary Status
PPRINT M_DMM? // Register result.
IF (*STB? & 8) != 0 // Set enable register to pass the
ELSE // Instrument Status Register result.
PPRINT "limit exceeded" // Clear all condition and event
ELSE // registers.
PPRINT "limit not exceeded" // Display DMM Operation Screen.
ENDIF // Print DMM reading to Host.
PPRINT "limit exceeded to Host." // If Status Byte reports
ELSE // Questionable Status bit as 1,
PPRINT "limit not exceeded to Host." // print limit exceeded to Host.
ENDIF // If Status Byte reports
ELSE // Questionable Status bit as 0,
PPRINT "limit not exceeded to Host." // print limit not exceeded to Host.
ENDIF // End IF statement.
END // End macro DMM_READ.
2-17  IEEE 488.2 COMPLIANCE COMMANDS

TMAC contains the following IEEE-488.2 mandated commands:

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>DESCRIPTION</th>
<th>WHERE USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>*CLS</td>
<td>Clear Status</td>
<td>HOST and Sp Tst</td>
</tr>
<tr>
<td>*DDT</td>
<td>Define Device Trigger</td>
<td>HOST</td>
</tr>
<tr>
<td>*DMC</td>
<td>Define Macro</td>
<td>HOST and Sp Tst</td>
</tr>
<tr>
<td>*EMC</td>
<td>Enable Macro</td>
<td>HOST</td>
</tr>
<tr>
<td>*EMC?</td>
<td>Enable Macro query</td>
<td>HOST</td>
</tr>
<tr>
<td>*ESE</td>
<td>Standard Event Status Enable</td>
<td>HOST</td>
</tr>
<tr>
<td>*ESE?</td>
<td>Standard Event Status Enable query</td>
<td>HOST</td>
</tr>
<tr>
<td>*ESR?</td>
<td>Standard Event Status Register query</td>
<td>HOST</td>
</tr>
<tr>
<td>*IDN?</td>
<td>Identification query</td>
<td>HOST and Sp Tst</td>
</tr>
<tr>
<td>*LMC?</td>
<td>List Macro Query</td>
<td>HOST</td>
</tr>
<tr>
<td>*OPC</td>
<td>Operation Complete</td>
<td>HOST</td>
</tr>
<tr>
<td>*OPC?</td>
<td>Operation Complete query</td>
<td>HOST</td>
</tr>
<tr>
<td>*OPT?</td>
<td>Option identification query</td>
<td>HOST</td>
</tr>
<tr>
<td>*PMC</td>
<td>Purge Macro</td>
<td>HOST and Sp Tst</td>
</tr>
<tr>
<td>*RCL</td>
<td>Recall</td>
<td>HOST</td>
</tr>
<tr>
<td>*RST</td>
<td>Reset</td>
<td>HOST</td>
</tr>
<tr>
<td>*SAV</td>
<td>Save</td>
<td>HOST</td>
</tr>
<tr>
<td>*SRE</td>
<td>Service Request Enable</td>
<td>HOST</td>
</tr>
<tr>
<td>*SRE?</td>
<td>Service Request Enable query</td>
<td>HOST</td>
</tr>
<tr>
<td>*STB?</td>
<td>Read Status Byte query</td>
<td>HOST</td>
</tr>
<tr>
<td>*TRG</td>
<td>Trigger</td>
<td>HOST</td>
</tr>
<tr>
<td>*TST?</td>
<td>Self-Test query</td>
<td>HOST</td>
</tr>
<tr>
<td>*WAI</td>
<td>Wait To Continue</td>
<td>HOST and Sp Tst</td>
</tr>
</tbody>
</table>

See Section 3 for a description of each command.

Table 2-3  IEEE 488.2 Compliance Commands

2-18  GENERAL AND SPECIFIC TMAC COMMANDS

Section 3 describes in detail TMAC commands that are common to both the HOST and Sp Tst. TMAC commands specific to HOST and Special Test are described in Sections 6 and 9, respectively.
SECTION 3 - GENERAL TMAC COMMANDS

The following pages contain General TMAC commands (commands that apply to IFR-1900CSA Host and Special Test [Sp Tst]); listing use, syntax and action. Examples are also provided to aid in the understanding of the commands. Cross references to related commands are also included. Commands are listed in alphabetical order.

+  

**USE**  
Addition operator or positive unary operator.

**SYNTAX**  
argument1+argument2 or +argument3

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Same as argument1, except must be of the same type used for argument1.</td>
</tr>
<tr>
<td>argument3</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

**ACTION**  
Adds two expressions or indicates the positive value of an expression. (argumentX may be any numerical value, variable or any mathematical operation resulting in a numerical value.)

**EXAMPLES**

G=4+F  // Assigns 4 added to the value of F to G.
AA = B1 + B2  // Assigns value of B1 added to B2 to AA.
+X  // The value of X (i.e., no change).

-  

**USE**  
Subtraction operator or negative unary operator.

**SYNTAX**  
argument1-argument2 or -argument3

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Same as argument1, except must be of the same type used for argument1.</td>
</tr>
<tr>
<td>argument3</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

**ACTION**  
Subtracts the value of argument2 from the value of argument1 or assigns a negative value to the value of argument3.

**EXAMPLES**

H= I-5  // Assigns the resultant value of 5 subtracted from the value of I to H.
V = U-T  // Assigns the resultant value of T subtracted from the value of U to V.
-B  // The negative or opposite value of B.

3-1
## USE

Bitwise complement.

### SYNTAX

\[-\text{argument}\]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

### ACTION

Inverts the binary representation of the operand using the one's complement form of negative numbers.

### SEE ALSO

\!, \&, ^, \!, AND, OR

### EXAMPLE

\[X = -6\]  // 0110 (6) is inverted to 1001  
// (one's complement of -7) and  
// assigns -7 to X.

### !

Logical negation (NOT) unary operator.

### USE

Logical negation (NOT) unary operator.

### SYNTAX

\![argument]\]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

### ACTION

Reverses the truth value.

### SEE ALSO

\!, \&, ^, \!, AND, OR

### EXAMPLE

\[!D\]  // The example is true when D equals 0.

### ** (Double Asterisk)

Exponential operator (does not conform to SCPI Standard).

### USE

Exponential operator (does not conform to SCPI Standard).

### SYNTAX

\[\text{argument1}^{*}\text{argument2}\]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>argument2</td>
<td>Same as argument1.</td>
</tr>
</tbody>
</table>

### ACTION

Raises value of argument1 to the power of the value of argument2.

### EXAMPLES

\[y=4^{*}\text{-}2\]  // Assigns 4 raised to the -2 power to y.  
\[\text{MM} = \text{NN}^{*}\text{4}\]  // Assigns NN raised to the 4th power  
// to MM.  
\[\text{Freq} = \text{C}^{*}\text{D}\]  // Assigns C raised to the D power to Freq.
* (Single Asterisk)

USE  Multiplication operator.

SYNTAX  \texttt{argument1*argument2}

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>argument2</td>
<td>Same as argument1.</td>
</tr>
</tbody>
</table>

ACTION  Multiplies argument1 by argument2.

SEE ALSO  /

EXAMPLES  
\begin{align*}
x &= 4*6 & // \text{Assigns 4 multiplied by 6 to } x. \\
T &= 25*S & // \text{Assigns 25 multiplied by S to } T. \\
JJ &= K*LL & // \text{Assigns K multiplied by LL to } JJ.
\end{align*}

/

USE  Division operator.

SYNTAX  \texttt{argument1/argument2}

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>argument2</td>
<td>Same as argument1 except cannot be zero.</td>
</tr>
</tbody>
</table>

ACTION  Divides the value of argument1 by the value of argument2.

SEE ALSO  *

EXAMPLES  
\begin{align*}
Y &= 8/3.867 & // \text{Assigns the quotient of 8 divided by 3.867 to } Y. \\
Z &= x/7 & // \text{Assigns the quotient of } x \text{ divided by 7 to } Z. \\
P &= Q/R & // \text{Assigns the quotient of } Q \text{ divided by } R \text{ to } P.
\end{align*}

A command to divide by zero results in an error message.
USE
Modulo operator (does not conform to SCPI Standard).

SYNTAX
argument1%argument2

ELEMENT DESCRIPTION
argument1 Any constant, variable, number or any expression that evaluates to a number.
argument2 Same as argument1, except cannot be zero.

ACTION
Returns the remainder (an integer) of a division operation. If not integers, the operands are truncated.

EXAMPLES
x=7%3 // Assigns 7 modulo 3 to x (x=1).
RR=16%4 // Assigns 16 modulo 4 to RR (RR=0).
N=20.5%3.3 // Assigns 20 modulo 3 to N (N=2).
M=27%L // Assigns 27 modulo the value of L to M.

<<

USE
Shift left operator.

SYNTAX
argument1<<argument2

ELEMENT DESCRIPTION
argument1 Any constant, variable, number or any expression that evaluates to a positive number.
argument2 Same as argument1.

ACTION
Shifts the binary representation of the integer value of argument1 to the left the number of bits specified by the integer value of argument2. Emptied bit positions (as result of shifting) are filled with zeros. Shift operators use the two's complement representation for negative numbers.

Shifts a maximum of 31 bit positions, but argument2 can be greater than 31. Use the following equation to calculate the number of bit positions shifted to the left: NumShifted = argument2 % 32.

SEE ALSO
>>

EXAMPLES
VAR DIV4
m = 2
DIV4 = 75<<m // Shifts 75 (1001011) left 2 bits to
// to become 300 (100101100) and is
// assigned to DIV4. DIV4 = 30010.

X = -19<<3 // Shifts -19
// (#b1111111111111111111111101010101)
// left 3 bits to become -152
// (#b111111111111111111111111011010100)
// and is assigned to X. X = -15210.
USE  Shift right operator.

SYNTAX  \texttt{argument1}>>\texttt{argument2}

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{argument1}</td>
<td>Any constant, variable, number or any expression that evaluates to a positive number.</td>
</tr>
<tr>
<td>\texttt{argument2}</td>
<td>Same as \texttt{argument1}.</td>
</tr>
</tbody>
</table>

ACTION Shifts the binary representation of the integer value of \texttt{argument1} to the right the number of bits specified by the integer value of \texttt{argument2}. Empty bit positions (as result of shifting) are filled with zeros. Shift operators use the two's complement representation for negative numbers.

Shifts a maximum of 31 bit positions, but \texttt{argument2} can be greater than 31. Use the following equation to calculate the number of bit positions shifted to the right: \texttt{NumShifted = argument2 \% 32}.

SEE ALSO \texttt{<<}

EXAMPLE  \texttt{Q = 14}>>\texttt{1} // Shifts 14 (1110) right 1 bit to // to become 7 (0111) and is // assigned to \texttt{Q}. \texttt{Q = 7_{10}}.

&&

USE  Bitwise AND

SYNTAX  \texttt{argument1}&&\texttt{argument2}

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{argument1}</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>\texttt{argument2}</td>
<td>Same as \texttt{argument1}.</td>
</tr>
</tbody>
</table>

ACTION Performs AND operation on each bit of the integer binary representation of two operands. Table 3-1 lists the bit combinations and the AND operation results.

<table>
<thead>
<tr>
<th>OPERAND 1</th>
<th>OPERAND 2</th>
<th>AND RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3-1 Bitwise AND Operation Results

SEE ALSO  !, ~, ^, |, AND, OR
EXAMPLES

Jj = 4 & 6
// 0100 (4) is ANDed to 0110 (6).
// The resultant value is 0100 (4).
// Jj is assigned the value of 4.

Ww = 12 & 10
// 1100 (12) is ANDed to 1010 (10).
// The resultant value is 1000 (8).
// Ww is assigned the value of 8.

USE

Bitwise XOR (conflicts with SCPI Standard for Exponential operator).

SYNTAX

argument1^argument2

ELEMENT DESCRIPTION

argument1 Any constant, variable, number or any expression that evaluates to a number.

argument2 Same as argument1.

ACTION

Performs Exclusive-OR (XOR) operation on each bit of the integer binary representation of argument1 and argument2. Table 3-2 lists the bit combinations and the XOR operation results. Operand 1 represents a bit in argument1 andOperand2 represents the associated bit in argument2.

<table>
<thead>
<tr>
<th>OPERAND 1</th>
<th>OPERAND 2</th>
<th>XOR RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3-2 Bitwise XOR Operation Results

SEE ALSO

!, ~, &, !

EXAMPLES

y = 4 ^ 6
// The left bits of 4 (100) and 6 (110) are the same (both 1) providing an XOR result of 0. The middle bits are different (0 and 1) providing an XOR result of 1. The right bits are the same (both 0) providing an XOR result of 0. Therefore the total result of the XOR operation is 010 (2); y is assigned the value of 2.

v = 12 ^ 10
// 12 (1100) and 10 (1010) is exclusive-OR'ed together, resulting in the value of 0110 (6) because the end bits are the same and the middle bits are different. v is assigned the value of 6.
USE
Bitwise OR operator.

SYNTAX
argument1:argument2

ELEMENT DESCRIPTION
argument1 Any constant, variable, number or any expression that evaluates to a number.
argument2 Same as argument1.

ACTION
Performs OR operation on each bit of the integer binary representation of the two operands. Table 3-3 lists the bit combinations and the OR operation results. Operand 1 represents a bit in argument1 and Operand2 represents the associated bit in argument2.

<table>
<thead>
<tr>
<th>OPERAND 1</th>
<th>OPERAND 2</th>
<th>OR RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3-3 Bitwise OR Operation Results

SEE ALSO
!, ~, &, ^

EXAMPLES
y = 4 | 2 // The left and middle bits of 4 (100) and 2 (010) both result in 1 (11), when OR'ed together, because at least one bit in each position is 1. The right bits of 4 (100) and 2 (010) result in 0 (0), when OR'ed together, because neither bit is 1. Therefore the total result of the OR operation is 110 (6). y is assigned the value of 6.

z = 12 | 10 // 12 (1100) and 10 (1010) are OR'ed together, resulting in the value 14 (1110), because at least one of the bits in each position of the two expressions is a 1 except for the right position which for both expression is 0. z is assigned the value of 14.
**USE**
Assignment or Equal logical operator.

**SYNTAX**
\[ x = \text{argument1} \text{ or command argument2} = \text{argument3} \]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x )</td>
<td>Any numerical or string variable.</td>
</tr>
<tr>
<td>( \text{argument1} )</td>
<td>Any variable, number, string or any expression that evaluates to the same type as ( x ).</td>
</tr>
<tr>
<td>( \text{command} )</td>
<td>Any decision point command (IF, ELIF, UNTIL, WHILE).</td>
</tr>
<tr>
<td>( \text{argument2} )</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>( \text{argument3} )</td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as ( \text{argument2} ).</td>
</tr>
</tbody>
</table>

**ACTION**
Assigns a value to a variable. Variable on the left of the = operator is given the value of the expression on the right if the same type as the variable.

-or-

Provides condition for a decision point command. If conditional expression created with = (logical equal) evaluates to TRUE, then resulting command associated with decision point command is performed.

**SEE ALSO**
!, AND, OR, !=, <, >, <=, >=, DO UNTIL, IF ..., WHILE

**EXAMPLES**
\begin{align*}
\text{x}=2 & \quad \text{// Assigns x the value of 2.} \\
\text{Mod} = w \times 2 + z & \quad \text{// Assigns the resultant value of expression} \\
& \quad \text{// to the right of = to Mod.} \\
\text{x} = x + 2 & \quad \text{// Assigns a new value to x that is 2} \\
& \quad \text{// greater than the previous value.}
\end{align*}

The assignment operator does not denote an equation.
!=

USE
Not Equal relational operator.

SYNTAX
argument1!=argument2

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as argument2.</td>
</tr>
</tbody>
</table>

ACTION
Evaluates a Not Equal condition to TRUE or FALSE. Used for decisions in macros.

SEE ALSO
!, AND, OR, =, <=, >=, DO UNTIL, IF ..., WHILE

EXAMPLE
x=3
if x!=4 // Evaluates conditional expression (x is not equal to 4) as TRUE.
    print y // Performs resulting command, because IF condition is TRUE.
endif

<

USE
Less Than relational operator.

SYNTAX
argument1<argument2

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as argument2.</td>
</tr>
</tbody>
</table>

ACTION
Evaluates a Less Than condition to TRUE or FALSE. Used for decisions in macros.

SEE ALSO
!, AND, OR, =, !=, >, <=, >=, DO UNTIL, IF ..., WHILE

EXAMPLE
var h=10, s=6
if h<s+4 // Evaluates conditional expression (h is less than s + 4) as FALSE.
    print y // Does not perform resulting command, because IF condition is FALSE.
endif
USE  
Greater Than relational operator.

SYNTAX  
argument1>argument2

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as argument2.</td>
</tr>
</tbody>
</table>

ACTION  
Evaluates a Greater Than condition to TRUE or FALSE. Used for decisions in macros.

SEE ALSO  
!, AND, OR, =, !=, >, <=, >=, DO UNTIL, IF ..., WHILE

EXAMPLES

```c
var x=3
if x>4  // Evaluates conditional expression (h is greater than 4) as FALSE.
     // Does not performs resulting command.
    print y
    // because IF condition is FALSE.
endif
if 5>4 // Evaluates conditional expression (5 is greater than 4) as TRUE.
    print y // Performs resulting command, because IF // condition is TRUE.
endif
```

<=

USE  
Less Than Or Equal relational operator.

SYNTAX  
argument1<=argument2

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument1</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td>argument2</td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as argument2.</td>
</tr>
</tbody>
</table>

ACTION  
Evaluates a Less Than Or Equal condition to TRUE or FALSE. Used for decisions in macros.

SEE ALSO  
!, AND, OR, =, !=, <, >, >=, DO UNTIL, IF ..., WHILE

EXAMPLE

```c
var t=16
if t*2<=32 // Evaluates conditional expression
    // (2t is less than or equal to 32) as
    // TRUE.
    print y // Performs resulting command, because IF
    // condition is TRUE.
endif
```
**USE**
Greater Than Or Equal relational operator.

**SYNTAX**
`argument1>=argument2`

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>argument1</code></td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string.</td>
</tr>
<tr>
<td><code>argument2</code></td>
<td>Any constant, variable, number, string or any expression that evaluates to the same type as <code>argument2</code>.</td>
</tr>
</tbody>
</table>

**ACTION**
Evaluates a Greater Than Or Equal condition to TRUE or FALSE. Used for decisions in macros.

**SEE ALSO**
!, AND, OR, =, !=, <, >, <=, DO UNTIL, IF ..., WHILE

**EXAMPLE**
```plaintext
var x=10, t=4
if x-2>=24/t    // Evaluates conditional expression (8 is greater than or equal to 6) as TRUE.
  print y      // Performs resulting command, because
  endif        // IF condition is TRUE.

x = y++
```

**USE**
Increment operator.

**SYNTAX**
`++variable` (increments before) or `variable++` (increments after)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>variable</code></td>
<td>Any numeric variable.</td>
</tr>
</tbody>
</table>

**ACTION**
Adds 1 to the value of a variable before or after execution of the command in which the operator is included.

**SEE ALSO**
`--`

**EXAMPLE**
```plaintext
y = 7
x = y++    // Assigns the value of y (7) to y, then
            // increments y. After the execution of
            // the command, x = 7 and y = 8.
```
**USE**

Decrement operator.

**SYNTAX**

--variable (decrements before) or
variable-- (decrements after)

**ELEMENT**  **DESCRIPTION**

variable  Any numeric variable.

**ACTION**

Subtracts 1 from the value of a variable before or after execution of the command in which the operator is included.

**SEE ALSO**

++

**EXAMPLE**

\[
y = 7
x = --y
\]

// Subtracts 1 from y, then assigns this value to x. After the second command is executed, x and y equal 6.

**CLS**

**USE**

To clear all Status Registers.

**SYNTAX**

*CLS

**ACTION**

Clear Status command. Sets all Status Registers including the Status Byte to zero.

**SEE ALSO**

SYSTem:ERRor?

**EXAMPLES**

See STATus:QUEStionable:INSTRument:ISUMmary.

**DDT (HOST Only)**

**USE**

To define what commands are executed by the Group Execute Trigger or *TRG command.

**SYNTAX**

*DDT

**ACTION**

Define Device Trigger command. Defines what commands are executed by the GET (Group Execute Trigger) command received at the GPIB Connector or the *TRG command.

**EXAMPLES**

*DDT SCOPE:TRIGGER:ONE;SCOPE:ARM  // Executes these commands when a GET (Group Execute Trigger) command is received at the GPIB Connector or a *TRG command is executed.
**DMC**

**USE**
To define macros and functions.

**SYNTAX**
*DMC "name", command; .... ; command

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the macro or function defined. The first character of the name is a letter while the remaining characters can be letters, digits and the underscore. The length of the name can be from 2 to 31 characters.</td>
</tr>
<tr>
<td>command</td>
<td>Any valid TMAC command or expression.</td>
</tr>
</tbody>
</table>

**ACTION**
Defines a macro or function with the name specified and containing the commands listed. After a macro or function is defined, entering the macro or function's name executes all command contained within.

Macros utilize several lines using the BEGIN and END commands. A function is a macro with a RETURN command used to return values. The RETURN command returns results directly through the active interface connector.

**SEE ALSO**
Macros (2-9), BEGIN, END, RETURN, *LMC, FORGET, *PMC

**EXAMPLES**

```
*DMC "Sq",SCREEN:USER;:INPUT X;Y=X**2;PPRINT X," squared is ",Y

/* The above *DMC command defines the following commands as a macro named Sq. Upon entering Sq, the INPUT command is executed, Y is set to X^2 and the PPRINT command is executed. */

*DMC "Square",BEGIN // Accomplishes the same as the Sq macro.
SCREEN:USER // The BEGIN and END commands allow the
INPUT X // macro to be written over several lines.
Y=X**2
PPRINT X," squared is ", Y
END

*DMC "SQ",RETURN($1**2) // Defines a function named SQ.
PRINT SQ 11 // Prints the result of the function
// named SQ (121).

*DMC "Av",RETURN(($1+$2+$3)/3) // Defines a function named Av.
X=Av 13,14,17 // Executes the function Av.
PPRINT X // Returns 14.66667.

*dmc "Sph_Sec_Surface",return(3.1415*$1/2*(4+$2+$3))
print Sph_Sec_Surface 4.4, 2.12, 6.7

/* The above *DMC command defines the function (3.1415*$1/2*(4+$2+$3)) as Sph_Sec_Surface. 104.9135 is printed on the screen. */
```
**EMC (HOST Only)**

**USE**
To enable/disable macros or return Enable Macro Command status.

**SYNTAX**
*EMC b or
*EMC?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Enable ($b = 1$) or disable ($b = 0$).</td>
</tr>
</tbody>
</table>

**ACTION**
Enables/disables execution of macros per value of $b$ or returns the Enable Macro Command status. Returns 1 if macro execution is enabled; 0 otherwise.

**SEE ALSO**
*DMC

---

**ESE (HOST Only)**

**USE**
To specify or return the contents of the Standard Event Status Enable Register.

**SYNTAX**
*ESE n or
*ESE?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Eight bit value. Range is 0 to 255.</td>
</tr>
</tbody>
</table>

**ACTION**
Specifies ($n$) or returns the contents of the Standard Event Status Enable Register.

**SEE ALSO**
Standard Event Status Register (2-16-3)
**ESR? (HOST Only)**

**USE**
To return the contents of the Standard Event Status Register.

**SYNTAX**
*ESR?*

**ACTION**
Returns and clears the contents of the Standard Event Status Register.

**SEE ALSO**
Status Subsystem (2-16)

**EXAMPLES**

```
*DMC "ERROR_CHECK", BEGIN
*ESE 61 // Enable Std Event Status Register.
IF (*ESR & 1)= 0 // Check Operation Complete bit.
  IF (*ESR? & 4)=1 // If Standard Event Status Register
    PRINT "QUERY ERROR" // reports Query Error, print message.
  ELIF (*ESR? & 8)=1 // If Standard Event Status Register
    PRINT "DEVICE SPECIFIC ERROR" // print message.
  ENDIF
ELSE // If Std Event Status Rgtr reports
  PRINT "EXECUTION ERROR" // Execution Error, print message.
  PRINT "COMMAND ERROR" // Register shows Command Error,
  // print message.
ENDIF
```

**IDN?**

**USE**
To return Test Set Identification.

**SYNTAX**
*IDN?*

**ACTION**
Returns the identification parameters of the Test Set. Includes the manufacturer and model names, serial number and various firmware versions.

**EXAMPLE**

An example return from a query to an HOST is as follows:

```
IFR SYSTEMS,IFR-1900, 4000, 1.05/C-03.01S/F-03.02S/M-3.00S
```

The serial number is 4000. The system firmware version is 1.05. The Counter firmware version is 3.01S. The Function Generator firmware version is 3.02S. The Monitor firmware version is 3.00S.

An example return from a query to an Sp Tst is as follows:

```
IFR 1900CSA, 200, 01.05, Aug 20 1997 15:19:53
```

The serial number is 200. The system firmware version is 1.05 with compile date and time.
*LMC? (HOST Only)

USE To return a listing of all currently defined macros.
SYNTAX *LMC?
ACTION Returns a listing of all currently defined macros (to include predefined macros; see Appendix A).
SEE ALSO *EMC

*OPC (HOST Only)

USE To set the Operation Complete bit of the Standard Event Status Register when all pending operations are finished.
SYNTAX *OPC
ACTION Sets the Operation Complete bit of the Standard Event Status Register when all pending operations are finished.
SEE ALSO *WAI

*OPC? (HOST Only)

USE To generate an ASCII 1 in output queue when operation is completed.
SYNTAX *OPC?
ACTION Generates an ASCII 1 in the output queue when operation is complete, activating the Message Available bit in the Status Byte.
SEE ALSO Standard Event Status Register (2-16-3.)

*OPT? (HOST Only)

USE To return the value from Options of the Configuration Report.
SYNTAX *OPT?
ACTION Returns the value from Options of the Configuration Report (see 3-3-10 of the IFR-1900 Operation Manual). Range of return values is 0 to 255.
**PMC**

**USE**
To purge all macros and declared variables.

**SYNTAX**
*PMC

**ACTION**
Deletes all macros and declared variables except the predefined macros listed in Appendix A.

HOST only: Global variables and array saved with the NVSAV command are not affected.

**SEE ALSO**
Macros (2-9), FORGET

---

**RCL (HOST Only)**

**USE**
To restore HOST to routings and settings stored in memory location n.

**SYNTAX**
*RCL n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 9.</td>
</tr>
</tbody>
</table>

**ACTION**
Recall command. Restores device to the environment (routings and settings) stored in local memory location n.

**SEE ALSO**
*RCL

---

**RST (HOST Only)**

**USE**
To reset HOST to factory defaults.

**SYNTAX**
*RST

**ACTION**
Reset command. Resets device to preset condition (factory defaults).

---

**SAV (HOST Only)**

**USE**
To save current routings and settings in memory location n.

**SYNTAX**
*SAV n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 9.</td>
</tr>
</tbody>
</table>

**ACTION**
Save command. Saves current environment (routings and settings) in local memory location n.

**SEE ALSO**
*RCL
**SRE (HOST Only)**

**USE**
To specify or return the contents of the Service Request Enable register.

**SYNTAX**
*SRE n or
*SRE?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Eight bit value. Range is 0 to 255.</td>
</tr>
</tbody>
</table>

**ACTION**
Service Request Enable command or query. Specifies or returns the contents of the Service Request Enable register.

**SEE ALSO**
Status Byte (2-16-2)

**STB? (HOST Only)**

**USE**
To return the contents of the Status Byte register.

**SYNTAX**
*STB?

**ACTION**
Status Byte query. Returns the contents of the Status Byte register.

*The Status Byte register is not cleared when read, but maintains the current value until the *CLS command is used (*CLS command also clears the condition and event registers of each Status Register).*

**SEE ALSO**
*CLS

**EXAMPLES**
See STATus:QUEStionable:INSTRument:ISUMmary.

**TRG (HOST Only)**

**USE**
To execute trigger command.

**SYNTAX**
*TRG

**ACTION**
Trigger command. Executes trigger command as defined by the *DDT command.

**SEE ALSO**
*DDT, (SCOPe:TRIGger:SOURce in Section 6-9)

**TST? (HOST Only)**

**USE**
To perform Self Test.

**SYNTAX**
*TST?

**ACTION**
Self-Test Query. Performs Self Test and returns 0 if passed; any non-zero number returned indicates Self Test failed.

**SEE ALSO**
Individual Self Test Commands (Appendix C)
**WAI**

**USE**
To pause command execution until previous operations are complete.

**SYNTAX**
*WAI

**ACTION**
Stops command execution until all pending operations are complete.

Use *WAI after SCREEN and SETUP commands and other commands involving routing changes or lengthy measurement times.

**SEE ALSO**
DELAY

**EXAMPLE**
SCREEN:REC
*WAI
// Displays the Receiver Screen.
// Stops macro execution until
// the screen display is completed.

RECEIVE:INPUT:TR
*WAI
// Selects T/R Connector for Receiver Input.
// Stops command execution until
// the routing change is completed.

---

**ABS**

**USE**
Absolute value.

**SYNTAX**
ABS(n)

**ELEMENT** | **DESCRIPTION**
---|---
$n$ | Any constant, variable, number or any expression that evaluates to a number.

**ACTION**
Returns the absolute value of $n$.

**SEE ALSO**
FLOOR, SIGN

**EXAMPLES**
y=ABS(-4) // $y$ is assigned the value of positive 4.
Z=-7
POS=ABS(Z) // POS is assigned the value of positive 7.
ACTIVATE

USE
To activate a task for multitasking.

SYNTAX
ACTIVATE "name"

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the task to be loaded into the schedule queue.</td>
</tr>
</tbody>
</table>

ACTION
Places a declared task on the schedule queue so the task can be multitasked.

A macro must be declared a task using the TASK command before the macro can be put in the schedule queue using the ACTIVATE command. This must be done to each macro prior to being multitasked. Once execution of the macro begins, execution starts from the beginning of the macro.

SEE ALSO
Multitasking Macros (2-12), TASK, TPAUSE, TSTOP

EXAMPLES
See Multitasking examples (2-12).

AND

USE
Logical AND.

SYNTAX
cmdA argument1 AND argument2 cmdB or
cmdA argument1 AND argument2 AND argument3 ... AND conditionX cmdB

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdA</td>
<td>Any decision point command (IF, ELIF, UNTIL, WHILE).</td>
</tr>
<tr>
<td>argument1</td>
<td>Any expression that evaluates to TRUE, FALSE, 0 (zero) or any non-zero number (0 = FALSE, any non-zero number = TRUE).</td>
</tr>
<tr>
<td>argument2 on</td>
<td>Same as argument1.</td>
</tr>
<tr>
<td>cmdB</td>
<td>Any valid TMAC command.</td>
</tr>
</tbody>
</table>

ACTION
Evaluates argument1 and argument2 and, if both are TRUE (or evaluated to a non-zero number), then the result of the AND operation is TRUE (FALSE, otherwise). If more than one AND operation in encountered then the AND operations are evaluated from left to right. Successive ANDs evaluate the result of previous AND operations along with individual expressions.
Table 3-4 lists the expression combinations and the AND operation results.

<table>
<thead>
<tr>
<th>argument1</th>
<th>argument2</th>
<th>AND Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>

Table 3-4 AND Operation Results

SEE ALSO  !, OR, =, !=, <, >, <=, >=, DO UNTIL, IF ..., WHILE

EXAMPLES

```plaintext
var x=6, z=5
if x=6 AND x!=z  // Decision point command utilizing AND
    // as condition with accompanying
    // expressions to evaluate. Condition is
    // TRUE, because both expressions evaluate
    // to TRUE.
    print y
    // Performs resulting command since condition
    // is TRUE.
ENDIF
```

ASC

USE ASCII value of character.

SYNTAX ASC(string)

ELEMENT DESCRIPTION

string Any string.

ACTION Returns the ASCII value of the first character of string.

SEE ALSO CHR, LEN, STR, STRPOS, VAL

EXAMPLES VAR Ascii_value
STRING Char
Char="Example"
Ascii_value = ASC(Char) // Ascii_value, a numeric variable,
    // is assigned the ASCII value
    // of E (67).
BCOLOR

USE To change background color.

SYNTAX BCOLOR b

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Color number (see Table 2-2). Range of colors is color numbers 0 to 7.</td>
</tr>
</tbody>
</table>

ACTION Changes the background color to the color b.

Can use color name constants defined in Appendix A.

SEE ALSO COLOR, EDIT:COLOR commands

BEGIN

USE To allow macros to contain multiple lines.

SYNTAX *DMC "name", BEGIN
   sequence
END

ACTION Combines with the END command to define the command lines in multiple line macros. The BEGIN command is not used with single line macros.

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the macro being defined.</td>
</tr>
<tr>
<td>sequence</td>
<td>Command sequence performed inside the macro.</td>
</tr>
</tbody>
</table>

SEE ALSO Macros (2-9), *DMC, END

EXAMPLES

*DMC "MPtr",VAR Mid;Mid=(Freq1+Freq2)/2;NVSAV Mid,60;PRINT Mid

The above macro can be entered as:

VAR Mid
*DMC "MPtr",BEGIN
Mid=(Freq1+Freq2)/2
NVSAV Mid,60
PRINT Mid
END
**BOX**

**USE**
To create a user defined box on the color display.

**SYNTAX**
BOX \( f, x_1, y_1, x_2, y_2, c \)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f )</td>
<td>Creates the box solid or hollow. ( f ) is 1 for a solid box. ( f ) is 0 for the box to appear hollow.</td>
</tr>
<tr>
<td>( x_1, y_1 )</td>
<td>Point signifying the top left corner of the box (0,0 to 639,349).</td>
</tr>
<tr>
<td>( x_2, y_2 )</td>
<td>Point signifying the bottom right corner of the box (0,0 to 639,349).</td>
</tr>
<tr>
<td>( c )</td>
<td>Number (or constant name - see Appendix A) specifying the color of the box. See Table 2-2 for the available colors.</td>
</tr>
</tbody>
</table>

**ACTION**
Creates a box using the points specified as the top left and bottom right corners. The box can be created hollow or solid.

Text cannot be printed in boxes as in windows.

**SEE ALSO**
Graphics (2-13-5)

**EXAMPLES**
BOX \( 1, 150, 50, 200, 250, 7 \)

The above command displays a light gray box on the color display as follows:

![Figure 3-1 Box Example](image)

Figure 3-1 Box Example
**CALCulate?**

**USE**  
To calculate a function and return the result to the Host.

**SYNTAX**  
`CALCulate? argument`

**ELEMENT** | **DESCRIPTION**
---|---
`argument` | Mathematical expression or formula calculated.

**ACTION**  
The `argument` is calculated and returned through the active interface connector.

This command cannot be used inside a macro.

**SEE ALSO**  
Mathematical Functions (2-7)

**EXAMPLES**

- `X=4, Y=3`  
  // Initializes variables.

- `CALC? X*Y**3`  
  // Calculates the mathematical expression and returns the result to the Host. 108 is returned.

---

**CASE OF OTHERWISE ENDCASE**

**USE**  
To provide a predetermined decision point inside a macro.

**SYNTAX**  
`CASE variable
  OF value1: command
  OF value2: command
  .
  .
  .
  OTHERWISE: command
ENDCASE`

**ELEMENT** | **DESCRIPTION**
---|---
`variable` | Expression compared with the values to decide which `OF` command to perform.

`value` | Expression compared with `variable`. If equal to `variable`, the command following `value` is performed.

`command` | Command performed if `value` is equal to `variable`.

**ACTION**  
Utilizes value of variable to select one of multiple possible preselected commands. There is no limit to the number of `OF` expressions, and the `OTHERWISE` expression is optional. The `OF` values are compared to the `CASE` variable. The `command` sequence following the `OF` value that equals the `CASE` variable is executed. If none of the `OF` values equal the `CASE` variable, the `OTHERWISE command`, if present, is executed, then the macro execution passes to the command following the `ENDCASE` command.

**SEE ALSO**  
Macro Decision Points (2-10), IF, IF ELSE, IF ELIF
EXAMPLES

*DMC "Val"
SCREEN:USER  // Selects the blank user screen.
CASE X
  OF 1: PRINT "X=1"
  OF 2: PRINT "X=2"
  OF 5: PRINT "X=5"
  OTHERWISE: PRINT "X unknown"
ENDCASE

/* X is compared to the OF values 1, 2 and 5. If X is equal
to one of the OF values, the command or commands following
that OF value are executed. For example: if X=2, the PRINT
"X=2" command is executed, if X=6, the PRINT "X unknown"
command is executed. */

*DMC "Menu", BEGIN
SCREEN:USER
VAR Choice
PRINT "Press a DATA ENTRY Key to select a Test"
PRINT "  and press the ENTER Key."
PRINT " 1. Test 1"
PRINT " 2. Test 2"
PRINT " 3. Test 3"
PRINT "Any other selection other than the ones listed"
PRINT "  displays the Receiver Operation Screen."
XY 100,250
INPUT Choice
CASE Choice
  OF 1: PRINT "Test_1 is running"
        DELAY 3000
        CLS
        Test_1
  OF 2: PRINT "Test_2 is running"
        DELAY 3000
        CLS
        Test_2
  OF 3: PRINT "Test_3 is running"
        DELAY 3000
        CLS
        Test_3
  OTHERWISE: PRINT "No Test Performed"
ENDCASE
SCREEN:REC
*WAI  // Pauses macro execution until screen
      // command is fully executed.

(Example is continued on next page.)
/* The macro, "Menu," displays a user-created menu allowing the selection of one of three tests. These tests are actually macros Test_1, Test_2 and Test_3 which are executed if 1, 2 or 3 is entered in response to the INPUT command. If the entered data is not 1, 2 or 3, the OTHERWISE command is performed and "No Test Performed" is printed. After a Test is executed or the PRINT command is performed, command execution resumes following the ENDCASE command. */

### CENTER (Sp Tst Only)

**USE**
To display text centered within a defined display area on the screen of the HOST.

**SYNTAX**

```CENTER text,x,y,length```

**ELEMENT** | **DESCRIPTION**
---|---
`text` | String of characters and spaces designated within quotation marks.
`x` | Horizontal starting point in pixels of horizontal display area (0 to 639).
`y` | Vertical starting point in pixels of horizontal display area (0 to 259).
`width` | Width in pixels of horizontal display area.

**ACTION**
Displays `text` centered within a horizontal display area. The starting and ending positions are defined by `x`, `y` and `width`.

**SEE ALSO**
HPRINT, LJPRINT, PRINT, RJPRINT, XYPRINT

**EXAMPLES**

```USER
CENTER "Test No. 12",0,0,639 // Displays Test No. 12
```
CHR

USE
Returns the character equivalent of ASCII number.

SYNTAX
CHR(n)

ELEMENT DESCRIPTION
n Any constant, variable, number or any expression that evaluates to a number. Range is from 0 to 255.

ACTION
Returns the character equivalent of ASCII number n.

SEE ALSO
STR, VAL

EXAMPLES
*pmc
*dmc 'ascii_test',begin
user
for n=0 to 255
$=chr(n)
// $ is assigned ASCII character associated with the present value of i.
print $," ",n
// Prints current ASCII character and value of n.
for j= 1 to 1000;next j
// Time delay.
next n
end

CLS

USE
To erase the contents of a window or the color display.

SYNTAX
CLS

ACTION
Clears the currently selected window. To clear the entire color display, select window 0 (the background) using the WSEL command.

SEE ALSO
WSEL
COLOR

USE
To specify or return foreground text color.

SYNTAX
COLOR f or
COLOR f,b or
COLOR?

ELEMENT DESCRIPTION
f Foreground text color (see Table 2-2).
b Optional background contrast color (see Table 2-2).

ACTION
Specifies or returns current foreground text color. An optional background contrast color may be included. If excluded, the present background color is assumed.

The actual color of text printed on the screen of the HOST is determined by the following formula:

Actual Foreground Text color value = ([selected foreground text color (f) xor'ed with selected or assumed background contrast color (b)]) xor'ed with current background color (see BCOLOR)]. See Table 2-2 for colors and color values.

SEE ALSO
BCOLOR, EDIT:COLOR commands

EXAMPLES
screen:user // Displays blank user screen (for Sp Tst
 // use USER).
cls // Initializes screen.
bcolor 1 // Selects Dark Blue for background.
color yellow,1 // Using predefined constant, selects
 // Yellow as foreground text color and
 // Dark Blue as background contrast color
 // (could have used dark_blue predefined
 // constant).
print "This is Yellow!" // Print Yellow text on Dark_Blue
 // screen (for Sp Tst use HPRINT).
**CONST**

**USE**
To define constants.

**SYNTAX**
CONSt name, argument or
CONSt name, argument, name, argument, ... ,name, argument

**ELEMENT** | **DESCRIPTION**
--- | ---
name | Name of the constant defined. The first character of the name is a letter while the remaining characters can be letters, digits and the underscore. The length of name must be from 2 to 31 characters.

argument | Any number or any expression that evaluates to a number.

**ACTION**
Declares constants and associated values. Constants are global and cannot be declared inside a macro. String data cannot be assigned to a constant. Constant values can only be changed using another CONSt command.

**SEE ALSO**
Constants and Data Arrays (2-4), DATA, NVRCL, NVSAV

**EXAMPLES**
CONSt PI,3.1415,Wid,X/4.3 // Defines PI as a constant and // gives it a value of 3.1415. // Defines Wid as a constant, // evaluates X/4.3 and assigns this // value to Wid.

---

**COS**

**USE**
To return the cosine of an angle.

**SYNTAX**
COS(n)

**ELEMENT** | **DESCRIPTION**
--- | ---

n | Any constant, variable, number or any expression that evaluates to a number.

**ACTION**
Returns the cosine of an angle (n) which is measured in radians.

**SEE ALSO**
SIN

**EXAMPLES**
VAR Cosine_of_radians
Cosine_of_radians=COS(5*3.1415/6) // Cosine_of_radians now // holds the cosine of 5π/6 // (≈-0.866).
DATA

USE
To define constant data arrays.

SYNTAX
DATA name={argument, argument, ..., argument}

ELEMENT  DESCRIPTION
name       Name of the constant defined. The first character of the name is a letter while the remaining characters can be letters, digits and the underscore. The length of the name must be from 2 to 31 characters.

argument   Any number or any expression that evaluates to a number.

The first argument is assigned to the first index (name[0]) and so on.

ACTION
Declares data arrays and associated values. A data array is an array of constants. Data arrays are global and cannot be declared inside a macro. String data cannot be assigned to a data array. Data arrays values can only be changed using another DATA command. Data array values are defined by including the order of occurrence in braces (curly brackets). Consequently, a value in the data array is retrieved by specifying the position of the data array value with an index in brackets following the name. This order begins with 0 so that Stat[3] returns the fourth constant in the data array Stat.

SEE ALSO
Constants and Data Arrays (2-4), CONST, ICON, NVRCL, NVSAV

EXAMPLES
DATA STAT=(2, 34.7, 2.874836593, 336/4)
X=STAT[1]     // Assigns 34.7 to X.
Z=STAT[0]     // Assigns 2 to Z.

DELAY

USE
To delay command execution.

SYNTAX
DELAY t

ELEMENT  DESCRIPTION

ACTION
Provides a time delay of length t in ms before the next command is executed. Delays are used to allow physical changes enough time to occur before command execution continues. Such changes could be: code generation, routing changes and displaying Operation Screens.

SEE ALSO
Macros (2-9), *WAI
DO UNTIL

USE
To perform a set of commands repeatedly until a specified condition occurs (becomes true).

SYNTAX
DO
 sequence
 UNTIL condition

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>condition</td>
<td>Conditional expression evaluated.</td>
</tr>
<tr>
<td>sequence</td>
<td>One or more valid commands or expressions.</td>
</tr>
</tbody>
</table>

ACTION
The DO loop performs sequence once, then evaluates condition. If condition is FALSE, then command execution returns (loops) to the beginning of the DO loop to again perform sequence and so on. If condition is evaluated TRUE, then command execution passes on to the following command, leaving the DO loop.

If the condition does not eventually become true, the DO loop continues to loop until power is cycled.

SEE ALSO
FOR, WHILE. For SYSTEM Commands, see 2-14.

EXAMPLES
*DMC "Count",BEGIN  // Defines a macro named Count.
N=0  // Assigns 0 to N.
DO  // Starts DO loop.
   N = N + 1  // Increments N by 1.
   PRINT N  // Prints current value of N.
   UNTIL N=2  // Loops back to first command after DO
    // command until N equals 2.
END  // End of macro Count.

(Examples continued on next page.)
(Examples continued from previous page.)

*DMC "Compute", RETURN(FLOOR((50-$1)/4+50))
   // Defines a function named Compute
   // with $1 as placeholder to pass a
   // value into the function.
*DMC "Compute_x", BEGIN
   // Defines a macro named Compute_x.
X = 6
   // Assigns 6 to X.
DO
   // Starts DO loop.
   X = Compute X
   // Assigns X the result of the
   // function Compute. The old value
   // of X is passed into function.
PRINT X
   // Prints new X.
UNTIL X=50
   // Loops back to first command after
   // the DO command until X equals 50.
END
   // End of macro Compute_x.

/* The following is a description of the above macro:

  o X is set to 6, then enters the DO loop.

  o The Compute function is calculated and sets X equal
to 61. X is printed.

  o The condition (X = 50) is evaluated. The condition is
FALSE; command execution loops to first command after
DO command.

  o The Compute function is calculated and sets X equal
to 47. X is printed.

  o The condition (X = 50) is evaluated. The condition is
FALSE; command execution, again, loops to first command
after DO command.

  o The Compute function is calculated and sets X equal
to 50. X is printed.

  o The condition (X = 50) is evaluated. The condition is
TRUE; command execution passes on to the command after
the UNTIL, leaving the DO loop. Macro execution ends. */

/* The conditional expression does not have to involve a
variable manipulated inside the loop. */

DO
   commands
   .
   .
UNTIL SYSTEM:KEY? > 0
   // Continues looping until a Front
   // Panel Key is pressed.
**DRAW**

**USE**
To create a user-defined line on the color display.

**SYNTAX**
```
DRAW x1,y1,x2,y2,c
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1,y1</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349.</td>
</tr>
<tr>
<td>x2,y2</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349.</td>
</tr>
<tr>
<td>c</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string. Number specifying color of line. See Table 2-2 for colors and associated values.</td>
</tr>
</tbody>
</table>

**ACTION**
Creates a line of color c starting at point x1,y1 and ending at point x2,y2.

**SEE ALSO**
Graphics (2-13-5), BOX, ELLIPSE, ICON

**EXAMPLES**
See Graphic examples (2-13-5).

---

**EDIT:COLOR:LETTER (HOST Only)**

**USE**
To change or return the color of text of all built-in menus.

**SYNTAX**
```
EDIT:COLOR:LETTER c or
EDIT:COLOR:LETTER?
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string. Color or numerical value of menu text color (see Table 2-2).</td>
</tr>
</tbody>
</table>

**ACTION**
Changes or returns the current color of text characters of all built-in menus of the HOST.

**SEE ALSO**
Other EDIT:COLOR commands, EDIT:WIDTH

**EXAMPLES**
```
edit:color:letter 15 // Changes the text color of all // built-in menus to White.
```
EDIT:COLOR:MENU (HOST Only)

USE
To change or return background color of all built-in menus.

SYNTAX
EDIT:COLOR:MENU c or
EDIT:COLOR:MENU?

ELEMENT DESCRIPTION

* c  Any constant, variable, number, string or any expression that evaluates to a number or string. Color or numerical value of menu background color (see Table 2-2).

ACTION
Changes or returns the current background color of all built-in menus of the HOST.

SEE ALSO
Other EDIT:COLOR commands, EDIT:WIDTH

EXAMPLES
edit:color:menu 6  // Changes the background color of all // built-in menus to Brown.

EDIT:COLOR:SOFT:BOX (HOST Only)

USE
To change or return background color of all built-in Soft Function Keys.

SYNTAX
EDIT:COLOR:SOFT:BOX c or
EDIT:COLOR:SOFT:BOX?

ELEMENT DESCRIPTION

* c  Any constant, variable, number, string or any expression that evaluates to a number or string. Color or numerical value of Soft Function Key background color (see Table 2-2).

ACTION
Changes or returns current background color of all built-in Soft Function Keys of the HOST.

SEE ALSO
Other EDIT:COLOR commands, EDIT:WIDTH

EXAMPLES
EDIT:COLOR:SOFT:LETTER (HOST Only)

USE To change or return the text color of all built-in Soft Function Keys.

SYNTAX

EDIT:COLOR:SOFT:LETTER c or
EDIT:COLOR:SOFT:LETTER?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string. Color or numerical value of Soft Function Key text color (see Table 2-2).</td>
</tr>
</tbody>
</table>

ACTION Changes or returns current color of the text of all built-in Soft Function Keys of the HOST.

SEE ALSO Other EDIT:COLOR commands, EDIT:WIDTH

EXAMPLES


EDIT:COLOR:SOFT:SELECT (HOST Only)

USE To change or return the text color of all built-in Soft Function Keys when selected.

SYNTAX

EDIT:COLOR:SOFT:SELECT c or
EDIT:COLOR:SOFT:SELECT?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string. Color or numerical value of selected Soft Function Key text color (see Table 2-2).</td>
</tr>
</tbody>
</table>

ACTION Changes or returns current text color of built-in Soft Function Keys when selected.

SEE ALSO Other EDIT:COLOR commands, EDIT:WIDTH

EXAMPLES

edit:color:soft:select 4  // Changes the color of the text (when selected) of all built-in Soft Function Keys to Dark Red.
EDIT:WIDTH (HOST Only)

USE
To specify the width of the input box that appears with the INPUT command.

SYNTAX
EDIT:WIDTH n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number, or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

ACTION
Specifies the width in n pixels of the input box that appears on the screen of the HOST when the INPUT command is executed.

Use a factor of 19 for each numerical character (including the decimal point) expected in input box; use a factor of 26 for each alphanumeric (string) character expected in input box.

SEE ALSO
EDIT:COLOR commands, INPUT

EXAMPLES
edit:width 38 // Changes the width of the input box on the screen in response to a INPUT command to accept two numeric characters.
edit:width 78 // Sizes input box to accept 3 alphanumeric (string) characters.

ELLIPSE

USE
To draw an ellipse

SYNTAX
ELLIPSE b,x,y,r,a,c

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Any constant, variable, number, or any expression that evaluates to a number. Range is 0 to 1.</td>
</tr>
<tr>
<td>x,y</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349.</td>
</tr>
<tr>
<td>r</td>
<td>Any constant, variable, number, or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>a</td>
<td>Any constant, variable, number, or any expression that evaluates to a number. Range is 0.1 to -30.</td>
</tr>
<tr>
<td>c</td>
<td>Any constant, variable, number, string or any expression that evaluates to a number or string. Color name or associated number (see Table 2-2).</td>
</tr>
</tbody>
</table>
**ACTION**

Draws an ellipse centered at \(x, y\) with radius \(r\) in pixels and aspect ratio \(a\). Ellipse is drawn in color selected for \(c\). If \(b\) is 1, the ellipse is solid. If \(b\) is 0, the ellipse is hollow. An aspect ratio of 0.75 creates a circle. An aspect ratio > 0.75 creates an ellipse that has greater height than width, and an aspect ratio < 0.75 creates an ellipse that has greater width than height. Parameters \(x, y\) and \(r\) are measured in pixels.

The height and width of any ellipse created by this command are found using the following equations:

<table>
<thead>
<tr>
<th>WHERE:</th>
<th>WIDTH</th>
<th>HEIGHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1 (\leq a \leq 1)</td>
<td>2(r)</td>
<td>8(ra/3)</td>
</tr>
<tr>
<td>(a &gt; 1)</td>
<td>2(r/a)</td>
<td>8(r/3)</td>
</tr>
</tbody>
</table>

Due to the pixel alignment, an ellipse which has a height to width ratio of 4:3 (or a height 1/3 greater than the width) has an aspect ratio equal to 1 \((a = 1)\).

Table 3-5 Ellipse Width to Height Equations

**SEE ALSO** Graphics (2-13-5), BOX, DRAW, ICON

**EXAMPLES**

```
ellipse 1,320,175,50,0.375,dark_green // Creates a solid green
    // ellipse that is
    // 50 pixels high by
    // 100 pixels wide in
    // the center of the
    // screen.

ellipse 0,320,175,75,1.5,dark_red  // Creates a hollow red
    // ellipse that is 200 pixels
    // high by 100 pixels wide
    // in the center of the
    // screen.
```
**END**

**USE**
To allow a macro to contain multiple lines.

**SYNTAX**
*DMC "name", BEGIN
  sequence
END

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of macro being defined.</td>
</tr>
<tr>
<td>sequence</td>
<td>Command sequence performed inside macro.</td>
</tr>
</tbody>
</table>

**ACTION**
When used with BEGIN, the END command defines the end of a multiple line macro.

**SEE ALSO**
Macros (2-9), *DMC, BEGIN

**EXAMPLES**
*dmc "Print_x",BEGIN
  INPUT X
  PRINT X
END

// Defines a macro named Print_x.
// Receives a value from the keyboard.
// Prints the received value.
// End of the macro Print_x.

**ERASE:TEXT (Sp Tst Only)**

**USE**
To erase a defined display area.

**SYNTAX**
ERASE:TEXT x,y, width

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>x,y</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349.</td>
</tr>
<tr>
<td>width</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 1 to 640</td>
</tr>
</tbody>
</table>

**ACTION**
Erases display area defined by an x,y starting point and width in pixels.
**EXEC**

**USE**
To execute a macro at a specified address.

**SYNTAX**
EXEC address or
EXEC &name

**ELEMENT** | **DESCRIPTION**
---|---
address | Any constant, variable, number or any expression that evaluates to a number.
name | Name of the macro to be executed.

**ACTION**
Executes macro name located at the address specified. An & (ampersand) before the name of the macro, as shown, returns the address of the macro. Macros requiring variables to be passed when the macro's name is entered cannot be executed using the EXEC command. Unpredictable results occur when using an invalid address.

The EXEC command can be used to call out undefined macros by simply declaring a variable and using the variable as the address.

**SEE ALSO**
Macros (2-9)

**EXAMPLES**
ADD=&First
   // The address of the macro First is loaded into the variable ADD.
EXEC ADD
   // The macro located at the address loaded in the variable ADD is executed.
EXEC &First
   // Macro First is executed.

---

**EXP**

**USE**
To calculate the result of \( e^n \)

**SYNTAX**
EXP(n)

**ELEMENT** | **DESCRIPTION**
---|---
n | Any constant, variable, number or any expression that evaluates to a number. Range is from -102.585781 to 88.72283554.

**ACTION**
Returns the result of \( e^n \).

**SEE ALSO**
LN, LOG

**EXAMPLES**
VAR Expo
x=1.57
Expo=EXP(x)
   // \( e \) to the 1.57th power (4.806648) is assigned to Expo.
FALSE

USE  To produce 0.
SYNTAX  FALSE or OFF
ACTION  Provides a concise method of specifying a logical 0.
SEE ALSO  I
EXAMPLES  y=OFF // y is assigned the value of 0.

FLOOR

USE  To return the truncated value of n.
SYNTAX  FLOOR(n)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

ACTION  Returns the truncated value of n, leaving only the integer portion to the left of the decimal point.
SEE ALSO  ABS, SIGN
EXAMPLES  VAR Round_off
          Round_off = FLOOR(5.99) // Assigns the truncated value // of 5.99 (5) to Round_off.
FLUSH

USE
To return pending responses.

SYNTAX
FLUSH

ACTION
The FLUSH command returns all pending return data upon execution.

Normally, without the FLUSH command, returning data from the queries in a macro is collected in the output queue of the Test Set. The contents of the output queue is returned at the end of any loops present, at the end of the macro or when the queue becomes full (the output queue is 2 kilobytes in size).

SEE ALSO
Macros (2-9)

EXAMPLES
*DMC "Without_FLUSH",BEGIN // Defines a macro named
  // Without_FLUSH.
SCREEN:REC // Displays the Receiver Operation
  // Screen.
*WAI // Waits for the Receiver Operation
  // Screen to be completely displayed.
REC:FREQ? // Queries the Receiver Frequency.
M_RF? // Queries the Frequency Error Meter
  // reading.
M_PWR? // Queries the Power Meter reading.
END // Ends macro Without_FLUSH.

/* The returned data in the above macro without using the FLUSH command appears as follows:
100320,100318,62 */

*DMC "With_FLUSH",BEGIN // Defines a macro named With_FLUSH.
SCREEN:REC // Displays the Receiver Operation
  // Screen.
*WAI // Waits for the Receiver Operation
  // Screen to be completely displayed.
REC:FREQ? // Queries the Receiver Frequency.
FLUSH // Returns impending responses.
M_RF? // Queries the Frequency Error Meter
  // reading.
FLUSH // Returns impending responses.
M_PWR? // Queries the Power Meter reading.
END // Ends macro With_FLUSH.

/* The returned data in the above macro _with_ the FLUSH commands appears as follows:
100320
100318
62 */
**FOR NEXT**

**USE**
To perform a set of commands in a loop a given number of times within a macro.

**SYNTAX**
```
FOR variable = initial TO ending STEP step
    sequence
NEXT variable
```

**ELEMENT** | **DESCRIPTION**
---|---
variable | Any variable.
initial | Any constant, variable or number or any expression that evaluates to a number.
ending | Any constant, variable or number or any expression that evaluates to a number.
step | Any constant, variable or number or any expression that evaluates to a number.
sequence | Any valid single command or command combination.

**ACTION**
Performs a command sequence the number of times required to increment variable, an amount equal to step, from an initial to an ending value.

The following steps are performed:
1. FOR command sets variable equal to initial.
2. Command sequence is executed.
3. NEXT command increments variable by the value step.
4. The value of variable is evaluated:
   a. If step is a positive number and the current value of variable ≤ ending, macro execution returns to Step 2 above.
   b. If step is a positive number and the current value of variable is greater than ending, macro execution continues with the command following the NEXT command, leaving the FOR .. NEXT loop.
   c. If step is a negative number and the current value of variable ≥ ending, macro execution returns to Step 2 above.
   d. If step is a negative number and the current value of variable is less than ending, macro execution continues with the command following the NEXT command, leaving the FOR .. NEXT loop.
   e. If step increments variable such that the difference between ending and variable is greater than when assigned the value of initial, macro execution continues with the command following the NEXT command, leaving the FOR .. NEXT loop.

If optional STEP and step are left off, the default step is +1 (even if initial and/or ending are negative).

Setting step equal to 0, causes a FOR .. NEXT loop to loop indefinitely and requires cycling power of the Test Set to stop execution of the loop.
DO, WHILE

for N=1 to 3
    command1
next N
/* commands following */

/* The following sequence occurs:
   o N is set equal to 1.
   o Command1 is executed.
   o N is incremented by 1 (N=2) (by default, N is incremented by +1 when step is omitted) and compared to ending (3).
   o Command1 is executed.
   o N is incremented by 1 (N=3) and compared to ending (3).
   o Command1 is executed.
   o N is incremented by 1 (N=4) and compared to ending (3).
   o Macro execution exits FOR .. NEXT loop and proceeds with the commands following the NEXT command. */

for A=10 to 4 step -3
    print A
next A
print 2*A

/* The following sequence occurs:
   o A is set equal to 10.
   o A is printed (10).
   o -3 is added to A (A = 7) and A is compared to ending (4).
   o A is printed (7).
   o -3 is added to A (A = 4) and A is compared to ending (4).
   o A is printed (4).
   o -3 is added to A (A = 1) and A is compared to ending (4).
   o 2*A is printed (2) as command execution proceeds below the NEXT command. */

(Examples continued on next page.)
(Examples continued from previous page.)

FOR MM = -1 TO -6 STEP -2
   PRINT MM
NEXT MM

/* -1, -3 and -5 is printed. When MM is incremented to -7,
   macro execution exits the FOR ... NEXT loop, because -7 is
   beyond the ending value. */

FOR N=1 TO 4 STEP -1
   PRINT N
NEXT N

/* N is set equal to 1 and printed once. N is then incre-
   mented by -1 and is now equal to 0. Because the difference
   between ending and N (4 - 0 = 4) is greater than when N was
   assigned the value of initial (4 - 1 = 3), command execution
   proceeds below the NEXT command. */

FORGET

USE
To delete a macro from memory.

SYNTAX
FORGET name

ELEMENT DESCRIPTION
name String variable or literal string surrounded by quotation
marks (e.g., "macro_1a").

ACTION
Deletes the macro specified by name.

All macros and variables declared after the specified macro was declared are also deleted.

SEE ALSO
*PMC

EXAMPLES
*DMC "FIRST",BEGIN
   PRINT "first macro"
END
*DMC "SECOND",BEGIN
   PRINT "second macro"
END
VAR TO_BE_ERASED
FORGET 'SECOND'
   // Variable TO_BE_ERASED is declared.
   // Macro SECOND and variable
   // TO_BE_ERASED are deleted from
   // memory. Macro FIRST still resides
   // in memory.
FORMat

USE To specify the form of the returned data for queries.

SYNTAX
FORMat BINary or
FORMat OCTal or
FORMat HEXadecimal or
FORMat ASCII (HOST Only) or
FORMat DECimal (Sp Tst Only)

ACTION Specifies form of the returned data for queries. The four forms are Binary (base 2), Octal (base 8), Hexadecimal (base 16) and ASCII (HOST) or Decimal (Sp Tst) (base 10). Default is base 10. For the HOST, all returned data (other than that in ASCII format) is preceded with a prefix signifying the format: #B for Binary, #Q for Octal and #H for Hexadecimal.

Data in scientific notation is unaffected by these commands.

SEE ALSO Numeric Formats (2-5-2)

EXAMPLES
REC:FREQ 103.7 MHZ // Sets Receiver Frequency to 103.7 MHz.
FORM BIN // Selects binary format for the
// returned data.
REC:FREQ? // Queries the Receiver Frequency.
// #81100101001000010100 is returned.
FORM HEX // Selects Hexadecimal format for
// the returned data.
REC:FREQ? // Queries the Receiver Frequency.
// #H19514 is returned.

GPIB:ADDRess (Sp Tst Only)

USE To specify the GPIB address.

SYNTAX GPIB:ADDRess address

ELEMENT DESCRIPTION
address Any constant, variable, integer or any expression that evaluates to a integer. Range is 0 to 31.

ACTION Specifies the GPIB address.

SEE ALSO GPIB:MASK, GPIB:SRQ

EXAMPLES See GPIB:SRQ.
## GPIB:MASK (Sp Tst Only)

**USE** 
To set the SRQ interrupt mask.

**SYNTAX** 
GPIB:MASK n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. 8 bit value with a range of 0 to 255.</td>
</tr>
</tbody>
</table>

**ACTION** 
Sets SRQ interrupt mask. The corresponding user-defined Status Byte bit is masked unless the mask bit is active (i.e., set to 1).

**SEE ALSO** 
GPIB:ADDRess, GPIB:SRQ

**EXAMPLES** 
See GPIB:SRQ.

## GPIB:SRQ (Sp Tst Only)

**USE** 
To enable a user-defined Status Byte bit to trigger a Service Request.

**SYNTAX** 
GPIB:SRQ n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. 8 bit value with a range of 0 to 255.</td>
</tr>
</tbody>
</table>

**ACTION** 
Sets a user-defined Status Byte bit to trigger a Service Request. The corresponding mask bit must be active to generate an SRQ.

The GPIB:SRQ command is used in decision point commands (IF, WHEN, DO .. UNTIL, etc.) to trigger a Service Request when a user-defined event occurs (e.g., cellular phone message capture).

**SEE ALSO** 
GPIB:ADDRess, GPIB:MASK

**EXAMPLES** 
GPIB:ADDR 30;MASK 32 // Set address and unmask bit 5.
FOCC:SET:CHAN 333;STAR // Set up and monitor FOCC channel 333.
FOCC:CAPT:MIN '316/522-4981' // Set to capture specified MIN.
IF FOCC:CAPT? =1 GPIB:SRQ 32;ENDIP // If capture occurs, send SRQ.
HEIGHT

USE
To change height of text printed on color display.

SYNTAX
HEIGHT n

DESCRIPTION

n
Any constant, variable, number or any expression that evaluates to a number. Range is 1 to 4.

ACTION
The height of text printed on the color display after this command is changed to n times the standard height (0.175 inches). Text can be printed in four heights: 0.175, 0.35, 0.525 or 0.7 inches.

SEE ALSO
Graphics (2-13-5), PRINT, ROTATE

EXAMPLES
HEIGHT 1 // Sets the text height to 0.175''.
PRINT "standard" // PRINT command.
PRINT // Each PRINT command moves the xy position down a line the current position set by the HEIGHT command (currently 0.175'').
PRINT
PRINT // Sets the text height to 0.7''.
PRINT "large" // PRINT command.
HEIGHT 2 // Sets the text height to 0.35''.
PRINT "double standard" // PRINT command.
HEIGHT 1 // Returns the HEIGHT setting to 1.

The following is printed on the color display: 1) standard in 0.175 inch high text, 2) large in 0.7 inch high text and 3) double standard in 0.35 inch high text.

Each PRINT command without an ending comma moves the xy position to the beginning of the next line using the height setting for the height of the line. The first four PRINT commands move the xy position down 0.7'' (0.175'' for each PRINT command). The PRINT "large" command moves the xy position down 0.7 inches. The PRINT "double standard" command moves the xy position down 0.35 inches.

HFLUSH (Sp Tst Only)

USE
To quickly send HOST commands through the internal SCSI Bus from the Sp Tst to HOST.

SYNTAX
HFLUSH

ACTION
Gives previous HOST commands priority by flushing all commands currently in the task queue out the SCSI at that time. (Normally, to avoid time consuming SCSI traffic, HOST commands are held in the task queue. When the task queue becomes full, commands are sent across the internal SCSI Bus to the HOST.)
EXAMPLES

host "fgen:gen1:mod1 5;shape:sin"
host "fgen:gen1:freq 1000;mod:fm
hflush

HPRINT (Sp Tst Only)

USE

To print on the display screen.

SYNTAX

HPRINT argument or
HPRINT argument, argument, ... , argument, or
HPRINT %0nl, argument, ...... ,argument, or
HPRINT argument, ...... ,%0nl, argument.

ELEMENT  DESCRIPTION

argument  Expression to be printed. Can be a mathematical
expression, a literal string or the contents of variables.
Mathematical expressions are calculated and string
functions are performed before the results are printed.

%  Indicates that formatting information follows which pertains
to the arguments that follows. Must be followed by an
associated argument. Must precede any combination
of 0, n and l.

0  Zero signifies leading zeros are added to fill the field width.
0 is omitted for no leading zeros. When used, must be
accompanied by n. Applies only to numeric expressions.

n  Specifies field width. n is the minimum number of digit
spaces (decimal point counts as one). Applies only to
numeric expressions.
If l is D, n can be in the form of b.a, where b designates
minimum number of total digit spaces (decimal point counts
as one) and a designates the maximum number of digits after
the decimal point.

l  Specifies the numeric base the data is printed in. Applies
only to numeric expressions. Omit for a default of Signed
Decimal or enter one of the following:

B  Binary
Q  Octal
H  Hexadecimal
U  Unsigned decimal
D  Signed decimal

ACTION

Prints argument on the display screen. The HPRINT command (HOST PRINT
command) prints numerical and string values at the current xy location on the
color display. Several values are printed on the same line by separating them
with commas. If the optional ending comma is used, the next HPRINT
command continues on the same line; otherwise a new line is started. If an
argument is a variable, the value of the variable is printed. Mathematical
expressions are calculated before printing.
%0nl contains the format settings for numeric expressions and can be placed anywhere in the HPRINT command. There can be several format settings in each HPRINT command, each one changing the leading zero format, the numeric base and the field width for the expressions that follow. Format settings do not affect the printing of strings.

SEE ALSO
PRINT, XYPRINT, PPRI NT, PS CREEN, XY

EXAMPLES
HPRINT "CELLULAR PHONE TEST"
// Prints the string text,
// between the quotation
// marks, on the color
// display, starting at the
// current xy location.

ICON (HOST Only)

USE
To create a user-defined graphic on the color display.

SYNTAX
ICON pixels,rows,name

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>pixels</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>rows</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
<tr>
<td>name</td>
<td>Name of data array containing the bit pattern used for the icon.</td>
</tr>
</tbody>
</table>

ACTION
Creates a user-defined graphic image. The icon is created by defining an area of pixels (consisting of rows of pixels and pixels per row) and a data array specifying with corresponding bits which pixels to turn on and which to turn off (1's to turn on, 0's to turn off). Pixels appear in the current foreground color. The top left corner of the defined area is placed at the current XY setting. The data array must consist of 32 bit words.

SEE ALSO
COLOR, Graphics (2-13-5), XY, DATA

EXAMPLES
See Graphic examples (2-13-5).

IDATA

USE
Same as DATA except uses integer values only.

SYNTAX
IDATA name = {argument, argument, ..., argument}

SEE ALSO
DATA
IF ENDIF

USE
To provide a decision point inside a macro.

SYNTAX
IF condition sequence; ENDIF
or
IF condition
sequence
ENDIF

ELEMENT DESCRIPTION
condition Any expression which is TRUE or FALSE when evaluated.
sequence Any valid TMAC command or, if using the 2nd syntax example above, a series of commands that is performed if condition is evaluated as TRUE.

Zero (0) is evaluated by TMAC as FALSE. Any non-zero number is evaluated by TMAC as TRUE.

ACTION
Performs sequence if condition is evaluated as TRUE. If the condition is FALSE, sequence is not executed and execution flow passes to the command following ENDIF.

Every IF command must be followed by an ENDIF. IF commands can be nested within other IF commands if the ENDIF to the inner IF command is also inside the outer IF command and so on.

SEE ALSO IF ELSE ENDIF, IF ELIF ELSE ENDIF, CASE

EXAMPLES
IF Mod>20 Flag=1;ENDIF // Sets Flag = 1 if Mod is greater // than 20. Does nothing if Mod is // not greater than 20.

IF Dev > 3 // The following IF ENDIF is executed // if Dev is greater than 3.

IF Freq_Err < 1000 // The command Freq=200000 is command Freq=2000000 // executed if Freq_Err is less // than 1000 and Dev is greater // than 3.

ENDIF
ENDIF
**IF ELIF ELSE ENDF**

**USE**
To provide multiple decision points inside a macro.

**SYNTAX**

```
IF condition
    sequence
ELIF condition
    sequence
ELIF condition
    sequence
...
ELSE
    sequence
ENDIF
```

**ELEMENT** | **DESCRIPTION**
---|---
condition | Any expression which is TRUE or FALSE when evaluated.
sequence | Any valid TMAC command or a series of commands that is performed if condition is evaluated as TRUE.

Zero (0) is evaluated by TMAC as FALSE. Any non-zero number is evaluated by TMAC as TRUE.

**ACTION**
Same as **IF ENDF** command, except the **IF ELIF** command allows multiple conditions to be evaluated.

Each ELIF (Else If) part allows the checking of a condition. If the condition is TRUE, the sequence immediately following is executed. The optional ELSE sequence is executed if no ELIF condition is TRUE. An ENDIF command must be included at the end of an IF ELIF group of commands. If two ELIF commands are TRUE, only the sequence following the first TRUE ELIF encountered is executed.

**SEE ALSO**
**IF ENDF, IF ELSE ENDF, CASE**

**EXAMPLES**

```
IF X=3
    PRINT "X=3" // If X=3, this command is executed and
                  // the rest are skipped.
ELIF X=4
    PRINT "X=4" // If X=4, this command is executed and
                  // the rest are skipped.
ELIF X=5
    PRINT "X=5" // If X=5, this command is executed and
                  // the rest are skipped.
ENDIF
```
IF ELSE ENDIF

To provide multiple decision points inside a macro.

**SYNTAX**

```plaintext
IF conditionA command1
ELSE command2
ENDIF

...or

IF conditionB
  sequence1
ELSE
  sequence2
ENDIF
```

**ELEMENT**  | **DESCRIPTION**
---|---
conditionA | Any expression which is TRUE or FALSE when evaluated.
command1 | Any valid TMAC command that is performed if conditionA is evaluated as TRUE.
command2 | Any valid TMAC command that is performed if conditionA is evaluated as FALSE.
conditionB | Any expression which is TRUE or FALSE when evaluated.
sequence1 | Any valid TMAC command or a series of commands that is performed if conditionB is evaluated as TRUE.
sequence2 | Any valid TMAC command or a series of commands that is performed if conditionB is evaluated as FALSE.

Zero (0) is evaluated by TMAC as FALSE. Any non-zero number is evaluated by TMAC as TRUE.

**ACTION**

Same as IF ENDIF command, except the IF ELSE command allows two conditions to be evaluated.

If the condition is TRUE, command1 or sequence1 is executed. If the condition is FALSE, command2 or sequence2 is executed. Every IF ELSE command must have an ENDIF.

**SEE ALSO**

IF ENDIF, IF ELIF ELSE ENDIF, CASE

**EXAMPLES**

```plaintext
IF D>4 PRINT "D HIGH"  // Prints D HIGH if D is greater than 4; otherwise ...
ELSE PRINT "D LOW"     // D LOW is printed.
ENDIF

IF T=3
  T=T+1                // 1 is added to T if T is equal to 3;
ELSE
  T=T-1                // otherwise ...
ENDIF
```
IF ELSE (Shorthand)

USE
To utilize an abbreviated version of the IF ELSE command.

SYNTAX
command (condition? true: false)

ELEMENT DESCRIPTION

command Incomplete command which is performed with one of the arguments (true or false).

condition Any expression which is TRUE or FALSE when evaluated.

true An argument which when combined with command creates a valid TMAC command.

false An argument which when combined with command creates a valid TMAC command.

ACTION
Evaluates condition to determine how to finish the incomplete command. If condition is TRUE, then the argument true is used in combination with command to create a valid TMAC command. Conversely, if condition is FALSE, then the argument false is used.

No ENDIF is used with this command.

SEE ALSO IF ENDIF, IF ELSE ENDIF, IF ELIF ELSE ENDIF, CASE

EXAMPLES
PRINT (D>4? "D HIGH";"D LOW") // If D is greater than 4, // then the complete command // is: PRINT "D HIGH". If D // is not greater than 4, then // the complete command is: // "PRINT "D LOW". The now // completed command is then // executed.

T= (T=3? T+1:T-1) // If the condition T=3 is true, then the // command T=T+1 is executed. If the // condition is false, then the command // T=T-1 is executed.
INPUT

To enter numerical or string data during macro execution.

SYNTAX

INPUT variable or
INPUT string

ELEMENT DESCRIPTION

variable Any valid variable.
string Any valid string.

ACTION

Stops execution of the macro until numerical or string data is entered and ENTER Key is pressed. The value of the data entered is assigned to variable or string and command execution continues. For the HOST, an input box is displayed with a width set using the EDIT:WIDTH command. Data is entered using the DATA ENTRY Keys and pressing the ENTER Key.

For the Sp Tst, data is entered via the RS-232 terminal.

SEE ALSO

Variables and Arrays in Macros (2-11), Colors (2-13-1), EDIT:WIDTH, VAR, STRING

EXAMPLES

*dmc 'GET_X', begin
screen:user // Displays blank user screen.
*wai // Allows time for user screen to be
       // set up prior to executing next
       // command.
        
cls // Initializes user screen.
bcolor 0 // Sets screen background color.
color white,black // Set on-screen text character color
       // and background contrast color.
edit:width 76 // Sets input box to 76 pixels wide.
       // (19 x 4 = 76. See EDIT:WIDTH.).

print "Enter 65.3"
keypad:claim // Claims keypad for input commands.
input x // Opens input box on screen and
       // pauses macro execution.

print x // Prints 65.3 on screen.
keypad:unclaim // Returns keypad control to HOST.
end
INTERP

USE
To execute commands that may be defined after the macro containing the commands is defined.

SYNTAX
INTERP "string;string;... ;string"

ELEMENT DESCRIPTION

string String variables, literal strings and string functions.

ACTION
The INTERP command executes each string as a command and is limited to one line.

The INTERP command may be run inside or outside a macro.

SEE ALSO
EXEC

EXAMPLES
*dmc "run_tests",begin
interp "test"+str($1)  // Runs macro named "test + place-
// holder value" entered with
// run_tests command.
end

/* Entering run_tests 4 executes this macro and passes 4 into the macro. If test4 macro is not defined before defining the run_tests macro, the run_tests macro will fail while being defined (typically during upload) unless the INTERP command is used. */

KEY

USE
To wait and return keycode of next key pressed.

SYNTAX
KEY

ACTION
Pauses macro and ...

(HOST only) returns keycode (See Appendix B) of the next key pressed on keypad of the HOST.

(Sp Tst only) returns the ANSI terminal keycode (usually ASCII) of the next key pressed on the RS-232 terminal keyboard.

Used with a PRINT, PPRINT, HPRINT or XYPRINT command.

SEE ALSO
INPUT, KEY?
EXAMPLES

PRINT KEY  // For HOST: After a Front Panel Key
            // is pressed, the keycode is printed on
            // the color display.
            // For Sp Tst Option: After a terminal key
            // is pressed, the keycode is printed on
            // the terminal monitor.

*dmc"keycodes",begin
var keyCode=0  // Initializes keycode value.
keypad:claim  // Claims keyboard (HOST only).
do
    keyCode=key  // Queries for keycode of key pressed
    print keyCode  // and assigns keycode to keyCode.
until keyCode=8200 // Prints keycode of last key pressed.
// Returns macro execution to beginning
// of loop unless asterisk (*) key
// is pressed.
    print "done"  // Prints done when DO UNTIL loop is
    keypad:unclaim // exited.
end  // Releases control of HOST keyboard.

KEY?

USE
To return 1 if key is pressed; 0 otherwise.

SYNTAX
KEY?

ACTION
Returns 1 if key is pressed on keypad of HOST (HOST only) or on keyboard of
RS-232 terminal (Sp Tst only) when command is encountered. Returns 0 if no
key is pressed.

Used with IF..., WHILE, DO UNTIL commands.

SEE ALSO
INPUT, KEY

EXAMPLES
*dmc"key_query",begin
keypad:claim  // Claims HOST keypad. Use only
            // for HOST.
while !key?  // Loops while no key is pressed.
    print "waiting"  // Prints "waiting" for each pass of
    // loop until a key is pressed.
wend
    print "key pressed!"  // Prints message when key is pressed
    // and WHILE loop is exited.
keypad:unclaim  // Releases HOST keypad (HOST
end  // only).
KEYPAD:CLAIM (HOST Only)

USE To direct all input from Front Panel Keypad to the TMAC Interpreter.

SYNTAX KEYPAD:CLAIM

ACTION Directs all input from the Front Panel Keyboard to the TMAC Interpreter.

Recommended for use with INPUT, KEY and KEY? commands.

SEE ALSO KEYPAD:UNCLAIM

EXAMPLES See KEY example.

KEYPAD:ERASE (HOST Only)

USE To erase Soft Function Key definitions.

SYNTAX KEYPAD:ERASE n

ELEMENT DESCRIPTION

n Any constant, variable, number or any expression that evaluates to a number. Range of n is 1 to 6.

ACTION Erases definition for Soft Function Key Fn.

Before KEYPAD:ERASE command is executed

![Before KEYPAD:ERASE command is executed](image1)

After KEYPAD:ERASE command is executed

![After KEYPAD:ERASE command is executed](image2)

Figure 3-2 KEYPAD:ERASE Example

SEE ALSO Soft Function Key Displays (2-13-4), KEYPAD:LABEL, KEYPAD:SOFT

KEYPAD:LABel

USE
To create Soft Function Key definitions.

SYNTAX
KEYPAD:LABel n, label

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. Range of n is 1 to 6.</td>
</tr>
<tr>
<td>label</td>
<td>Any string variable or literal string surrounded by quotation marks (e.g., &quot;TEST1&quot;).</td>
</tr>
</tbody>
</table>

ACTION
Creates a new label for specified Soft Function Key Fn.

Issuing this command with label as an empty set of quotation marks ("") is the same as executing the KEYPAD:ERASE command.

![Before KEYPAD:LABel command is executed](image1)
![After KEYPAD:LABel command is executed](image2)

Figure 3-3 KEYPAD:LABel Example

SEE ALSO
Soft Function Key Displays (2-13-4), KEYPAD:ERASE, KEYPAD:SOFT

EXAMPLES
KEYPAD:LAB 4, "Menu"
KEYPAD:SOFT (HOST Only)

USE To create a Soft Function Key frame.
SYNTAX KEYPAD:SOFT
ACTION Creates the Soft Function Key frame at the bottom of the current screen.

Before KEYPAD:SOFT command is executed

After KEYPAD:SOFT command is executed

Figure 3-4 KEYPAD:SOFT Example

SEE ALSO Soft Function Key Displays (2-13-4), KEYPAD:ERASE, KEYPAD:LABEL

EXAMPLES KEYPAD:SOFT
*dmc"panic",begin
screen:user;"wai;cls
keypad:claim
keypad:soft

keypad:label 6,"panic"
k=key
if k=32896
print "AIIIIIIEEE!!!"
endif
keypad:unclaim
end

// Sets up user screen.
// Claims keypad for TMAC interpreter.
// Creates blank Soft Function Key frame.
// Labels Soft Function Key F6.
// Waits for key press and assigns keycode to k.
// Checks to see if key pressed was F6.
// Provides response to user input if F6 was pressed.
// Releases keypad.
KEYPAD:UNCLAIM (HOST Only)

USE  To release the Front Panel Keypad for normal use.
SYNTAX KEYPAD:UNCLAIM
ACTION Releases the Front Panel Keypad for normal use. Used after claiming the keypad (See KEYPAD:CLAIM) for routing all keypad input to the TMAC Interpreter.
SEE ALSO KEYPAD:CLAIM
EXAMPLES See KEY example.

KILL

USE  To delete multi-tasking macro.
SYNTAX KILL name
ELEMENT DESCRIPTION
name Any string variable or literal string surrounded by quotation marks (e.g., "TEST1").
ACTION Removes specified task specified by name from the schedule queue and frees memory.
SEE ALSO TASK, ACTIVATE, TPAUSE, TSTOP, SLEEP, WAKE

LEN

USE  To return the number of characters in a string.
SYNTAX LEN(string)
ELEMENT DESCRIPTION
string String variable or literal string surrounded by quotation marks (e.g., "radio test").
ACTION Returns the length of string in number of characters.
SEE ALSO ASC, STR, STRPOS, VAL
EXAMPLES

string how_long
how_long = "any string"
B = len(how_long) // Assigns 10 to B.
L = len("another string") // Assigns 14 to L.
LJPRINT (Sp Tst Only)

**USE**
To display numbers or text, left justified, on color display of HOST.

**SYNTAX**
LJPRINT value,x,y,width

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Any constant, variable, string variable, literal string (surrounded by quotation marks), number or any expression that evaluates to a string or number. Strings or string variables and numbers or number variables cannot be mixed.</td>
</tr>
<tr>
<td>x,y</td>
<td>Starting point of new display area in pixels (0,0 to 639,349).</td>
</tr>
<tr>
<td>width</td>
<td>Any constant, variable or integer or any expression that evaluates to an integer. Expressed in number of pixels</td>
</tr>
</tbody>
</table>

**ACTION**
Erases the display area on the screen of the HOST defined by the x,y starting position and width in pixels, then prints value, left justified, in the defined area.

**SEE ALSO**
CENTER, HPRINT, PRINT, RJPRINT, XYPRINT

**EXAMPLES**

```plaintext
$s = "How many lambs did ";
ljprint $+"Mary have?",100,100,410 // Prints, left justified
    // in erased area display
    // area, "How many lambs
    // did Mary have?"

ljprint 3*2-5,100,130,20 // Prints 1.
```

---

LN

**USE**
To return the natural logarithm of n.

**SYNTAX**
LN(n)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, positive number or any expression that evaluates to a positive number.</td>
</tr>
</tbody>
</table>

If n ≤ 0, LN command execution results in an error message as this function is not defined for that range of values.

**ACTION**
Returns the natural logarithm (base e) of n.

**SEE ALSO**
EXP, LOG

**EXAMPLES**

```plaintext
VAR Nat_log
D = 5
Nat_log = LN(D) // The natural logarithm of 5 (1.609438)
    // is assigned Nat_log.
```
LOG

USE
To return the logarithm of n (log n).

SYNTAX
LOG(n)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, positive number or any expression that evaluates to a positive number.</td>
</tr>
</tbody>
</table>

If n ≤ 0, LOG command execution results in an error message as this function is not defined for that range of values.

ACTION
Returns the base 10 logarithm of n.

SEE ALSO
**, EXP, LN

EXAMPLES
VAR Log_C
C = 5
Log_C = LOG(C)     // The base 10 logarithm of 5 (0.698970004)
                 // is assigned to Log_C.

NVRCL (HOST Only)

USE
To recall a variable or array stored in non-volatile memory.

SYNTAX
NVRCL name,location

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of variable or array to be saved.</td>
</tr>
<tr>
<td>location</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. Range 0 to 283.</td>
</tr>
</tbody>
</table>

ACTION
Recalls global variables and arrays saved in nonvolatile memory. The recalled value(s) are assigned to any previously declared variable or array. Location of the variable or first element of the array in nonvolatile memory is identified by location. Arrays take a memory location for each element.

SEE ALSO
Saving Variables and Arrays (2-3-2), NVSAV

EXAMPLE
See NVRCL examples (2-3-2).
**NVSAV (HOST Only)**

**USE**
To save variables and arrays in non-volatile memory.

**SYNTAX**

```
NVSAV name, location
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of variable or array to be saved.</td>
</tr>
<tr>
<td>location</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. Range 0 to 283.</td>
</tr>
</tbody>
</table>

**ACTION**
Saves global variables or arrays in non-volatile memory. Memory location to store variable or first element of array is identified by `location`.

Global variables and arrays are not saved when Test Set is turned off and need to be saved in non-volatile memory.

**SEE ALSO**
Saving Variables and Arrays (2-3-2), NVRCL

**EXAMPLE**
See NVSAV examples (2-3-2).

---

**OFF**

See FALSE.

---

**ON**

See TRUE.

---

**OR**

**USE**
Logical OR.

**SYNTAX**

```
cmdA argument1 OR argument2 cmdB  or
cmdA argument1 OR argument2 OR argument3 ... OR conditionX cmdB
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdA</td>
<td>Any decision point command (IF, ELIF, UNTIL, WHILE).</td>
</tr>
<tr>
<td>argument1</td>
<td>Any expression that evaluates to TRUE, FALSE, 0 (zero) or any non-zero number (0 = FALSE, any non-zero number = TRUE).</td>
</tr>
<tr>
<td>argument2 &amp; on</td>
<td>Same as <code>argument1</code>.</td>
</tr>
<tr>
<td>cmdB</td>
<td>Any valid TMAC command.</td>
</tr>
</tbody>
</table>
ACTION

Evaluates argument1 and argument2 and, if either are TRUE (or evaluate to a non-zero number), then the result of the OR operation is TRUE (FALSE, otherwise). If more than one OR operation is encountered then the OR operations are evaluated from left to right. Successive ORs evaluate the result of previous OR operations along with individual expressions.

Table 3-6 lists the expression combinations and the OR operation results.

<table>
<thead>
<tr>
<th>argument1</th>
<th>argument2</th>
<th>OR Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>

Table 3-6 OR Operation Results

SEE ALSO

!, AND, =, !=, <, >, <=, =>, DO UNTIL, IF ..., WHILE

EXAMPLES

```
var x=7, z=4
if x=7 OR x<z      // Decision point command utilizing OR
                  // as condition with accompanying
                  // expressions to evaluate. Condition is
                  // TRUE, because one of the expressions
                  // evaluates to TRUE.
     print y       // Performs resulting command since
                  // condition is TRUE.
ENDIF
```

PAINT

USE

To perform a paint fill.

SYNTAX

PAINT x,y,fc,bc

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>x,y</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349.</td>
</tr>
<tr>
<td>fc</td>
<td>Any constant, variable, number or any expression that evaluates to an number. Range is 0 to 15. See Table 2-2 for the available colors and associated numerical values.</td>
</tr>
<tr>
<td>bc</td>
<td>Any constant, variable, number or any expression that evaluates to an number. Range is 0 to 15. See Table 2-2 for the available colors and associated numerical values.</td>
</tr>
</tbody>
</table>

ACTION

Performs a paint fill of simple or complex shapes using Fill Color (fc) specified, starting from Starting Location (x,y) and attempts to fill entire screen stopping only at the Boundary Color (bc) specified.

SEE ALSO

Graphics and Text (2-13-5).
PIXEL

USE  To create a user-defined dot on the color display.
SYNTAX  PIXEL
ACTION  Creates a dot in the current foreground color at the current x,y location.

Changing foreground color or xy location after placing creating a dot on the color display, leaves the dot unaffected.

SEE ALSO  COLOR, Graphics and Text (2-13-5). XY
EXAMPLES  XY 300,150
COLOR 6
PIXEL  // Creates a brown dot at location 300,150.

PIXLEN (HOST Only)

USE  To return length of a string in pixels.
SYNTAX  PIXEL\[string\]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>Any string variable or literal string surrounded by quotation marks (e.g., &quot;TELEPHONE TESTS&quot;)</td>
</tr>
</tbody>
</table>

ACTION  Returns the length of string in pixels.

Useful for centering text on the color display of the HOST.

EXAMPLES  STRING Mode  // Declares string Mode.
Mode = "Test Mode 4"  // Initializes Mode.
P = PIXEL\[Mode\]  // Assigns pixel length of Mode (160),
                  // which is the length of "Test Mode 4"
                  // in pixels to p.
XY \[(640-P)/2\]-1,40  // Sets point for horizontal center.
Print Mode  // Prints Mode string.
PIXLEN? (Sp Tst Only)

**USE**  
To return number of pixels required to display a given value.

**SYNTAX**  
PIXLEN? value

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Any constant, variable, number, any expression that evaluates to a number, string variable or literal string surrounded by quotation marks (e.g., &quot;REPEATER TEST&quot;). Any numerical or string expression is evaluated by TMAC prior to the pixel length being calculated.</td>
</tr>
</tbody>
</table>

**ACTION**  
Returns the length in pixels required to display value on the screen of the HOST.

**SEE ALSO**  
CENTER, LPRINT, RPRINT
PPRINT (HOST Only)

**USE**  
To print responses out the RS-232 Connector.

**SYNTAX**  
PPRINT argument or  
PPRINT argument, argument, ... ,argument or  
PPRINT argument, argument, ... ,argument, "\ccc" or  
PPRINT %0nl, argument, argument, ... ,argument or  
PPRINT argument, %0nl, argument, ... ,argument, "\ccc"

<table>
<thead>
<tr>
<th>ELEMENTS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument</td>
<td>Expression to be printed. Can be a mathematical expression, a literal string or the contents of variables. Mathematical expressions are calculated and string functions are performed before the results are printed.</td>
</tr>
<tr>
<td>0</td>
<td>Zero signifies leading zeros are added to fill the field width. The 0 is omitted for no leading zeros. Applies only to numeric expressions.</td>
</tr>
<tr>
<td>n</td>
<td>Specifies the field width. n is the number of digit spaces (decimal point counts as one). Applies only to numeric expressions.</td>
</tr>
<tr>
<td>l</td>
<td>Specifies the numeric base the data is printed in. Applies only to numeric expressions. Omit for a default of Signed Decimal or enter one of the following:</td>
</tr>
<tr>
<td></td>
<td>B Binary</td>
</tr>
<tr>
<td></td>
<td>Q Octal</td>
</tr>
<tr>
<td></td>
<td>H Hexadecimal</td>
</tr>
<tr>
<td></td>
<td>U Unsigned decimal</td>
</tr>
<tr>
<td></td>
<td>D Signed decimal</td>
</tr>
<tr>
<td>ccc</td>
<td>Optional non-printable characters in octal form used to specify the output.</td>
</tr>
<tr>
<td></td>
<td>-or-</td>
</tr>
<tr>
<td></td>
<td>n or l</td>
</tr>
<tr>
<td></td>
<td>r</td>
</tr>
</tbody>
</table>

**ACTION**  
Sends all responses that would normally be printed on the color display to the device connected to the RS-232 Connector. (x,y location has no affect on the PPRINT command.)

%0nl contains the format settings for numeric expressions and can be placed anywhere in the PPRINT command. There can be several format settings in each PPRINT command, each one changing the leading zero format, the numeric base and the field width for the expressions that follow. Format settings do not affect the printing of strings.

**SEE ALSO**  
PRINT, PSCREEN
**PRINT**

**USE**
To print on the display screen (HOST) or to print responses out the OPT. RS-232 Connector (Sp Tst Option).

**SYNTAX**

```
PRINT argument or
PRINT argument, argument, .... ,argument, or
PRINT %0nl, argument, .... ,argument, or
PRINT argument, .... ,%0nl, argument,
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>argument</td>
<td>Expression to be printed. Can be a mathematical expression, a literal string or the contents of variables. Mathematical expressions are calculated and string functions are performed before the results are printed.</td>
</tr>
<tr>
<td>0</td>
<td>Zero signifies leading zeros are added to fill the field width. The 0 is omitted for no leading zeros. Applies only to numeric expressions.</td>
</tr>
<tr>
<td>n</td>
<td>Specifies field width. ( n ) is the minimum number of digit spaces (decimal point counts as one). Applies only to numeric expressions. For Sp Tst, if ( l ) is D, ( n ) can be in the form of ( b.a ), where ( b ) designates minimum number of digits before the decimal point and ( a ) designates the maximum number of digits after the decimal point.</td>
</tr>
<tr>
<td>l</td>
<td>Specifies the numeric base the data is printed in. Applies only to numeric expressions. Omit for a default of Signed Decimal or enter one of the following:</td>
</tr>
<tr>
<td></td>
<td>B  Binary</td>
</tr>
<tr>
<td></td>
<td>Q  Octal</td>
</tr>
<tr>
<td></td>
<td>H  Hexadecimal</td>
</tr>
<tr>
<td></td>
<td>U  Unsigned decimal</td>
</tr>
<tr>
<td></td>
<td>D  Signed decimal</td>
</tr>
</tbody>
</table>

**ACTION**
Prints numerical and string values out the OPT. RS-232 Connector (Sp Tst) or at the current \( x,y \) location on the color display (HOST). Several values are printed on the same line by separating them with commas. If the optional ending comma is used, the next PRINT command continues on the same line; otherwise a new line is started. If an argument consists of a variable, the value of the variable is printed. Mathematical expressions are calculated before printing.
%0n/ contains the format settings for numeric expressions and can be placed anywhere in the PRINT command. There can be several format settings in each PRINT command, each one changing the leading zero format, the numeric base and the field width for the expressions that follow. Format settings do not affect the printing of strings.

SEE ALSO

HPRINT, XYPRINT, PPRINT, PSCREEN, XY

EXAMPLES

X=44
Y=37.83
$="String"
PRINT %04,X," ",,$ // Specifies leading zeros, a field width // of 4 characters and signed decimal (by // default) for any following numerical // outputs. The following is printed: // 0044 String.

PRINT X,%3," ",,Y // Prints X as is. Then specifies no // leading zeros, a field width of 3 and // signed decimal (by default) for any // following numerical outputs. A single // space is printed. When Y is printed, // the print field is expanded to accommodate // Y. Because each argument of the PRINT // command (except for Y) is following by a // comma, output is displayed on one line. // The following is printed: 44 37.83

PRINT %4B,X // Specifies no leading zeros, a field // width of 4 and Binary numeric base for // any following numerical outputs. The // following is printed: 101100. // The field width is enlarged to // accommodate X.

PRINT %7,$,Y // Specifies no leading zero, a field // width of 7 and signed decimal (by // default) for any following numerical // outputs. The following is printed: // String 37.83. 2 blank character // spaces before 3 are provided by the // field width of 7.

PRINT %03Q,X,$,6,Y // Specifies leading zeros, a field width // of 3 and Octal numeric base. Then X // is printed as formatted. Then String // is printed immediately after because // of the comma between. Then no leading // zeros, a field width of 6 and signed // decimal numeric base (by default) is // specified. Then Y is printed. // The complete printed expression is as // follows: 054String 37.83.
PSCREEN (HOST Only)

USE To print the display screen to output device.

SYNTAX PSSCREEN

ACTION Prints the display screen to device connected to the RS-232, GPIB or PRINTER (parallel) Connectors. Output device, connector and print parameters are specified by the Printer submenu under menu item "External I/O" in the Auxiliary Functions Menu (see 3-3-10 of IFR-1900 Operation Manual). Command is entered through RS-232 or GPIB Connector (see 6-2-1 or 6-2-4 of IFR-1900 Operation Manual) or placed in a macro.

Issuing this command is equivalent to pressing the PRINT SCRN Key on the Front Panel of the Test Set. See Figure 3-1 of IFR-1900 Operation Manual.

SEE ALSO PRINT, PPRINT

RAND

USE To specify seed for RND random generator.

SYNTAX RAND x

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

ACTION Sets the seed (starting point) for the random generator.

Using the same seed causes the random generator to produce the same series of pseudo-random numbers.

SEE ALSO RND

EXAMPLES

*DMC"RND_GEN",BEGIN
RAND 12 // Sets a starting point for the RND command.
FOR X=1 TO 10 STEP 1
Z=RND(100) // Generates the following numbers:
PRINT Z // 47, 77, 31, 27, 39, 100, 57, 46, 41 and 30.
NEXT X
END

/* Each time RAND is set to 12, the RND(100) command produces the same series of random numbers. */
RETURN

USE
To return a value from a function or to return macro execution to the
previous macro.

SYNTAX
RETURN

ACTION
Stops command execution of the current macro when command is
encountered. If command is encountered and the current macro was called
from another macro, command execution returns to the macro from which the
current macro was called. Command execution resumes at the command
immediately following the call of the macro or function. If a variable is
specified with the RETURN command, the value of the variable is returned.

```
"dmc"macro_2",begin
.
.
return
end

"dmc"macro_1",begin
.
.
.
macro_2  '/" more
 commands "/
end

```

Figure 3-5 RETURN Command

EXAMPLE
```
*DMC "Even",BEGIN
   // Macro named "Even" that is called
   // by the macro named "Test" below.
IF $1 % 2 = 0
   PRINT "$1," IS EVEN"
ELSE
   RETURN
ENDIF
END

*DMC "Test",BEGIN
   // Macro is called with two arguments:
   // x raised to the power of y (Test x,y).
Y = $1**$2
PRINT "$1,"TO THE POWER OF ",$2," IS ",Y
Even Y
PRINT "TEST IS DONE"
END

/* If passed variable is odd ($1 % 2 equals 1), the RETURN
command returns command execution back to the macro Test at
the PRINT "TEST IS DONE" command. */

(Examples continued on next page.)
```
(Examples continued from previous page.)

*DMC "Maximum",BEGIN
VAR Max=0
EDIT:WIDTH 45
FOR X=1 TO 10
INPUT Y
IF Y>Max
  Max=Y
ENDIF
NEXT X
RETURN Max
END

/* The RETURN command returns the value of Max. The function
  Maximum cannot be executed by simply entering it's name.
  Functions are executed by assigning them to a variable or
  using them in a PRINT command. */

Z=Maximum           // Function Maximum is executed and the
                    // result is assigned to Z.

PPRINT Maximum     // Function Maximum is executed and the
                    // result is printed on the Host
                    // Terminal (HOST only).

---

RJPRINT (Sp Tst Only)

USE
To display numbers or text, right justified, on color display of HOST.

SYNTAX
RJPRINT value,x,y,width

ELEMENT DESCRIPTION

value Any constant, variable, string variable, literal string
(surrounded by quotation marks), number or any expression
that evaluates to a string or number.

- Strings or string variables and numbers or number variables
cannot be mixed.

x,y Both either a constant, variable, number or any expression
that evaluates to a number. Ranges are 0 to 639 and
0 to 349.

width Any constant, variable or integer or any expression that
evaluates to an integer. Expressed in number of pixels

ACTION Erases the display area on the screen of the HOST defined by the x,y starting
position and width in pixels, then prints value, right justified, in the defined
area.

SEE ALSO CENTER, HPRINT, LJPRINT, PRINT, XYPRINT

3-72
**RND**

**USE**
To generate a pseudo-random integer between 0 and n.

**SYNTAX**
RND(n)

**ELEMENT** | **DESCRIPTION**
---|---
$n$ | Any constant, variable, number or any expression that evaluates to a number.

**ACTION**
Generates a pseudo-random integer between 0 and n. If n is not an integer, the value is truncated.

**SEE ALSO**
RAND

**EXAMPLES**

| WW=RND(100) | // Assigns a pseudo-random number between 0 and 100 to WW.

---

**ROOM**

**USE**
To return the amount of memory size available.

**SYNTAX**
ROOM

**ACTION**
Returns the available memory size, in bytes, when printed.

**SEE ALSO**
Macros (2-9)

**EXAMPLES**

| PRINT ROOM | // Prints the amount of memory available to the color display (HOST) or out // the OPT. RS-232 (Sp Tst Option).

---

**ROTATE (HOST Only)**

**USE**
To rotate text, character by character.

**SYNTAX**
ROTATE $n$

**ELEMENT** | **DESCRIPTION**
---|---
$n$ | Degrees the character is rotated.

**ACTION**
Specifies the orientation (in 90° intervals) new text is printed, character by character, on the color display. Table 3-7 shows the amount of clockwise rotation for values of $n$.

<table>
<thead>
<tr>
<th>$n$</th>
<th>ROTATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 89</td>
<td>0</td>
</tr>
<tr>
<td>90 to 179</td>
<td>90°</td>
</tr>
<tr>
<td>180 to 269</td>
<td>180°</td>
</tr>
<tr>
<td>270 to 359</td>
<td>270°</td>
</tr>
</tbody>
</table>

Table 3-7 Character Rotation for $n$ Values
SCREEN:USER (HOST Only)

USE
To create a blank screen in active window.

SYNTAX
SCREEN:USER

ACTION
Creates a blank user screen (with a dark blue background) in the active window.

- Stops any active task.
- Blank screen may be used to create user-developed screens.
- User screen decreases run time because of no readings to update.

SEE ALSO
USER (for Sp Tst only)

EXAMPLES

```
screen: user  // Displays blank user screen.
*wait  // Forces macro execution to wait until
       // user screen is completed.
cls  // Initializes user screen.
bcolor 0  // Establishes background color (user
       // should establish background color
       // of new screen).
```

SIGN

USE
To return -1, 0, 1 if $n$ is a negative number, zero or a positive number.

SYNTAX
SIGN($n$)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

ACTION
Returns -1 if $n < 0$, 0 if $n = 0$ or 1 if $n > 0$.

SEE ALSO
ABS, FLOOR

EXAMPLES

```
S=SIGN(-6)  // Assigns -1 to S.
```
## SIN

**USE**
To return the sine of angle $n$ measured in radians.

**SYNTAX**
SIN($n$)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

**ACTION**
Returns the sine of angle $n$ measured in radians.

**SEE ALSO**
COS

**EXAMPLES**
VAR Sine_of.radians
Sine_of.radians=SIN(3.1415/3)  // Assigns the sine of
\[ \frac{\pi}{3} \] (0.86601) to
// Sine_of.radians.

## SLEEP

**USE**
To temporarily stop multitasking a macro.

**SYNTAX**
SLEEP $name$

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$name$</td>
<td>Name of the macro to remove from schedule queue. String variable or literal string surrounded by quotation marks (e.g., &quot;TEST_4&quot;).</td>
</tr>
</tbody>
</table>

**ACTION**
Removes the specified task from the schedule queue. The task can be put back into the schedule queue using the WAKE command.

**SEE ALSO**
Multitasking Macros (2-12). WAKE

**EXAMPLES**
See Multitasking examples (2-12).

## SOUND

**USE**
To create audio tones.

**SYNTAX**
SOUND $freq,duration$

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$freq$</td>
<td>Any constant, variable, positive number or any expression that evaluates to a positive number. Effective range is 125 to 15000.</td>
</tr>
<tr>
<td>$duration$</td>
<td>Any constant, variable, positive number or any expression that evaluates to a positive number.</td>
</tr>
</tbody>
</table>

**ACTION**
Generates a tone of frequency $freq$ in Hz for a length of time $duration$ in ms and routes the tone to the Test Set Speaker.

**EXAMPLES**
SOUND 600,1000  // Creates a 600 Hz tone for 1 sec.
**SQR**

**USE**

To return to the positive square root of \( n \).

**SYNTAX**

\[
SQR(n)
\]

**ELEMENT** | **DESCRIPTION**
---|---
\( n \) | Any constant, variable, positive number or any expression that evaluates to a positive number.

**ACTION**

Returns the positive square root of \( n \).

The square root function returns 0 for the square root of negative numbers.

**SEE ALSO**

EXP, SIGN

**EXAMPLES**

```plaintext
VAR Square_Root
Square_Root = SQR(30)  // Assigns the positive square root of 30 (5.477226) to Square_Root.
```

**STATus:OPERation:INSTRument (HOST Only)**

**USE**

To control or query the Operation Instrument Register.

**SYNTAX**

\[
\text{STATus:OPERation:INSTRument:CONDition?} \quad \text{or} \quad \text{STATus:OPERation:INSTRument:ENABLE} \quad n \quad \text{or} \quad \text{STATus:OPERation:INSTRument:ENABLE?} \quad \text{or} \quad \text{STATus:OPERation:INSTRument:EVENT?}
\]

**ELEMENT** | **DESCRIPTION**
---|---
\( n \) | Any constant, variable, number or any expression that evaluates to a number (16 bit value). Range is 0 to 65535.

**ACTION**

Specifies or returns the contents of the following registers within the Operation Instrument Register: Condition register, Enable register and Event register.

The Operation Instrument Register is used with the \texttt{INITiate} and \texttt{FETCH} commands. Once an \texttt{INITiate} command is completed and the meter is ready to be read, the appropriate meter ready bit is set to 1. This indicates the meter is ready for the \texttt{FETCH} command.

**SEE ALSO**

Status Subsystem (2-16). Figures 2-5 and 2-6
STATus:OPERation (HOST Only)

**USE**
To control or query the Operation Status Register.

**SYNTAX**
- `STATus:OPERation:CONDition?` or
- `STATus:OPERation:ENABLE n` or
- `STATus:OPERation:ENABLE?` or
- `STATus:OPERation:EVENt?`

**ELEMENT** | **DESCRIPTION**
---|---
`n` | Any constant, variable, number or any expression that evaluates to a number (16 bit value). Range is 0 to 65535.

**ACTION**
Specifies or returns the contents of the following registers within the Operation Status Register: Condition register, Enable register and Event register.

The Operation Status Register contains the Waiting For Arm, Waiting For Trigger and Instrument Summary bits. The Operation Instrument Register sets the Instrument Summary bit.

**SEE ALSO**
Status Subsystem (2-16), Figures 2-5 and 2-6,
`STATus:OPERation:INSTRumen` commands

---

STATus:PRESet (HOST Only)

**USE**
To set all the bits in the enable registers to an initialized condition.

**SYNTAX**
`STATus:PRESet`

**ACTION**
Sets all bits in enable registers to an initialized condition. Presets the enable registers of the following status registers to the associated condition:

<table>
<thead>
<tr>
<th>STATUS REGISTER</th>
<th>CONDITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Status</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Operation Instrument</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Questionable Status</td>
<td>OFF (#h0000)</td>
</tr>
<tr>
<td>Instrument Status</td>
<td>ON (#h7FFF)</td>
</tr>
<tr>
<td>Instrument Summary</td>
<td>ON (#h7FFF)</td>
</tr>
</tbody>
</table>

Table 3-8 Enable Register Preset Conditions

**SEE ALSO**
Status Subsystem (2-16), Figure 2-6
STATus:QUEStionable:INSTRument:ISUMmary (HOST Only)

USE
To control or query the Instrument Summary Status Register.

SYNTAX
STATus:QUEStionable:INSTRument:ISUMmary:CONDITION? or
STATus:QUEStionable:INSTRument:ISUMmary:ENABLE n or
STATus:QUEStionable:INSTRument:ISUMmary:ENABLE? or
STATus:QUEStionable:INSTRument:ISUMmary:EVENt?

ELEMENT DESCRIPTION

n Any constant, variable, number or any expression that evaluates to a number (16 bit value). Range is 0 to 65535.

ACTION
Specifies or returns the contents of the following registers within the Instrument Summary Status Register: Condition register, Enable register and Event register.

The Instrument Summary Status Register acts as an extension of the Instrument Status Register. The Instrument Summary Status Register reports the exceeding of the upper and lower limit of the SINAD and Phase Meter and Digital Multimeter.

SEE ALSO
Status Subsystem (2-16), Figure 2-6

EXAMPLES

*DMM "DMM_READ",BEGIN
STAT:QUES:INSTR:ISUM:ENABLE 12 // Sets enable register to read DMM Upper and Lower
  // Limit exceeded.
STAT:QUES:INSTR:ENABLE 2 // Sets enable register to pass the Instrument Summary Status Register result.
STAT:QUES:ENABLE 8192 // Sets enable register to pass the Instrument Status Register result.
*CLS // Clears all condition and event registers.
SCREEN:DMM // Displays DMM Operation Screen.
FPRI NT M_DMM? // Prints DMM reading to Host.
IF (*STB? & 8) != 0 // If Status Byte reports Questionable Status bit as 1,
  PPRINT "limit exceeded" // print limit exceeded to Host.
ELSE
  PPRINT "limit not exceeded" // print limit not exceeded to Host.
ENDIF
END

(Examples continued on next page.)
(Examples continued from previous page.)

/* The above macro performs a DMM reading and checks the Status Byte for the reporting of the DMM Upper or Lower Limit being exceeded. The enable registers of the Instrument Summary, Instrument and Questionable Status Registers are set to pass the information to the Status Byte. */

STATus:QUEStionable:INSTrument (HOST Only)

<table>
<thead>
<tr>
<th>USE</th>
<th>To control or query the Instrument Status Register.</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYNTAX</td>
<td>STATus:QUEStionable:INSTrument:CONDition? or</td>
</tr>
<tr>
<td></td>
<td>STATus:QUEStionable:INSTrument:ENABLE n or</td>
</tr>
<tr>
<td></td>
<td>STATus:QUEStionable:INSTrument:ENABLE? or</td>
</tr>
<tr>
<td></td>
<td>STATus:QUEStionable:INSTrument:EVENt?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number (16 bit value). Range is 0 to 65535.</td>
</tr>
</tbody>
</table>

| ACTION | Specifies or returns the contents of the following registers within the Instrument Status Register: Condition register, Enable register and Event register. |

| SEE ALSO | Status Subsystem (2-16), Figure 2-6 |
| EXAMPLES | See STATus:QUEStionable:INSTrument:ISUMmary. |
STATus:QUESTionable (HOST Only)

USE
To control or query the Questionable Status Register.

SYNTAX
STATus:QUESTionable:CONDITION? or
STATus:QUESTionable:ENABLE n or
STATus:QUESTionable:ENABLE? or
STATus:QUESTionable:EVENt?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number (16 bit value). Range is 0 to 65535.</td>
</tr>
</tbody>
</table>

ACTION
 Specifies or returns the contents of the following registers within the Questionable Status Register: Condition register, Enable register and Event register.

The Questionable Status Register reports events that could bring Test Set results into question. The Instrument Status Register provides the input. Figure 2-6 shows events reported by both registers.

SEE ALSO Status Subsystem (2-16), Figure 2-6

EXAMPLES See STATus:QUESTionable:INSTRument:ISUMmary.

STOP

USE
For Multitasking. To stop command execution of all macros.

SYNTAX
STOP

ACTION
Stops command execution of all macros regardless of where the STOP command is located.

EXAMPLES
*DMC "Test", BEGIN
Y= $1**$2
IF $2 < 0
    STOP // If the passed variable is negative,
    ENDIF // the STOP command is executed and all
    PRINT "$1,"TO THE POWER OF ",$2," IS ",Y
    PRINT "TEST IS DONE"
ENDIF

SEE ALSO Multitasking Macros (2-12)
STR

USE
To return the string equivalent of a number.

SYNTAX
STR(n)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

ACTION
Returns the string equivalent (character representation) of number n.

SEE ALSO
ASC, CHR, VAL

EXAMPLES
$s = STR(236) // Assigns the characters 236 to // string variable $.

STRING

USE
To declare string variables and arrays.

SYNTAX
STRING name or
STRING name, name, ....,name or
STRING name[index] or
STRING name[index], name[index], ....,name[index] or
STRING name[index] or
STRING name[index][length] (Sp Tst only)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the string declared. The first character of name must be a letter while the remaining characters can be letters, digits and the underscore character. The length of the name can be from 2 to 31 characters.</td>
</tr>
<tr>
<td>index</td>
<td>Number of array elements minus one.</td>
</tr>
<tr>
<td>length</td>
<td>Maximum length of each array element specified by index (Sp Tst only).</td>
</tr>
</tbody>
</table>

ACTION
Declares string variables and arrays. String variables and arrays cannot be initialized when declared. (Sp Tst only: Specifying maximum length of each element in a string array may be used to save system memory. Not to be confused with substringing; see 2-8.)

String variables and arrays are either local or global. Local string variables are declared inside a macro and have no meaning outside that macro. Global string variables are declared outside macros and are used inside or outside any macro.

SEE ALSO
String Variables and Functions (2-8)

EXAMPLES
STRING Mess_1,Mess_2 // Declares string variables named // Mess_1 and Mess_2.
Mess_1="Test " // Assigns "Test " to variable Mess_1.
Mess_2=Mess_1+"complete" // Assigns "Test complete" to the // string variable Mess_2.
**STRPOS**

**USE**
To return position of a string within another.

**SYNTAX**

```
STRPOS(string1,string2)
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>string1</td>
<td>Any string variable or literal string surrounded by quotation marks (e.g., &quot;RADIO TESTS NO. 2&quot;).</td>
</tr>
<tr>
<td>string2</td>
<td>Any string variable or literal string surrounded by quotation marks (e.g., &quot;TESTS NO.&quot;).</td>
</tr>
</tbody>
</table>

**ACTION**
Returns the position of `string2` inside `string1`. If `string2` is not found, -1 is returned.

The first position of `string1` is 0.

**SEE ALSO**
LEN, VAL

**EXAMPLES**

```
STRING One, Two
One = "preset"
Two = "set"
X = STRPOS(One,Two) // Assigns 1 to X. "set" starts with the 4th character in "preset."
```

**SYSTem:COMMunicate:GPIB:ADDRes (HOST Only)**

**USE**
To specify Test Set address and select Talk/Listen for GPIB mode.

**SYNTAX**

```
SYSTem:COMMunicate:GPIB:ADDRes a
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Any constant, variable, integer or any expression that evaluates to an integer. Range is 0 to 31.</td>
</tr>
</tbody>
</table>

**ACTION**
Specifies Test Set address and selects Talk/Listen for GPIB mode.

**SEE ALSO**
GPIB Operation (2-15)

**EXAMPLES**
See SYSTem:COMMunicate:GPIB:SPOLL?
SYStem:COMMunicate:GPIB:CMD (HOST Only)

USE To issue commands through GPIB Connector.
SYNTAX SYStem:COMMunicate:GPIB:CMD string
ELEMENT DESCRIPTION
string Any string variable or literal string surround by quotation marks.
ACTION When operating the Test Set as a GPIB Controller, issues a command sequence through the GPIB Connector.
SEE ALSO GPIB Operation (2-15)

SYStem:COMMunicate:GPIB:CONTroller (HOST Only)

USE To make Test Set a GPIB controller.
SYNTAX SYStem:COMMunicate:GPIB:CONTroller
ACTION Selects Controller for the GPIB Operation Mode making the Test Set a GPIB controller.

For the Test Set to become a GPIB controller, the address of the Test Set must be set to an address that does not conflict with a slave device address and the GPIB Operation Mode must be set to Controller.

SEE ALSO GPIB Operation (2-15), SYStem:COMMunicate:GPIB:ADDResS
EXAMPLES See SYStem:COMMunicate:GPIB:SPOLL?

SYStem:COMMunicate:GPIB:DCL (HOST Only)

USE To issue a Device Clear message.
SYNTAX SYStem:COMMunicate:GPIB:DCL
ACTION When operating the Test Set as a GPIB Controller, issues a Device Clear message.
SEE ALSO GPIB Operation (2-15)

SYStem:COMMunicate:GPIB:GET (HOST Only)

USE To issue a Group Execute Trigger.
SYNTAX SYStem:COMMunicate:GPIB:GET
ACTION When operating the Test Set as a GPIB Controller, issues a Group Execute Trigger.
SEE ALSO GPIB Operation (2-15)
**SYSTem:COMMunicate:GPIB:LONly (HOST Only)**

**USE** To select Listen Only as the GPIB Operation Mode.

**SYNTAX** SYSTem:COMMunicate:GPIB:LONly

**ACTION** Selects Listen Only as the GPIB Operation Mode.

To select the Talk/Listen GPIB Mode of Operation, again, reset the Test Set address using the SYST:COMM:GPIB:ADDR command.

When the Test Set becomes a GPIB Listen Only device, the address is first set using the SYST:COMM:GPIB:ADDR command. After setting the address, this command selects Listen Only as the GPIB Mode of Operation.

**SEE ALSO** GPIB Operation (2-15), SYSTem:COMMunicate:GPIB:ADDRESS

**SYSTem:COMMunicate:GPIB:PRINTER (HOST Only)**

**USE** To direct the output of the printer to the GPIB Connector.

**SYNTAX** SYSTem:COMMunicate:GPIB:PRINTER

**ACTION** Selects the GPIB Connector for printer output.

**SEE ALSO** GPIB Operation (2-15)

**SYSTem:COMMunicate:GPIB:SLAVE (HOST Only)**

**USE** To specify the destination of SYSTem:PTHrough:GPIB commands.

**SYNTAX** SYSTem:COMMunicate:GPIB:SLAVE address

**ELEMENT** DESCRIPTION

| address | Any constant, variable, integer or any expression that evaluates to a integer. Range is 0 to 31. |

**ACTION** When operating the Test Set as a GPIB Controller, specifies the destination of SYSTem:PTHrough:GPIB commands to the peripheral device with address.

**SEE ALSO** GPIB Operation (2-15), SYSTem:PTHrough:GPIB

**EXAMPLES** See SYSTem:COMMunicate:GPIB:SPOLL?
SYSTem:COMMunicate:GPIB:SPOLL? (HOST Only)

USE
To perform a Serial Poll on a device.

SYNTAX
SYSTem:COMMunicate:GPIB:SPOLL? address

ELEMENT DESCRIPTION
address Any constant, variable, integer or any expression that evaluates to a integer. Range is 0 to 31.

ACTION
When operating the Test Set as a GPIB Controller, performs a Serial Poll on the device with address and returns the result. The 8 bit result is device dependent, except bit 6, which is the Service Request bit.

SEE ALSO
GPIB Operation (2-15)

EXAMPLES
*DMC "Control",BEGIN
SYST:COMM:GPIB:ADDR 30
SYST:COMM:GPIB:CONT
END

/* The macro Control establishes the Test Set as a GPIB Controller. */

*DMC "Slave",SYST:COMM:GPIB:SLAV $1
/* The macro Slave sets the slave address for a peripheral device to $1. */

*DMC "M1200",SYST:PTHR:GPIB 5,STR($1)+'\n'
/* The macro M1200 passes $1, as a string, through the GPIB Connector to the peripheral device with an address of 5. The \n provides a line feed and marks the end of the command passed. */

*DMC "Read_1200",BEGIN
Control
M1200 "DVMAC"
WHILE SYST:PTHR:SER:QUE? =0 // While no Input from RS-232 Host.
IF SYST:COMM:GPIB:SRQ? =1 // If there is a Service Request,
P=SYST:COMM:GPIB:SPOLL? 5 // Poll to see if from 1200.
IF P&64 != 0 // If bit 6 is 1, SRQ is from 1200.
M1200 "DVMRM" // If SRQ is from 1200, take a reading.
ENDIF
END

(Examples continued on next page.)
SYSTem:COMMunicate:GPIB:SRQ? (HOST Only)

(* Examples continued from previous page. *)

The macro Read_1200 assumes a FM/AM-1200S/A (1200) is connected to the GPIB Connector of the Test Set and has an address of 5. The following sequence occurs:

- Makes the Test Set a controller.
- Passes the DVMAC command to the 1200. Sets the 1200 for autoranging voltmeter readings.
- While there is no Input from the RS-232 Host of the Test Set, continually checks for Service Requests from the 1200.
- Upon each 1200 Service Request, the Test Set passes the DVMRM command to the 1200. This command returns a voltmeter reading to the Host of the Test Set. */

SYSTem:COMMunicate:GPIB:SRQ? (HOST Only)

**USE**
To check peripheral devices for a Service Request.

**SYNTAX**
SYSTem:COMMunicate:GPIB:SRQ?

**ACTION**
When operating the Test Set as a GPIB Controller, checks peripheral devices for a Service Request. Returns a 1 if there was a Service Request since last check. Once read, returns 0 until next Service Request.

**SEE ALSO**
GPIB Operation (2-15)

**EXAMPLES**
See SYSTem:COMMunicate:GPIB:SPOLL?

SYSTem:COMMunicate:GPIB:TONly (HOST Only)

**USE**
To select Talk Only as the GPIB Operation Mode.

**SYNTAX**
SYSTem:COMMunicate:GPIB:TONly

**ACTION**
Selects Talk Only as the GPIB Operation Mode.

To select the Talk/Listen GPIB Mode of Operation, again, reset the Test Set address using the SYST:COMM:GPIB:ADDR command.

When the Test Set becomes a GPIB Talk Only device, addressing is unnecessary. The Test Set must be the only talking device on the GPIB bus and the remaining devices must be compatible listeners.

**SEE ALSO**
GPIB Operation (2-15), SYSTem:COMMunicate:GPIB:ADDRes
SYSTEM:COMMunicate:SERial

**USE**
To control the RS-232 Connector or select for printer output.

**SYNTAX**
- `SYSTEM:COMMunicate:SERial:BAUD baud_rate` or
- `SYSTEM:COMMunicate:SERial:PARity parity` or
- `SYSTEM:COMMunicate:SERial:BITS data_bits` or
- `SYSTEM:COMMunicate:SERial:SBITS stop_bits` or
- `SYSTEM:COMMunicate:SERial:PACE handshake` or
- `SYSTEM:COMMunicate:SERial:ECHO echo` or
- `SYSTEM:COMMunicate:SERial:PRINTER`

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>baud_rate</code></td>
<td>Any constant, variable, number or any expression that evaluates to a number. Valid values: 300, 600, 1200, 2400, 4800, 9600, 19200, 36400, 57600¹ and 115200¹.</td>
</tr>
<tr>
<td><code>parity</code></td>
<td><strong>NONE, EVEN, ODD, MARK</strong> or <strong>SPACE</strong>.</td>
</tr>
<tr>
<td><code>data_bits</code></td>
<td>Any constant, variable, number or any expression that evaluates to a number. Valid values: 7 or 8.</td>
</tr>
<tr>
<td><code>stop_bits</code></td>
<td>Any constant, variable, number or any expression that evaluates to a number. Valid values: 1 or 2.</td>
</tr>
<tr>
<td><code>handshake</code></td>
<td><strong>XON</strong> (Xon/Xoff), <strong>HW</strong> (Hardware) or <strong>NONE</strong>.</td>
</tr>
<tr>
<td><code>echo</code></td>
<td>Any constant, variable, number or any expression that evaluates to a number. Valid values: 1 (Enable) or 0 (disable).</td>
</tr>
</tbody>
</table>

1. Sp Tst only.

**ACTION**
Specifies serial communication parameters for operation through the RS-232 Connector or selects the RS-232 Connector for printer output.

**EXAMPLES**

```plaintext
*DMC "Control","BEGIN
STRING Result
SYSTEM:COMM:SER:BAUD 19200
SYSTEM:COMM:SER:PAR NONE
SYSTEM:COMM:SER:BITS 8
SYSTEM:COMM:SER:SBITS 1
SYSTEM:COMM:SER:PACE XON
SYSTEM:COMM:SER:ECHO 1
SYST:PTHR:SER "query?"
WHILE !(SYST:PTHR:SER:QUE?)" // Loops while RS-232 queue is empty.
  TPAUSE
WEND
Result = SYST:PTHR:SER?
// Stores received string in Result.
END
```

3-87
**SYSTem:CURsor:DEFaults (HOST Only)**

**USE**  
To restore cursor and Soft Function Keys to default state.

**SYNTAX**  
SYSTem:CURsor:DEFaults

**ACTION**  
Restores the cursor positioning and Soft Function Key level of active screen to default state.  
This command should be followed by a screen command to ensure that screens appear correctly.

**SEE ALSO**  
SYSTem:DEFaults, SYSTem:DISPlay:DEFaults

---

**SYSTem:DATE**

**USE**  
To set or return system date.

**SYNTAX**  
SYSTem:DATE year,month,day or  
SYSTem:DATE?

**ELEMENT**  
**DESCRIPTION**

<table>
<thead>
<tr>
<th>element</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>year</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 99.</td>
</tr>
<tr>
<td>month</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 1 to 12.</td>
</tr>
<tr>
<td>day</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 1 to 31.</td>
</tr>
</tbody>
</table>

**ACTION**  
Specifies or returns system date in the form: year, month, day.  
Be careful to observe correct day of month when setting.

**SEE ALSO**  
SYSTem:TIME

---

**SYSTem:DEFaults (HOST Only)**

**USE**  
To restore Test Set to default state.

**SYNTAX**  
SYSTem:DEFaults

**ACTION**  
Restores the Test Set to factory default state.

**SEE ALSO**  
SYSTem:CURsor:DEFaults, SYSTem:DISPlay:DEFaults
SYSTEM:DISPLAY:DEFAULTS (HOST Only)

USE To restore Test Set to default Color Set.
SYNTAX SYSTEM:DISPLAY:DEFAULTS
ACTION Restores the Test Set to the default Manufacturer Color Set.
SEE ALSO SYSTEM:CURSOR:DEFAULTS SYSTEM:DEFAULTS

SYSTEM:ERROR?

USE To return number and description of system errors.
SYNTAX SYSTEM:ERROR?
ACTION Returns earliest error not yet read. The error number and description is returned.

Once read, the data for the error read is removed from memory. The earliest 16 errors that have not been read are held in memory with the rest being ignored. The 16 stored errors can be cleared from memory using the *CLS command.

SEE ALSO *CLS

SYSTEM:FREQLIST (HOST Only)

USE To specify or return Frequency List settings.
SYNTAX SYSTEM:FREQLIST:GENERATOR list_num,freq or SYSTEM:FREQLIST:GENERATOR? list_num or SYSTEM:FREQLIST:OFFSET list_num,freq or SYSTEM:FREQLIST:OFFSET? list_num or SYSTEM:FREQLIST:RECEIVER list_num,freq or SYSTEM:FREQLIST:RECEIVER? list_num or SYSTEM:FREQLIST:SCAN list_num,b or SYSTEM:FREQLIST:SCAN? list_num

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>list_num</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 99.</td>
</tr>
<tr>
<td>freq</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 250.0 to 2010000.0.</td>
</tr>
<tr>
<td>b</td>
<td>Enable (1) or disable (0).</td>
</tr>
</tbody>
</table>

ACTION Specifies or returns Generator, Receiver, Offset and Scan enable settings. Specifies one of the 100 frequency listings (list_num). Frequencies (freq) are specified in kHz. The scan status (b) of each frequency listing can be turned on or off to be included or removed from frequency list scanning.
SYSTem:FREQuency:LOCK (HOST Only)

USE
To slave the RF Generator, Receiver and Spectrum Analyzer frequency together or to return frequency lock status.

SYNTAX
SYSTem:FREQuency:LOCK <b> or
SYSTem:FREQuency:LOCK?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;b&gt;</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Valid values: 1 (Enable) or 0 (disable).</td>
</tr>
</tbody>
</table>

ACTION
Enables/disables or returns the frequency lock state of the Receiver, Generator and Spectrum Analyzer RF lock feature.

- When user enters a frequency in any one of the above three operations modes, the remaining two operation modes automatically change to the identical frequency.
- Works only in Direct Mode.

SYSTem:KEY (HOST Only)

USE
To simulate or return a key pressed or simulated pressed of the Front Panel Keypad.

SYNTAX
SYSTem:KEY <n> or
SYSTem:KEY?

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;b&gt;</td>
<td>Any constant, variable, number or any expression that evaluates to a number. For valid keycodes see Appendix B.</td>
</tr>
</tbody>
</table>

ACTION
Simulates pressing a Front Panel Key with keycode <n> or returns the keycode of the last key pressed or simulated pressed. If a key has not been pressed, -1 is returned.

SEE ALSO
KEY?, SYSTem:KEY:DEFinite

EXAMPLES
SYST:KEY 2112 // Simulates pressing RCVR MODE Key.

3-90
**SYSTeM:KEY:DEFine (HOST Only)**

**USE**
To assign command sequence to a Front Panel Key.

**SYNTAX**
SYSTeM:KEY:DEFine n,sequence

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number. For valid keycodes see Appendix B.</td>
</tr>
<tr>
<td>sequence</td>
<td>String variable or literal string surround by quotation marks (e.g., &quot;Test_1; print Max; a=Set_value&quot;). Limited to 80 string characters.</td>
</tr>
</tbody>
</table>

**ACTION**
Assigns the specified command sequence string to key with keycode n. If the key is pressed, the command sequence is executed.

A maximum of 16 keys can be assigned at any one time. Appendix A lists Front Panel Keys with predefined constants.

**SEE ALSO**
SYSTeM:KEY:DELeTe

**EXAMPLES**
SYST:KEY:DEF 8200, "Test_1" // Assigns macro Test_1 to the * (asterisk) DATA ENTRY Key.
SYST:KEY:DEL 8200 // Deletes the *-Test_1 assignment.

**SYSTeM:KEY:DELeTe (HOST Only)**

**USE**
To cancel a Front Panel keycode command sequence definition.

**SYNTAX**
SYSTeM:KEY:DELeTe n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number. For valid keycodes see Appendix B.</td>
</tr>
</tbody>
</table>

**ACTION**
Cancels the command sequence string assigned to keycode n with command:
SYSTeM:KEY:DEFine.

**SEE ALSO**
SYSTeM:KEY:DEFine
**SYSTem:MILLIsec?**

**USE**
To return current value of the millisecond counter.

**SYNTAX**
SYSTem:MILLIsec?

**ACTION**
Returns the current value of the millisecond counter. Range of returned values is 0 to 4294967295 ($2^{32} - 1$).

Counter returns to zero approximately every 49.7 days.

**SEE ALSO**
SYSTem:TIME, TICKS?

**SYSTem:PLOT (HOST Only)**

**USE**
To select plotter output.

**SYNTAX**
SYSTem:PLOT:GPIB or SYSTem:PLOT:SERial

**ACTION**
Selects GPIB or HOST RS-232 Connector for plotter output.

**SEE ALSO**
PSSCREEN, SYSTem:COMMunicate:GPIB:PRINTer, SYSTem:COMMunicate:SERial

**SYSTem:PTHThrough:GPIB (HOST Only)**

**USE**
To send out or receive a string through the GPIB Connector.

**SYNTAX**
SYSTem:PTHThrough:GPIB string or SYSTem:PTHThrough:GPIB address,string or SYSTem:PTHThrough:GPIB? or SYSTem:PTHThrough:GPIB? address

**ELEMENT**

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
</tr>
<tr>
<td>address</td>
</tr>
</tbody>
</table>

**ACTION**
When operating the Test Set as a GPIB Controller, sends out or waits for and accepts (receives) a string through the GPIB Connector. Optional peripheral address defaults to the address set by the last SYSTem:COMMunicate:GPIB:SLaVe command.

Strings contain commands unique to the controlled device.

**SEE ALSO**
GPIB Operation (2-15), SYSTem:COMMunicate:GPIB:SLaVe

**EXAMPLES**
See SYSTem:COMMunicate:GPIB:SPOLL?
SYSTem:PTHRough:SERial (HOST Only)

**USE**
To send or receive a string through the RS-232 Connector.

**SYNTAX**
SYSTem:PTHRough:SERial "string" or
SYSTem:PTHRough:SERial? or
SYSTem:PTHRough:SERial:KEY?

**ELEMENT** | **DESCRIPTION**
--- | ---
string | Any string variable or literal string surrounded by quotation marks (e.g., "VOLTS").

**ACTION**
Sends or waits and receives a string (or, in the case of the (SYSTem:PTHRough:SERial:KEY? query, one character) through the HOST RS-232 Connector.

Strings contain commands unique to the controlled device.

**SEE ALSO**

SYSTem:PTHRough:SERial:QUEue? (HOST Only)

**USE**
To determine is RS-232 queue is empty or contains data.

**SYNTAX**
SYSTem:PTHRough:SERial:QUEue?

**ACTION**
Returns a non-zero value if data is in the RS-232 queue, 0 if the RS-232 queue is empty.

SYSTem:TIME

**USE**
To set or return time in 24 hour time format.

**SYNTAX**
SYSTem:TIME hours, minutes, seconds or
SYSTem:TIME?

**ELEMENT** | **DESCRIPTION**
--- | ---
hours | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 23.
minutes | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 59.
seconds | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 59.

**ACTION**
Sets 24 hour time in hours, minutes and seconds or returns 24 hour time in hours, minutes, seconds and milliseconds.

Although 0.1 milliseconds are displayed when returned, accuracy is only assured to within 16.5 milliseconds.

**SEE ALSO**
SYSTem:MILLIsec?, TiCKS?

**EXAMPLES**
SYST:TIME?
/* Sample time returned: 13,17,32.760 (1:17:32.760 PM) */
**TAB**

**USE**
To return string spaces (or tabs) to the right.

**SYNTAX**
TAB(n)

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number.</td>
</tr>
</tbody>
</table>

**ACTION**
Returns a blank string of \( n \) character spaces in length.

**EXAMPLES**

CLS
PRINT TAB(?),"FREQ = " // Starts with blank screen.
PRINT \( \text{FREQ} = \) starting seven // spaces from the left edge of color // display.

---

**TASK**

**USE**
For Multitasking.

**SYNTAX**
TASK name

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String variable or literal string surrounded by quotation marks. Name of the macro to be declared a task.</td>
</tr>
</tbody>
</table>

**ACTION**
Declares the specified macro as a task. A macro must be declared a task before it can be placed on the schedule queue for multitasking. The macro must already be loaded in memory.

**SEE ALSO**
Multitasking Macros (2-12), **ACTIVATE**

**EXAMPLES**
See Multitasking example (2-12).

---

**TICKS?**

**USE**
To return elapsed time in milliseconds.

**SYNTAX**
TICKS?

**ACTION**
Returns the current value of the millisecond counter. Range of returned values is 0 to \( 4294967295 \) (\( 2^{32} - 1 \)).

Used for calculating elapsed time in milliseconds

**SEE ALSO**
SYSTem:MILLIsec?
TPAUSE

USE For multitasking and sharing execution time with other Test Set activities.
SYNTAX TPAUSE
ACTION While multitasking, stops command execution and passes command execution to the next task in the schedule queue, if one exists.
SEE ALSO Multitasking Macros (2-12), TSTOP
EXAMPLES See Multitasking example (2-12).

TRUE

USE To produce a 1.
SYNTAX TRUE or ON
ACTION Provides a concise method of specifying a logical 1.
SEE ALSO !
EXAMPLES X=TRUE // Assign 1 to X.

TSTOP

USE For Multitasking.
SYNTAX TSTOP
ACTION While multitasking, stops command execution and removes the currently running task from the schedule queue and performs a TPAUSE. Command execution passes to the next task in the schedule queue.
SEE ALSO Multitasking Macros (2-12), TPAUSE
EXAMPLES See Multitasking example (2-12).
USER (Sp Tst Only)

USE
To create a blank screen.

SYNTAX
USER

ACTION
Creates a blank user screen in the active window.

Blank screen may be used to create user-developed screens.

SEE ALSO
SCREEN:USER

EXAMPLES
user
  // Displays blank user screen.
*wai
  // Forces macro execution to wait until
  // user screen is completed.
cls
  // Initializes user screen.
bcolor 0
  // Establishes background color (user
  // should establish background color
  // of new screen).

VAL

USE
To return the numeric value of a string.

SYNTAX
VAL(string)

ELEMENT   DESCRIPTION
string     String variable or literal string surrounded by quotation marks.

ACTION
Returns the numeric value of a character representation of a number in string.

SEE ALSO
ASC, CHR, STR,

EXAMPLES
STRING Commd
Commd = "567"
  // Assigns the string "567" to Commd.
C = VAL(Commd)
  // Assigns the number 567 to C, a numeric variable.
VAR

USE
To declare variables and arrays.

SYNTAX
VAR name or
VAR name, name, .... ,name or
VAR name[index] or
VAR name[index], name[index], .... ,name[index] or
VAR name, name[index]

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of variable or array declared. The first character of the name is a letter while the remaining characters can be letters, digits or the underscore. The length of the name can be from 2 to 31 characters.</td>
</tr>
<tr>
<td>index</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Number of array elements minus 1 (Index = # of array elements - 1). Index of an array. Identifies the elements of the array by specifying the order of occurrence. The count of elements starts with 0.</td>
</tr>
</tbody>
</table>

ACTION
Variables are declared with a name. Arrays are declared with a name and an index. Variables and arrays can be initialized when declared using an equals (=) character. Using a variable name previously used to declare a variable voids the previous variable.

Variables and arrays are either local or global. Local variables and arrays are declared inside a macro and have no meaning outside that macro. Global variables and arrays are declared outside macros and are used inside or outside any macro.

SEE ALSO
Numeric Variables and Arrays (2-3), NVSAV

EXAMPLES
VAR Flag=0, Set_Squ=0.9  // Variables Flag and Set_Squ are declared and initialized to 0 and 0.9.

VAR Dat[2]={3.2,3.4,3.6} // Array Data is declared with 3 elements and each element is initialized to a value.
**USE**
To transfer the video image from one video page to another.

**SYNTAX**
```
VIDEOpage:COPY from, to
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>from</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 1.</td>
</tr>
<tr>
<td>to</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 1.</td>
</tr>
</tbody>
</table>

**ACTION**
Copies video from the active screen to the static screen or from the static screen to the active screen.

- Two video pages (screens) are available: video page 0 and video page 1.
- This command is used in combination with VIDEOpage:SET to build screens in the background, then display new screen, instantly, and for animation applications.

**SEE ALSO**
VIDEOpage:SET

**EXAMPLES**
```plaintext
video:set 0,1 // Makes default screen 0 to continue to be displayed and screen 1 to be active (or be written to) but remain in background.

/* Build screen 1 by sending print commands, etc., as would normally. However, screen results do not appear. */

video:copy 1,0 // Copies the video from screen 1 in the background to screen 0 presently being displayed. Changes made to screen 1 is displayed immediately.
```
VIDEOpage:SET (HOST Only)

**USE**
To specify Video Page settings.

**SYNTAX**
VIDEOpage:SET disp, write

**ELEMENT** | **DESCRIPTION**
--- | ---
disp | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 1.
write | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 1.

**ACTION**
Specifies which Video Page to display and which Video Page is active (may be written to); the other screen remains static. Two video pages are available: video page 0 and video page 1. See Table 3-9.

<table>
<thead>
<tr>
<th>disp</th>
<th>write</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>(Default at power up) Screen 0 is displayed; Screen 0 is active. Screen 1 remains static.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Screen 0 is displayed; Screen 1 is active. Screen 0 remains static.</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Screen 1 is displayed; Screen 0 is active. Screen 1 remains static.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Screen 1 is displayed; Screen 1 is active. Screen 0 remains static.</td>
</tr>
</tbody>
</table>

Table 3-9 Video Page Settings

Two screen displays are available (screen 0 and screen 1): one active and one static for use as pages to switch back and forth. Two video pages allow user to construct or make changes to a screen display before actually displaying the new screen. Video Page Control allows user to specify which video page to currently display or to copy from one to the other.

**SEE ALSO**
VIDEOpage:COPY

---

**WAKE**

**USE**
For Multitasking.

**SYNTAX**
WAKE name

**ELEMENT** | **DESCRIPTION**
--- | ---
name | String variable or literal string surrounded by quotation marks. Name of the task put back on the schedule queue.

**ACTION**
Reenters a task (taken off the schedule queue by a SLEEP command) into the schedule queue.

**SEE ALSO**
Multitasking macros (2-12), SLEEP

**EXAMPLES**
See Multitasking examples (2-12).
WCLOSE

USE  
To close a window.

SYNTAX  
WCLOSE n

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Number of the window to close. Windows are numbered in the order they are created. Overlapping windows must be closed in the reverse order they are opened.</td>
</tr>
</tbody>
</table>

ACTION  
Closes (deletes) a generated window specified by n.

SEE ALSO  
Windows (2-13-3), WOPEN, WINDOW?

EXAMPLES  
See Windows examples (2-13-3).

WHILE WEND

USE  
To perform a set of commands repetitiously while a conditional expression is true.

SYNTAX  
WHILE condition
  sequence
WEND

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>condition</td>
<td>Any expression which is TRUE or FALSE when evaluated.</td>
</tr>
<tr>
<td>sequence</td>
<td>Command sequence performed as long as condition is true.</td>
</tr>
</tbody>
</table>

ACTION  
The WHILE loop repeatedly executes command sequence until condition is evaluated FALSE. If condition is FALSE when first evaluated, the command sequence is not executed once like the DO loop, because the condition is tested before the command sequence.

EXAMPLES  
N=0  // Assigns 0 to N.
WHILE N!=2  // Evaluates Condition.
  N=N+1  // Performs command while condition is TRUE.
WEND  // End of WHILE loop.

/* While performs the following steps:
   o N is compared to 2 (N=0).
   o 1 is added to N.
   o N is compared to 2 (N now equals 1).
   o 1 is added to N.
   o N is compared to 2 (N now equals 2).
   o N skips over commands inside loop and commands are not executed. Macro execution continues after WEND command.

(Examples continued on next page.)
(Examples continued from previous page.)

VAR FF
WHILE FF<=LN(2*S)
   FF=FF<<2
   PRINT FF // The value of FF is shifted left 2 bits
   // and printed until it is greater than
   // LN(2*S).
WEND

WHILE SYSTEM:KEY? #= 32896 // Continues looping until the
   // Front Panel Key with a keycode
   // of 32896 is pressed.
WEND

WINDOW?

USE To return the number of the window currently selected.
SYNTAX WINDOW?
ACTION Returns the number of the currently selected window. The selected window is
    the window affected by the WMOVE and WCLOSE commands. The last
    opened window is the selected window unless a WSEL command has selected
    another window.
SEE ALSO Windows (2-13-3), WSEL, WMOVE, WCLOSE
EXAMPLES See Windows examples (2-13-3).

WMOVE

USE To move windows.
SYNTAX WMOVE x,y

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>x,y</td>
<td>Both either a constant, variable, number or any expression</td>
</tr>
<tr>
<td></td>
<td>that evaluates to a number. Ranges are 0 to 639 and</td>
</tr>
<tr>
<td></td>
<td>0 to 349.</td>
</tr>
<tr>
<td></td>
<td>New coordinates on the color display of the top left corner of</td>
</tr>
<tr>
<td></td>
<td>the window to be moved.</td>
</tr>
</tbody>
</table>

ACTION Moves the currently selected window so that the top left corner of the window
is located at the specified point.

Window size and shape are unaffected.

Moving a window that overlaps another window disrupts the window
overlapped.
**WOPEN**

**USE**
To create windows.

**SYNTAX**

```
WOPEN c, x1, y1, x2, y2
```

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 15. Number of the color selected for the window. See Table 2-2.</td>
</tr>
<tr>
<td>x1, y1</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349. Color display coordinates of the top left corner of the window being opened.</td>
</tr>
<tr>
<td>x2, y2</td>
<td>Both either a constant, variable, number or any expression that evaluates to a number. Ranges are 0 to 639 and 0 to 349. Color display coordinates of the bottom right corner of the window being opened.</td>
</tr>
</tbody>
</table>

**ACTION**
Creates a window on the color display of color c with a top left point of x1, y1 and a bottom right corner of x2, y2. Height and width of window are forced to 16 pixel increments.

A lack of available memory can cause a window not to open.

**SEE ALSO**
Windows (2-13-3), WCLOSE

**EXAMPLES**
See Windows examples (2-13-3).
**WSEL**

**USE**
To change which window is currently selected.

**SYNTAX**
WSEL n, hide

**ELEMENT** | **DESCRIPTION**
---|---

n | Any constant, variable, number or any expression that evaluates to a number.

hide | Any constant, variable, number or any expression that evaluates to a number. Valid values: 1 (Hide window) or 0 (show window).

**ACTION**
Makes window n the currently selected window and hides window if hide is set to 1.

- The selected window is the window affected by the WMOVE and WCLOSE command.
- Windows are numbered in the order they are opened.
- A hidden window does not appear on the color display and must be closed and opened again to reappear.

**SEE ALSO**
Windows (2-13-3), WINDOW?, WMOVE, WCLOSE

**EXAMPLES**
See Windows examples (2-13-3).

---

**XY**

**USE**
To specify a point on the color display for other display commands to use.

**SYNTAX**
XY horiz, vert

**ELEMENT** | **DESCRIPTION**
---|---

horiz | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 639.

vert | Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 349.

**ACTION**
Specifies a point (one pixel) on the color display to be used by the PRINT, PIXEL, HPRINT and ICON commands. Where point 0,0 is the top left corner of the color display, positive vert (y coordinate) is the number of pixels in the downward direction and positive horiz (x coordinate) is the number of pixels to the right.

- The PRINT command moves the xy position to a new line using the current HEIGHT setting for the height of the new line.

**SEE ALSO**
Color Display (2-13-2), Graphics (2-13-5)

**EXAMPLES**
See Graphics example (2-13-5).
XYPRINT

USE
To print on the display screen.

SYNTAX
XYPRINT x,y, argument or
XYPRINT x,y, argument, argument, ...... argument, or
XYPRINT x,y, %0nl, argument, ...... argument, or
XYPRINT x,y, argument, ...... %0nl, argument,

ELEMENT DESCRIPTION

x Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 639.
y Any constant, variable, number or any expression that evaluates to a number. Range is 0 to 349.
argument Expression to be printed. Can be a mathematical expression, a literal string or the contents of variables. Mathematical expressions are calculated and string functions are performed before the results are printed.
0 Zero signifies leading zeros are added to fill the field width. The 0 is omitted for no leading zeros. Applies only to numeric expressions.
n Specifies field width. n is the minimum number of digit spaces (decimal point counts as one). Applies only to numeric expressions. If l is D, n can be in the form of b.a, where b designates minimum number of digits before the decimal point and a designates the maximum number of digits after the decimal point.
l Specifies the numeric base the data is printed in. Applies only to numeric expressions. Omit for a default of Signed Decimal or enter one of the following:

<table>
<thead>
<tr>
<th>l</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>Binary</td>
</tr>
<tr>
<td>Q</td>
<td>Octal</td>
</tr>
<tr>
<td>H</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>U</td>
<td>Unsigned decimal</td>
</tr>
<tr>
<td>D</td>
<td>Signed decimal</td>
</tr>
</tbody>
</table>

ACTION
Combines the XY and HPRINT commands. Prints numerical and string values at the selected x,y location on the color display. Several values are printed on the same line by separating them with commas. If an argument is a variable, the value of the variable is printed. Mathematical and string expressions are calculated before printing.

%0nl contains the format settings for numeric expressions and can be placed anywhere in the XYPRINT command. There can be several format settings in each XYPRINT command, each one changing the leading zero format, the numeric base and the field width for the expressions that follow. Format settings do not affect the printing of strings.

SEE ALSO
PRINT, PPRINT, PSSCREEN
EXAMPLES
U=44
T=37.83
STRING NOT
NOT="MOBILE PHONE DOES NOT RESPOND"

XYPRINT 5,5,%04,U,"",NOT // Contains a format setting
// specifying leading zeros and a width of
// four characters. Signed decimal is the
// numeric base by default. The command
// prints 0044 MOBILE PHONE DOES NOT RESPOND
// starting at point 5, 5 location on the
// color display.

XYPRINT 319,174,U,%3,"",T // Prints U as is starting in the
// middle of the screen. Then
// specifies no leading zeros, a
// field width of 3 and signed
// decimal (by default) for any
// following numerical outputs.
// A single space is printed.
// When T is printed, the print field
// is expanded to accommodate the
// value of T. Because each
// argument of the PRINT command
// (except for T) is followed by a
// comma, output is displayed on one
// line. The following is printed:
// 44 37.83

XYPRINT 0.0,%4B,U // Specifies no leading zeros, a field
// width of 4 and Binary numeric base for
// any following numerical outputs. The
// following is printed in the upper left
// corner of the screen: 101100.
// The field width is enlarged to
// accommodate U.

XYPRINT 30,55,NOT,%7,T // Specifies no leading zeros, a field
// width of 7 and signed decimal (by default)
// for any following numerical outputs.
// The following is printed starting at xy
// location (30,55):
// MOBILE PHONE DOES NOT RESPOND 37.83.
// 2 blanks character spaces before 3 are
// provided by the field width of 7.

XYPRINT 319,174,%3.1,T // Specifies no leading zeros, a
// minimum field width of 3 with a
// maximum of 1 digit after the
// decimal point. Displays 37.8
// starting in the middle of the
// screen. The field width of 3 was
// expanded to fit T, counting the
// decimal point.
This section explains the following procedures:

- How to establish communication between an RS-232 Terminal (typically a PC with a terminal emulation program) and the IFR-1900 CSA.
- How to create a TMAC program.
- How to upload a TMAC program into the IFR-1900 CSA working memory (RAM).
- How to execute the TMAC program.
- How to store the uploaded TMAC program into Flash Memory.
- How to execute a TMAC program stored in Flash Memory.
## 4-2 SETUP

### 4-2-1 RS-232 TERMINAL EMULATION PROGRAM SETUP

For the purpose of clarity and actual operation, the following procedures use Procomm Plus™ as the RS-232 Terminal Emulation program.

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Enter Procomm Plus™ from RS-232 terminal.</td>
</tr>
<tr>
<td>2.</td>
<td>Set Line Settings as follows:</td>
</tr>
<tr>
<td></td>
<td><strong>PARAMETER</strong></td>
</tr>
<tr>
<td></td>
<td>Baud Rate</td>
</tr>
<tr>
<td></td>
<td>Parity</td>
</tr>
<tr>
<td></td>
<td>Data Bits</td>
</tr>
<tr>
<td></td>
<td>Stop Bits</td>
</tr>
<tr>
<td></td>
<td>Port</td>
</tr>
<tr>
<td>3.</td>
<td>Set Terminal Options as follows:</td>
</tr>
<tr>
<td></td>
<td><strong>PARAMETER</strong></td>
</tr>
<tr>
<td></td>
<td>Terminal emulation</td>
</tr>
<tr>
<td></td>
<td>Duplex</td>
</tr>
<tr>
<td></td>
<td>Soft flow ctrl (Xon/Xoff)</td>
</tr>
<tr>
<td></td>
<td>Hard Flow ctrl (RTS/CTS)</td>
</tr>
<tr>
<td></td>
<td>Line wrap</td>
</tr>
<tr>
<td></td>
<td>Screen scroll</td>
</tr>
<tr>
<td></td>
<td>CR translation</td>
</tr>
<tr>
<td></td>
<td>BS translation</td>
</tr>
<tr>
<td></td>
<td>Break length (milliseconds)</td>
</tr>
<tr>
<td></td>
<td>Enquiry (ENQ)</td>
</tr>
<tr>
<td></td>
<td>EGA/VGA true underline</td>
</tr>
<tr>
<td></td>
<td>Terminal width</td>
</tr>
<tr>
<td></td>
<td>ANSI 7 or 8 bit commands</td>
</tr>
<tr>
<td>4.</td>
<td>Set ASCII Protocol Options as follows:</td>
</tr>
<tr>
<td></td>
<td><strong>PARAMETER</strong></td>
</tr>
<tr>
<td></td>
<td>Echo locally</td>
</tr>
<tr>
<td></td>
<td>Expand blank lines</td>
</tr>
<tr>
<td></td>
<td>Expand tabs</td>
</tr>
<tr>
<td></td>
<td>Character pacing (millisecond)</td>
</tr>
<tr>
<td></td>
<td>Line pacing (1/10 sec)</td>
</tr>
<tr>
<td></td>
<td>Pace character</td>
</tr>
<tr>
<td></td>
<td>Strip 8th bit</td>
</tr>
<tr>
<td></td>
<td>ASCII download timeout</td>
</tr>
<tr>
<td></td>
<td>CR translation (upload)</td>
</tr>
<tr>
<td></td>
<td>LF translation (upload)</td>
</tr>
<tr>
<td></td>
<td>CR translation (download)</td>
</tr>
<tr>
<td></td>
<td>LF translation (download)</td>
</tr>
</tbody>
</table>
### 4-2-2 HOST SETUP

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Perform &quot;Remote Operation Using Host System (RS-232)&quot; configuration procedure in Section 6 of the IFR-1900 Operation Manual (1002-3402-200). (Make sure Echo is toggled On.)</td>
</tr>
<tr>
<td>2.</td>
<td>Verify communication with HOST by pressing Enter key on RS-232 terminal keyboard and verify OK and exclamation mark appears on RS-232 terminal monitor.</td>
</tr>
</tbody>
</table>

### 4-2-3 SPECIAL TEST SETUP

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.</td>
<td>Verify communication with Special Test by pressing Enter key on RS-232 terminal keyboard and verify question mark appears on RS-232 terminal monitor.</td>
</tr>
</tbody>
</table>

### 4-3 CREATING A TMAC PROGRAM

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Write TMAC program using RS-232 terminal text editor. Refer to examples in Sections 2, 3 and 4 (e.g., Figure 4-2).</td>
</tr>
<tr>
<td>2.</td>
<td>Save as text file (program file).</td>
</tr>
</tbody>
</table>
4-4 UPLOADING TMAC PROGRAM

Uploading a TMAC program places the program into working memory (RAM) for immediate execution or for storing into Flash Memory.

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Enter Procomm Plus™ from RS-232 terminal.</td>
</tr>
<tr>
<td>2.</td>
<td>Issue *PMC command from RS-232 terminal to purge all existing files (TMAC programs).</td>
</tr>
<tr>
<td></td>
<td>The user should include the *PMC command at the beginning of all TMAC programs.</td>
</tr>
<tr>
<td>3.</td>
<td>Press PgUp key on RS-232 terminal and select ASCII upload protocol.</td>
</tr>
<tr>
<td>4.</td>
<td>Type in path and file name of text file (TMAC program) written in 4-3, and press Enter key.</td>
</tr>
<tr>
<td>5.</td>
<td>If failure occurs:</td>
</tr>
<tr>
<td></td>
<td>• Utilize Procomm Plus™ Log function to create separate text file (log file) of Upload function (program file with error messages).</td>
</tr>
<tr>
<td></td>
<td>• Repeat Steps 1 through 4.</td>
</tr>
<tr>
<td></td>
<td>• View log file and correct errors in original program file.</td>
</tr>
<tr>
<td></td>
<td>• Repeat Steps 1 through 4.</td>
</tr>
</tbody>
</table>

4-5 EXECUTING TMAC PROGRAM

After TMAC program is uploaded (or manually entered) into working memory (RAM), perform the following:

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Enter name of macro that is the entry point for TMAC program.</td>
</tr>
<tr>
<td>2.</td>
<td>Press Enter key of RS-232 Terminal keyboard to execute program.</td>
</tr>
</tbody>
</table>
/* MACRO NAME: minitl
   SYNTAX: minitl
   PURPOSE: Perform a Mobile Station originated call.
   MACRO: (The following example uses the predefined constants F1 and F6 [see Appendix A].)
*/

dmc "minitl",begin
var done=0
var key_code
do
   css:conf:user
   freq:band 1
   css:channel 334
   css:rflv 1 -60
   css:start
   css:call:type 0
   host ":keypad:soft"
   color 1,15
   keypad:label 1,"START"
   keypad:label 6,"Ret"
   color 1,11
   center "Mobile Init Call",0,5,640
do
   key_code=val(host? "syst:key?")
until (key_code=F6) or (key_code=F1) // Waits for valid key.
if key_code=6
   done=1
else
   keypad:label 1,"STOP"
   keypad:label 6,"=" // Erases F6 label.
   css:call:proc:moabin
   center "Place Call",0,35,640
do
   key_code=val(host? "syst:key?")
until (key_code=F1) or css:recc:status?
   erase:txt 0,35,640
   center "Assign Channel",0,35,640
if(key_code!=F1)
   css:call:proc:assign
   do
      key_code=val(host? "syst:key?")
until (key_code=F1) or (meas:stat> 5960) // Waits for somewhat valid SAT.
if(key_code!=F1)
   erase:txt 0,35,640
   center "Call Successfully",0,35,640
endif
endif
while key_code!=F1
   key_code=val(host? "syst:key?")
wend
endif
until done
end

Figure 4-2 Example TMAC Program
4-6 STORAGE/uploaded TMAC PROGRAM INTO FLASH MEMORY

Storing an uploaded TMAC program into flash memory allows TMAC program to remain in Test Set though power is removed or a *PMC (Purge Macro) command is issued.

Use MMEMory:STORe:MACRo "m","f" command to store TMAC program (uploaded in 4-4, plus any existing program files uploaded since last issue of *PMC or power-up) into Flash Memory:

- Set m to name (See Figure 4-3) of macro to be designated as entry point of uploaded TMAC program to run from Front Panel of the HOST.
- Set f to desired Flash File name (i.e., f is the name to appear in the Flash File Directory) of TMAC program.

Figure 4-4 shows the relationship between the macro name and the Flash File name displayed in Flash File Directory from the front panel of the HOST.

Refer to 6-15 (HOST) or 9-13 (Sp Tst) for more information on Flash Memory commands.

Examples: mmem:store:macro "minit1","minit1" / Stores file, written in 4-3 and and uploaded in 4-4, into Flash // Memory.

mmem:store:macro "sense_main","s_test"  // Stores file from 4-8 into Flash // Memory.

Figure 4-3 Macro Name Identification  Figure 4-4 Flash File and Macro Name Relationship
4-7 EXECUTING TMAC PROGRAM STORED IN FLASH MEMORY

4-7-1 FROM RS-232 TERMINAL

Issue the following command from RS-232 terminal:

\texttt{mmem:load:macr "m","f"}

Where: \( m \) is the name of the macro to be executing, and \( f \) is the file name of the program file stored in Flash Memory.

If \( m \) is ", then the macro previously designated by the \texttt{mmem:stor:macr} command as the entry point is executed.

4-7-2 FROM HOST FRONT PANEL

A. HOST TMAC Programs

Refer to the description of "User Program" in Section 3-3-10 in the IFR-1900 Operation Manual for further explanation of the following:

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Select desired Flash File from Flash Memory File Directory.</td>
</tr>
<tr>
<td>2.</td>
<td>Execute selected Flash File (macro).</td>
</tr>
</tbody>
</table>

B. Special Test Programs

Refer to "User Files" in Appendix C in the IFR-1900 CSA Option Operation Manual for a detailed explanation of the following:

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Select desired Flash File from 1900CSA Flash Files Directory (see Figure 4-4).</td>
</tr>
<tr>
<td>2.</td>
<td>Execute selected Flash File (macro).</td>
</tr>
</tbody>
</table>
4-8  EXAMPE TMAC PROGRAM WITH MULTIPLE MACROS

The example TMAC program in this section applies only to the Special Test.

Use MMEM:STOR:MACR "sense_main","s_test" to store the following TMAC program file into Flash Memory (after being uploaded).

The following TMAC program uses the predefined constants F1 and F6 (see Appendix A). TMAC program is available on the IFR BBS (316-524-0270).

/*
 * FILE NAME: stest.mac
 * PURPOSE: Demonstrate several capabilities of Sp Tst TMAC including:
 *   o Sp Tst TMAC Graphics
 *   o Cell Site Simulation
 *   o Using multiple macros
 *   o Storing macros in Flash Memory
 *   o Running a TMAC program from the Front Panel.
 *   o Using HOST TMAC commands through the HOST command.
 */

var key_code   // Declares a global variable.

/*
 * MACRO: delay1
 * SYNTAX: delay1 n
 * (n=delay time in ms.)
 * PURPOSE: Provide a delay if the F1 Soft Function Key is not pressed. */

*dmc "delay1",begin
if (key_code!=F1)
  delay $1
endif
end

/*
 * MACRO: focc_setup
 * SYNTAX: focc_setup
 * PURPOSE: Set up the System Parameter Overhead message sent on the Forward Control Channel. */

*dmc "focc_setup",begin
freq:band 1       // Sets operating band to U$ (800 MHz).
css:chan 333      // Sets Forward Control Channel to 333 (879.99 MHz).
css:rflev -60.0   // Sets output RF Level to -60.0 dBm.
css:focc:rcf 0     // Sets Read Control Filler bit.
css:focc:sid 0     // Sets System Identification Number.
css:focc:n 1       // Sets Number of Paging Channels.
css:focc:cmax 1    // Sets Number of Access Channels.
css:focc:auth 0    // Sets authentication bit.
css:start
end

(s_test.mac program file continues on following page.)
MACRO: call_setup
SYNTAX: call_setup
PURPOSE: Set up parameters used in making a digital call to the Mobile Station (cellular phone). */

dmc "call_setup", begin

css: call: type 1 // Sets call type to digital.
css: call: slot 2 // Sets to Timeslot 2.
css: call: pm 0 // Sets Privacy Mode bit.
css: call: mem 0 // Sets Message Encryption Mode to 0.
css: call: ef 0 // Sets Extended Protocol Forward Channel Indicator.
css: fdtc: enable: signal 1 // Enables Signal field.
css: fdtc: signal: pitch 0; cadence 1 // Sets pitch and pattern of Alert tone.
css: fdtc: enable: calling: num 1 // Enables Calling Party Number field.
css: fdtc: calling: type 0 // Sets Calling Party Type.
css: fdtc: calling: plan 0; pi 0; si 0 // Identification, Presentation Indicator and Screening Indicator.
css: fdtc: enable: dmac 0; ta 1; dtx 0; dic 0 // Disables Digital Mobile Attenuation Code, Discontinuous Transmission bit and Delay Interval. Enables Time Alignment field.
css: fdtc: ta 2 // Sets Time Alignment to 2.
end

MACRO: registration
SYNTAX: registration
PURPOSE: Force Mobile Station to register. */

dmc "registration", begin

center "Registering Mobile", 0, 35, 640
call_setup
do
css: call: proc: reg
delay 1000
key_code = val (host? "syst:key?")
until (key_code = F1) or css: recc: status?
print "MIN of mobile is ", recc: min? // Prints information out
print "ESN of mobile is ", recc: esn? // OPT. RS-232 Connector.
end

(s_test.mac program file continues on following page.)
/*
 * MACRO: pagel
 * SYNTAX: pagel
 * PURPOSE: Page Mobile Station until Page Response is received. */

*dmc 'pagel',begin
erase: text 0,35,640
center "Paging Mobile",0,35,640
css: call: proc: page
do
  delayl 1000
  key_code:=val(host? 'syst:key?')
until (key_code=F1) or css: recc: status?
erase: text 0,35,640
center "Page Response Received",0,35,640
end

/*
 * MACRO: assign
 * SYNTAX: assign
 * PURPOSE: Assign Mobile Station to a digital voice channel. */

*dmc "assign",begin
css: call: proc: assign
delayl 1000  // Waits for channel change.
erase: text 0,35,640
center "Digital Voice Channel Assigned",0,35,640
end

/*
 * MACRO: plc
 * SYNTAX: plc
 * PURPOSE: Send Physical Layer Control message to the Mobile Station
 * until a Physical Layer Control Ack message is received. */

*dmc 'plc',begin
do
css: fdtc: facch: plc
delayl 500
  key_code:=val(host? "syst:key" )
until ((key_code=F1) or (rdtc: facch: mesg? = "PLC ACK" ))
erase: text 0,35,640
center "Physical Layer Control",0,35,640
end

(s_test.mac program file continues on following page.)
MACRO: alert
SYNTAX: alert
PURPOSE: Send Alert message to Mobile Station until Alert with Info
         Ack is received. Wait until Connect message is received.
         */
dmc "alert", begin
   do
      css: fdtc: facch: alert
      delay1 500
      key_code = val(host? "syst: key?")
      until {(key_code = F1) or (rdtc: facch: amt? = "ALERT")}
      erase: text 0, 35, 640
      center "Answer Phone", 0, 35, 640
      while !(key_code = F1) and (rdtc: facch: msg? != "CONNECT")
         key_code = val(host? "syst: key?")
      wend
      erase: text 0, 35, 640
      center "Connect", 0, 35, 640
   end

(s_test.mac program file continues on following page.)
/ * MACRO: sense
 SYNTAX:  sense
 PURPOSE: Vary the output RF Level of the Sp Tst until the Mobile
 Station (phone) reports a BER of 2% to 4%. */

dmc "sense",begin
var timeout, ber
i=0
j=0
erase: text 0,35,640
center "Finding Sensitivity of Mobile",0,35,640
do
  css:fdtc:facch:meas // Sends Measurement Order until
  delay1 400 // Measurement Order Ack message is
  key_code=val(host? "syst:key?") // received.
  until (key_code=F1) or (rdtc:facch:msg?="MEAS ACK")
  css:rflvl -90 // Starts with RF Level set at -90 dBm.
  delay1 3000
  do
    timeout=25
    css:rflvl -90 - i
    erase: text 0,140,640
    xyprint 40,140,"RFLVL: ",%d,-90 - i
    while ((rdtc:sacch:msg? != "CHAN QUAL1") and (--timeout))
      delay1 100
    wend
    ber=rdtc:sacch:ber?
    erase: text 0,150,640
    xyprint 40,160,"BER: ",%d,ber
    if ber=0 // BER < 0.01%
      i=i+2
    elseif ber=1 // 0.01 < BER < 0.1
      i=i+1
    elseif ber=2 // 0.1 < BER < 0.5
      i=i+.2
    elseif ber=3 // 0.5 < BER < 1.0
      i=i+.1
    elseif ber=4 // 1.0 < BER < 2.0
      i=i+.1
    elseif ber=5 // 2.0 < BER < 4.0
      j=j+1
    elseif ber=6 // 4.0 < BER < 8.0
      i=i-.1
    else // BER > 8.0
      i=i-.2
    endif
    key_code=val(host? "syst:key?")
  until j=6 or !(timeout) or key_code=F1 // Continues when BER=5 is found
  // 6 times, timeout occurs or
  // F1 is pressed.

(s_test.mac program file and sense macro continues on following page.)
if key_code != F1  // Checks for STOP.
   print "Sensitivity of Mobile is ",%4d,-90-i  // Prints information out
   print ""  // OPT. RS-232 Connector.
   erase: text 0,35,640
   center "Test Done, Press End or STOP",0,35,640
   while (key_code != F1) and (rdtc:facch:msg? != "RELEASE")
      key_code=val(host? "syst: key?")
      delay1 100
   wend
   erase: text 0,35,640
   center "Test Completed",0,35,640
endif
css: stop
delay 500
end

(s_test.mac program file continues on following page.)
MACRO: sense_main
SYNTAX: sense_main
PURPOSE: Provide the main calling routine for the sensitivity test.

*dmc 'sense_main', begin
var done=0
do
  focc_setup
  user
  host ":keypad:soft"
  color 1,15
  keypad:label 1,"START"
  keypad:label 6,"Ret"
  color 1,11
  center 'Sensitivity Test',0,5,640
  do
    key_code=val(host? "syst: key?")
  until (key_code=F6) or (key_code=F1) // Waits for valid key.
  if key_code=F6
    done=1
    // Goes to end if Ret key is pressed.
  else
    keypad:label 1,"STOP"
    keypad:label 6,"*" // Erases F6 label.
    registration
    if(key_code != F1)
      pagel
      endif
    if(key_code != F1)
      assign
      endif
    if(key_code != F1)
      plc
      endif
    if(key_code != F1)
      alert
      endif
    if(key_code != F1)
      delay 3000
      endif
    if(key_code != F1)
      sense
      endif
    endif
  until done
end

(end of s_test.mac program file.)
This Quick Reference List is a brief listing of the Specific commands used with the HOST. The Quick Reference List is an aid to the experienced TMAC user. If more detailed information is needed, refer to the specified page.

Effort has been made to arrange all commands alphabetically; therefore some headings which are more descriptive than the actual commands, may appear out of alphabetical order.

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCESSORY COMMANDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACCessory:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INIT</td>
<td>6-123</td>
<td></td>
<td>Initializes MIC/ACC Connector prior to conducting communication. PTT (Press to Talk) is turned On (low).</td>
</tr>
<tr>
<td>IO</td>
<td>6-123</td>
<td></td>
<td>Sends/receives data through MIC/ACC Connector.</td>
</tr>
<tr>
<td>RELEASE</td>
<td>6-123</td>
<td></td>
<td>&quot;Releases&quot; MIC/ACC Connector. PTT is turned Off (high).</td>
</tr>
<tr>
<td>STATE?</td>
<td>6-123</td>
<td></td>
<td>Checks presence of MGC/ACC I/O device. 1 = device present. 0 = none.</td>
</tr>
</tbody>
</table>

AF GENERATOR COMMANDS
See Function Generator Commands.

AF LEVEL METER COMMAND
See M_VRMS?

AF METER COMMANDS
See M.AF:

ANALYZER (SPECTRUM ANALYZER) COMMANDS

ANLZ:
| AVERAGE [n] | 1 to 100 | 6-79 | Selects Average Mode for Analyzer using n samples. |
| CHANnel n   | 1 to 2047, depending on format | 6-79 | Sets RF Frequency to cellular channel. |

CHANnel BAND:
| 6-80 | Returns band for Analyzer Channel Format. Returns one of the following bands for the associated channel format. (NADC) U8, U4 or HY; (ETACS) NOT AVAILABLE; (NAMPS) LOWER, MIDDLE or UPPER. |

FORMat:
<p>| AMPs:        |             |      |                                                                             |
| FORward      | 6-79        |      | Selects AMPs (NADC-U8) Forward channels.                                  |
| REVerse      | 6-79        |      | Selects AMPs (NADC-U8) Reverse channels.                                  |
| ETACS:       |             |      |                                                                             |
| FORward      | 6-79        |      | Selects ETACS Forward channels.                                           |
| REVerse      | 6-79        |      | Selects ETACS Reverse channels.                                           |
| NADC:        |             |      |                                                                             |
| BAND:xx U8, U4 or HYper | 6-79 |      | Selects NADC band.                                                       |
| FORward      | 6-79        |      | Selects NADC Forward channels.                                            |
| REVerse      | 6-79        |      | Selects NADC Forward channels.                                            |
| NAMPS:       |             |      |                                                                             |
| BAND:x Lower, Middle or Upper | 6-80 |      | Selects NAMPS Narrow Analog channel designator.                           |
| FORward      | 6-80        |      | Selects NAMPS Forward channels.                                           |
| REVerse      | 6-80        |      | Selects NAMPS Reverse channels.                                           |
| NT400:       |             |      |                                                                             |
| FORward      | 6-80        |      | Selects NT400 Forward channels.                                           |
| REVerse      | 6-80        |      | Selects NT400 Reverse channels.                                           |</p>
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANLZ:COMPar e</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPare n</td>
<td>1 to 9</td>
<td>6-81</td>
<td>Selects Compare Mode for Analyzer. Trace stored at n memory location is compared with Live Trace.</td>
</tr>
<tr>
<td>FIND:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-81</td>
<td>Returns frequency of first signal with amplitude larger than Find reference level. Returns 0 if no signal is found.</td>
</tr>
<tr>
<td>REFe rence n</td>
<td>(varies)</td>
<td>6-81</td>
<td>Sets Find reference level in dB.</td>
</tr>
<tr>
<td>REFerence?</td>
<td></td>
<td>6-81</td>
<td>Returns Find reference level in dB.</td>
</tr>
<tr>
<td>FREQuency f</td>
<td>250.0 to 2010000.0</td>
<td>6-81</td>
<td>Sets Analyzer Frequency in kHz.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-81</td>
<td>Returns Analyzer Frequency in kHz.</td>
</tr>
<tr>
<td>FULL</td>
<td></td>
<td>6-81</td>
<td>Selects full size Analyzer display for RF Generator, Receiver and Duplex Operation Screens.</td>
</tr>
<tr>
<td>INPUT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANTenna</td>
<td></td>
<td>6-82</td>
<td>Selects ANTENNA IN Connector for Analyzer Input.</td>
</tr>
<tr>
<td>ATTenuation n</td>
<td>0, 5, 10, 15, 20, 25, 30</td>
<td>6-82</td>
<td>Sets Analyzer Input Attenuation in dB.</td>
</tr>
<tr>
<td>ATTenuation:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LNA</td>
<td></td>
<td>6-82</td>
<td>Sets Analyzer Input Attenuation to LNA.</td>
</tr>
<tr>
<td>LNA?</td>
<td></td>
<td>6-82</td>
<td>Returns current state of the Low Noise Amplifier.</td>
</tr>
<tr>
<td>ATTenuation?</td>
<td></td>
<td>6-82</td>
<td>Returns 1 if LNA is selected; 0 otherwise.</td>
</tr>
<tr>
<td>TR</td>
<td></td>
<td>6-82</td>
<td>Returns the current value of Input Attenuation.</td>
</tr>
<tr>
<td>INPUT?</td>
<td></td>
<td>6-82</td>
<td>Selects T/R Connector for the Analyzer Input.</td>
</tr>
<tr>
<td>LIVE</td>
<td></td>
<td>6-82</td>
<td>Returns Analyzer Input setting.</td>
</tr>
<tr>
<td>Marker:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AOFF</td>
<td></td>
<td>6-83</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>DELTA:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLtitude?</td>
<td></td>
<td>6-83</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-83</td>
<td>Returns difference between the two Marker positions in MHz.</td>
</tr>
<tr>
<td>POIN?</td>
<td></td>
<td>6-83</td>
<td>Returns difference between Marker positions in graticules with 100 graticules equal to the Analyzer display width.</td>
</tr>
<tr>
<td>TRACK b</td>
<td>1 or 0</td>
<td>6-83</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>MARKER1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLtitude?</td>
<td></td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-84</td>
<td>Returns amplitude of the Trace at the Marker 1 crossing.</td>
</tr>
<tr>
<td>POIN? n</td>
<td>1 to 100</td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>POINT?</td>
<td></td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>MARKER2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLtitude?</td>
<td></td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-84</td>
<td>Returns amplitude of the Trace at the Marker 2 crossing.</td>
</tr>
<tr>
<td>POIN? n</td>
<td>1 to 100</td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>POINT?</td>
<td></td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-84</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>MODE type</td>
<td></td>
<td>6-85</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>NORMALize</td>
<td></td>
<td>6-85</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>PEAK</td>
<td></td>
<td>6-85</td>
<td>Enables/disables Marker Tracking Feature. Tracking feature keeps Markers a constant distance apart.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>ANLZ:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLOT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRID</td>
<td></td>
<td>6-85</td>
<td>Draws Analyzer grid on attached plotter.</td>
</tr>
<tr>
<td>TRACE</td>
<td></td>
<td>6-85</td>
<td>Draws Analyzer trace on attached plotter.</td>
</tr>
<tr>
<td>UNITS</td>
<td></td>
<td>6-85</td>
<td>Draws Analyzer units on attached plotter.</td>
</tr>
<tr>
<td>QTR</td>
<td></td>
<td>6-85</td>
<td>Selects 1/4 size Analyzer display for RF Generator, Receiver</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and Duplex Operation Screens.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-85</td>
<td>Recalls Analyzer Trace and parameters stored in memory location n.</td>
</tr>
<tr>
<td>RLEVEL n</td>
<td>0 to 64</td>
<td>6-85</td>
<td>Sets reference or offset level in dB for Analyzer only in 2 dB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>scale.</td>
</tr>
<tr>
<td>RLEVEL?</td>
<td></td>
<td>6-85</td>
<td>Returns reference or offset value in dB for Analyzer in 2 dB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>scale.</td>
</tr>
<tr>
<td>SCALE n</td>
<td>2 or 10</td>
<td>6-85</td>
<td>Selects Analyzer Units/Division Factor in dB.</td>
</tr>
<tr>
<td>SCALE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T/R Analyzer Input:</td>
<td>6-86</td>
<td></td>
<td>Sets Analyzer Scale Units.  If DBW is selected, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>T/R Connector is selected for the Analyzer Input. If DBV,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DBMV, DBUV or DBUW is selected, the ANTENNA IN Connector is</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>selected for the Analyzer input.</td>
</tr>
<tr>
<td>ANTENNA Analyzer Input:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBM (dBm)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBW (dBW)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBM (dBm)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBMV (dBmV)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBUV (dBmV)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBV (dBV)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBUW (dBV)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT?</td>
<td></td>
<td>6-86</td>
<td>Returns Analyzer Scale Units.</td>
</tr>
<tr>
<td>SCALE?</td>
<td></td>
<td>6-86</td>
<td>Returns Analyzer Scale in dB.</td>
</tr>
<tr>
<td>SCAN n</td>
<td>0 (for zero scan), 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 50000, 100000</td>
<td>6-86</td>
<td>Selects Analyzer Scan Width in kHz.</td>
</tr>
<tr>
<td>SCAN?</td>
<td></td>
<td>6-86</td>
<td>Returns Analyzer Scan Width in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-86</td>
<td>Enables/disables Analyzer display in RF Generator, Receiver</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and Duplex Operation Screens.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-86</td>
<td>Stores current Analyzer Trace and environment (routings and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>settings) in memory location n.</td>
</tr>
<tr>
<td>TOP?</td>
<td></td>
<td>6-87</td>
<td>Returns top of screen value in current units. Spectrum</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Analyzer Operation Screen must be displayed.</td>
</tr>
</tbody>
</table>
**ANLZ:TRACE:DATA**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ANLZ:TRACE:</strong></td>
<td></td>
<td></td>
<td>For the following <strong>ANLZ:TRACE</strong> commands, the Analyzer display is divided into 400 positions horizontally (0 signifying the left edge of the display, 399 signifying the right edge of the display) and 255 values vertically (0 signifying the bottom of the display, 255 signifying the top of the display).</td>
</tr>
</tbody>
</table>
| DATA n,offset,value,...,value | n = 1 to 9  
offset = 0 to 399  
value = 0 to 255 | 6-87 | Replaces points of stored Trace n with specified vertical values (value) starting with horizontal offset (offset). Intended for remote GPIB or RS-232 use only; command and data are sent via GPIB or RS-232 from an external application, etc. |
| DATA? [[n],offset,points] | n = 0 to 9,  
0 = Live Trace  
(0 default),  
offset = 0 to 399  
(0 default)  
points = 1 to 400  
(400 default) | 6-87 | Returns the vertical values for each of the points specified, starting with offset (horizontal offset) of trace n. Intended for remote GPIB or RS-232 use only; data is sent out GPIB or RS-232 for use by an external application, etc. |
| GET name,n          | n = 0 to 9,  
0 = Live Trace. | 6-88 | Assigns vertical values of trace n (in graticules) to each element of declared array name. First element of array corresponds with first point of trace n. If array is less than 400 values in length, the remaining portion of trace is left unassigned to an array. See **ANLZ:TRACE:PUT**. |
| GET? n,offset       | n = 0 to 9,  
0 = Live Trace  
(0 default),  
offset = 0 to 399 | 6-88 | Returns vertical value of a point in Trace n located at offset (horizontal position from the left edge of display). |
| MAX? [[n],offset,points] | n = 0 to 9,  
0 = Live Trace  
(0 default),  
offset = 0 to 399  
(0 default)  
points = 1 to 400  
(400 default) | 6-88 | Returns the maximum point of Trace n within specified number of points starting with given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the live Trace or a Trace stored in memory. Intended for remote GPIB or RS-232 use only; data is sent out GPIB or RS-232 for use by an external application, etc. |
| MIN? [[n],offset,points] | n = 0 to 9,  
0 = Live Trace  
(0 default),  
offset = 0 to 399  
(0 default)  
points = 1 to 400  
(400 default) | 6-88 | Returns the minimum point of Trace n within specified number of points starting with the given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the Live Trace or a Trace stored in memory. Intended for remote GPIB or RS-232 use only; data is sent via GPIB or RS-232 from an external application, etc. |
| PUT name,n          | n = 1 to 9 | 6-89 | Assigns values of an array to trace n. name is array name. n signifies stored trace. Each element value represent a vertical value for associated horizontal positions. If array is less than 400 values in length (stored trace length), the array values are assigned to the trace the length of the array starting from horizontal position 0. The remaining portion of the stored trace is left intact. See **ANLZ:TRACE:GET**. |
ANLZ: TRACK: BWIDth

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANLZ:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRACK:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BWIDth f</td>
<td>3, 30, 300 or 3000</td>
<td>6-89</td>
<td>Sets Tracking Generator Bandwidth in kHz.</td>
</tr>
<tr>
<td>BWIDth?</td>
<td>6-89</td>
<td></td>
<td>Returns Tracking Generator Bandwidth in kHz.</td>
</tr>
<tr>
<td>LEVeL n</td>
<td>-137.0 to 0.0</td>
<td>6-89</td>
<td>Sets Tracking Generator Level in dBm.</td>
</tr>
<tr>
<td>LEVeL?</td>
<td>6-89</td>
<td></td>
<td>Returns Tracking Generator Level in dBm.</td>
</tr>
<tr>
<td>OUTput:</td>
<td>DUPlex</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTput?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESolution:</td>
<td>HIGH</td>
<td>6-90</td>
<td>Selects DUPLex OUT Connector as Tracking Generator Output Connector.</td>
</tr>
<tr>
<td>RESolution:</td>
<td>LOW</td>
<td>6-90</td>
<td>Selects T/R Connector as Tracking Generator Output Connector.</td>
</tr>
<tr>
<td>RESolution:</td>
<td>MED</td>
<td>6-90</td>
<td>Returns current Tracking Generator Output Connector.</td>
</tr>
<tr>
<td>RESolution?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-91</td>
<td>Enables/disables Tracking Generator.</td>
</tr>
<tr>
<td>STATe?</td>
<td></td>
<td>6-91</td>
<td>Returns 1 if Tracking Generator is active; 0 if inactive.</td>
</tr>
</tbody>
</table>

AUXILIARY TEST SET (SPECIAL TEST) COMMANDS

AHIT? 6-173 Returns 1 if there is input waiting from the Sp Tst; 0 otherwise.

AUX "string" 6-173 Issues commands, as strings, to the Sp Tst.

AUX? "string?" 6-173 Issues queries, as strings, to the Sp Tst.

BER (BIT ERROR RATE) METER COMMANDS

See M.BER:

CELLULAR AMPS/NAMPS COMMANDS

Queries for received data return -1 if data is not available or has already been read.

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CELL:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACTion?</td>
<td>6-126</td>
<td>Returns current Action field value.</td>
</tr>
<tr>
<td>B_.?</td>
<td>6-125</td>
<td>Returns current state of Busy/Idle bit.</td>
</tr>
<tr>
<td>BAND?</td>
<td>6-130</td>
<td>Returns NADC or NAMPS band. Returns one of the following: U8, U4, HY, L, M or U.</td>
</tr>
<tr>
<td>BIS?</td>
<td>6-127</td>
<td>Returns current state of BIS bit.</td>
</tr>
<tr>
<td>BOTH</td>
<td>6-125</td>
<td>Selects decoding of both Stream A and B words.</td>
</tr>
<tr>
<td>C12?</td>
<td>6-129</td>
<td>Returns current C12 value.</td>
</tr>
<tr>
<td>C13?</td>
<td>6-129</td>
<td>Returns current C13 value.</td>
</tr>
<tr>
<td>CAPTure:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIN &quot;xxxx/xxxx-xxxx&quot;</td>
<td>6-131</td>
<td>Specifies MIN capture value.</td>
</tr>
<tr>
<td>MIN?</td>
<td>6-131</td>
<td>Returns current MIN Capture value in the format: &quot;xxxx/xxxx-xxxx.&quot;</td>
</tr>
<tr>
<td>MODE:xxx</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODE?</td>
<td>6-131</td>
<td>Returns current capture mode: MIN, ORDER, BOTH or OFF.</td>
</tr>
</tbody>
</table>
**CELL: CAPTURE: ORDER**

<table>
<thead>
<tr>
<th>Command</th>
<th>Range</th>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CELL: CAPTURE: ORDER:xxx</td>
<td>PAGE, ALERT, RELease, REORDER, SALERT, AUDIT, SNDAddr, INTERCEPT, MAINTenance, POWer, DRETRY, AUTREG, AINTERCEPT, AREORDER, AALERT, VCDES</td>
<td>6-131</td>
<td>Specifies Order on which to capture.</td>
</tr>
<tr>
<td>ORDer?</td>
<td></td>
<td>6-131</td>
<td>Returns current Order on which to capture.</td>
</tr>
<tr>
<td>CHANnel n</td>
<td>1 to 2047, depending on format</td>
<td>6-125</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANnel?</td>
<td></td>
<td>6-125</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>CHANPOS1?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of first channel position field from directed retry message.</td>
</tr>
<tr>
<td>CHANPOS2?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of second channel position field from directed retry message.</td>
</tr>
<tr>
<td>CHANPOS3?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of third channel position field from directed retry message.</td>
</tr>
<tr>
<td>CHANPOS4?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of fourth channel position field from directed retry message.</td>
</tr>
<tr>
<td>CHANPOS5?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of fifth channel position field from directed retry message.</td>
</tr>
<tr>
<td>CHANPOS6?</td>
<td></td>
<td>6-127</td>
<td>Returns current value of sixth channel position field from directed retry message.</td>
</tr>
<tr>
<td>CMAC?</td>
<td></td>
<td>6-126</td>
<td>Returns current CMAC value.</td>
</tr>
<tr>
<td>CMAX_1?</td>
<td></td>
<td>6-126</td>
<td>Returns current CMAX-1 value.</td>
</tr>
<tr>
<td>CPA?</td>
<td></td>
<td>6-127</td>
<td>Returns current state of CPA bit.</td>
</tr>
<tr>
<td>DCC?</td>
<td></td>
<td>6-125</td>
<td>Returns current DCC value.</td>
</tr>
<tr>
<td>DIGItS?</td>
<td></td>
<td>6-128</td>
<td>Returns current Call address value.</td>
</tr>
<tr>
<td>DTX?</td>
<td></td>
<td>6-126</td>
<td>Returns current DTX bit.</td>
</tr>
<tr>
<td>E?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of E bit.</td>
</tr>
<tr>
<td>EF?</td>
<td></td>
<td>6-128</td>
<td>Returns current EF value.</td>
</tr>
<tr>
<td>END?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of END bit.</td>
</tr>
<tr>
<td>ESN?</td>
<td></td>
<td>6-128</td>
<td>Returns current ESN value.</td>
</tr>
<tr>
<td>FORMAT?</td>
<td></td>
<td>6-129</td>
<td>Sets channel format to AMPS (800 MHz).</td>
</tr>
<tr>
<td>AMPS</td>
<td></td>
<td>6-130</td>
<td>Sets NADC band to HY (Hyperband - 1900 MHz).</td>
</tr>
<tr>
<td>NADC</td>
<td></td>
<td>6-130</td>
<td>Sets NADC band to U4 (NT400© - 450 MHz).</td>
</tr>
<tr>
<td>BAND: HY</td>
<td></td>
<td>6-130</td>
<td>Sets NADC band to U8 (AMPS - 800 MHz).</td>
</tr>
<tr>
<td>U4</td>
<td></td>
<td>6-130</td>
<td></td>
</tr>
<tr>
<td>U8</td>
<td></td>
<td>6-130</td>
<td></td>
</tr>
<tr>
<td>NAMPS</td>
<td></td>
<td>6-130</td>
<td>Sets NAMPS band to Lower.</td>
</tr>
<tr>
<td>BAND: Lower</td>
<td></td>
<td>6-130</td>
<td>Sets NAMPS band to Middle.</td>
</tr>
<tr>
<td>Middle</td>
<td></td>
<td>6-130</td>
<td>Sets NAMPS band to Upper.</td>
</tr>
<tr>
<td>Upper</td>
<td></td>
<td>6-130</td>
<td>Sets channel format to NT400 (450 MHz).</td>
</tr>
<tr>
<td>NT400</td>
<td></td>
<td>6-129</td>
<td>Sets channel format to NT400 (450 MHz).</td>
</tr>
<tr>
<td>PCS</td>
<td></td>
<td>6-129</td>
<td>Sets channel format to PCS (1900 MHz).</td>
</tr>
<tr>
<td>FORMAT?</td>
<td></td>
<td>6-130</td>
<td>Returns cellular format. Returns one of the following: AMPS, NT400 or PCS.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------</td>
<td>------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>CHANnel n</td>
<td>1 to 2047, depending on format</td>
<td>6-132</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANnel?</td>
<td></td>
<td>6-132</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>CMAC n</td>
<td>0 to 7</td>
<td>6-132</td>
<td>Specifies Control Mobile Attenuation Code in Control-Filler Message.</td>
</tr>
<tr>
<td>CMAX n</td>
<td>1 to 128</td>
<td>6-132</td>
<td>Specifies Maximum number of channels in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>CMAX?</td>
<td></td>
<td>6-132</td>
<td>Returns current value of Maximum number of channels in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>CPA b</td>
<td>1 or 0</td>
<td>6-133</td>
<td>Enables/disables Combined Paging/Access bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>DCC n</td>
<td>0 to 3</td>
<td>6-133</td>
<td>Specifies Digital Color Code in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>DTX n</td>
<td>0 to 3</td>
<td>6-133</td>
<td>Specifies Discontinuous Transmission in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>E b</td>
<td>1 or 0</td>
<td>6-133</td>
<td>Enables/disables Extended Address bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>EP b</td>
<td>1 or 0</td>
<td>6-133</td>
<td>Enables/disables Extended Protocol bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>N n</td>
<td>1 to 32</td>
<td>6-134</td>
<td>Specifies Number of paging channels in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>N?</td>
<td></td>
<td>6-134</td>
<td>Returns current value of Number of paging channels in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>RCF b</td>
<td>1 or 0</td>
<td>6-134</td>
<td>Enables/disables Read Control Filler bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>RCF?</td>
<td></td>
<td>6-134</td>
<td>Returns current state of Read Control Filler bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>REGH b</td>
<td>1 or 0</td>
<td>6-134</td>
<td>Enables/disables Registration for Home Stations bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>REGR b</td>
<td>1 or 0</td>
<td>6-134</td>
<td>Enables/disables Registration for Roaming Mobile Phones bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>REGR?</td>
<td></td>
<td>6-134</td>
<td>Returns current state of Registration for Roaming Mobile Phones bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>S b</td>
<td>1 or 0</td>
<td>6-134</td>
<td>Enables/disables Serial Number bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>S?</td>
<td></td>
<td>6-134</td>
<td>Returns current state of Serial Number bit in System Parameter Overhead Message.</td>
</tr>
<tr>
<td>SEND</td>
<td></td>
<td>6-132</td>
<td>Begin transmitting System Parameter Overhead Message.</td>
</tr>
<tr>
<td>SETUP</td>
<td></td>
<td>6-132</td>
<td>Configures Test Set for FOCC Screen without going to screen.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CELL:GEN:FOCC:SID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SID n</td>
<td>0 to 32767</td>
<td>6-135</td>
<td>Specifies System Identification number in System Parameter Overhead Message</td>
</tr>
<tr>
<td>SID?</td>
<td></td>
<td>6-135</td>
<td>Returns current value of System Identification number in System Parameter Overhead Message</td>
</tr>
<tr>
<td>STOP</td>
<td></td>
<td>6-132</td>
<td>Stops transmitting System Parameter Overhead Message</td>
</tr>
<tr>
<td>WFOM b</td>
<td>1 or 0</td>
<td>6-135</td>
<td>Enables/disables state of Wait for Overhead Message bit in Control-Filler Message</td>
</tr>
<tr>
<td>WFOM?</td>
<td></td>
<td>6-135</td>
<td>Returns current state of Wait for Overhead Message bit in Control-Filler Message</td>
</tr>
<tr>
<td>FVC:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C12 b</td>
<td>1 or 0</td>
<td>6-149</td>
<td>Enables/disables C12 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>C12?</td>
<td></td>
<td>6-149</td>
<td>Returns current state of C12 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>C13 b</td>
<td>1 or 0</td>
<td>6-149</td>
<td>Enables/disables Mobile Station Control Message</td>
</tr>
<tr>
<td>C13?</td>
<td></td>
<td>6-149</td>
<td>Returns current state of C13 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>CHAN n</td>
<td>1 to 1024</td>
<td>6-149</td>
<td>Specifies voice channel to which call is assigned in Mobile Station Control Message</td>
</tr>
<tr>
<td>CHAN?</td>
<td></td>
<td>6-149</td>
<td>Returns current voice channel to which call is assigned in Mobile Station Control Message</td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>1 to 2047; depending on format</td>
<td>6-149</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td>6-149</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>CLI &quot;string&quot;</td>
<td>0-9, #, * and N (Null).</td>
<td>32 characters, max.</td>
<td>Specifies Call Line Identifier string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CLI?</td>
<td></td>
<td>6-149</td>
<td>Returns current Call Line Identifier string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>DSCC n</td>
<td>0 to 7</td>
<td>6-150</td>
<td>Specifies DSAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>DSCC?</td>
<td></td>
<td>6-150</td>
<td>Returns current value of DSAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>LOCAL n</td>
<td>0 to 31</td>
<td>6-150</td>
<td>Specifies LOCAL in Mobile Station Control Message</td>
</tr>
<tr>
<td>LOCAL?</td>
<td></td>
<td>6-150</td>
<td>Returns current value of LOCAL in Mobile Station Control Message</td>
</tr>
<tr>
<td>MSL n</td>
<td>0 to 31</td>
<td>6-150</td>
<td>Specifies Message Length in Mobile Station Control Message</td>
</tr>
<tr>
<td>MSL?</td>
<td></td>
<td>6-150</td>
<td>Returns current value of Message Length in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MST n</td>
<td>0 to 225</td>
<td>6-150</td>
<td>Specifies Message Type in Mobile Station Control Message</td>
</tr>
<tr>
<td>MST?</td>
<td></td>
<td>6-150</td>
<td>Returns current value of Message Type in Mobile Station Control Message.</td>
</tr>
<tr>
<td>NAMPS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BER n</td>
<td>0 to 127</td>
<td>6-153</td>
<td>Specifies number of allowable bit errors in Mobile Station Control Message.</td>
</tr>
<tr>
<td>BER?</td>
<td></td>
<td>6-153</td>
<td>Returns current number of allowable bit errors in Mobile Station Control Message</td>
</tr>
<tr>
<td>C12 b</td>
<td>1 or 0</td>
<td>6-154</td>
<td>Enables/disables C12 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>C12?</td>
<td></td>
<td>6-154</td>
<td>Returns current state of C12 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>C13 b</td>
<td>1 or 0</td>
<td>6-154</td>
<td>Enables/disables C13 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>C13?</td>
<td></td>
<td>6-154</td>
<td>Returns current state of C13 bit in Mobile Station Control Message</td>
</tr>
<tr>
<td>CHAN n</td>
<td>1 to 1024</td>
<td>6-153</td>
<td>Specifies voice channel to which call is assigned in Mobile Station Control Message (11 least significant bits).</td>
</tr>
<tr>
<td>CHAN?</td>
<td></td>
<td>6-153</td>
<td>Returns current voice channel to which call is assigned in Mobile Station Control Message (11 least significant bits).</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| CELL:GEN:FVC:CHAN
CELL:GEN:   |          |       |                                                                             |
| FVC:       |          |       |                                                                             |
| NAMPS:     |          |       |                                                                             |
| CHANNEL n  | 0 to 1023| 6-153 | Specifies Channel.                                                           |
| CHANNEL? x | LOWer, MIDDLE, UPPER | 6-153 | Specifies Band.                                                             |
| CLI "string" | 0-9, #, *, and N (null). max 32 characters, max | 6-154 | Returns current Channel.                                                   |
| CLI?        |          |       |                                                                             |
| CTYP b      | 1 or 0   | 6-154 | Enables/disables Channel Type indicator in Mobile Station Control Message.   |
| CTYP?       |          |       |                                                                             |
| DSAT n      | 0 to 6   | 6-155 | Specifies Digital Supervisory Audio Tone in Mobile Station Control Message.  |
| DSAT?       |          |       |                                                                             |
| DSCC n      | 0 to 7   | 6-154 | Specifies DSAT Color Code in Mobile Station Control Message.                |
| DSCC?       |          |       |                                                                             |
| LOCAL n     | 0 to 31  | 6-156 | Specifies LOCAL in Mobile Station Control Message.                          |
| LOCAL?      |          |       |                                                                             |
| MSL n       | 0 to 31  | 6-155 | Specifies Message Length in Mobile Station Control Message.                 |
| MSL?        |          |       |                                                                             |
| MST n       | 0 to 255 | 6-155 | Specifies Message Type in Mobile Station Control Message.                   |
| MST?        |          |       |                                                                             |
| NEXT        |          |       |                                                                             |
| O_E b       | 1 or 0   | 6-155 | Enables/disables Odd/Even bit in Mobile Station Control Message.            |
| O_E?        |          |       |                                                                             |
| ORDER:xxx   |          | 6-156 | Specifies Order in Mobile Station Control Message.                          |

Note: PAGE: SNDADDR, EVEN, SNDADDR, ODD, S, ALERT, AUDIT, MAINTENANCE, ALERT, RELEASE, FADE, SUSP, CALLED, ADDR, HANDOFF, CONFIRM, PWRLV, HANDOff, M, EXTENDED.
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CELL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMPS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDSCC / PDSCC?</td>
<td>0 to 7</td>
<td>6-156</td>
<td>Specifies Present DSAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>PWRLVL / PWRLVL?</td>
<td>0 to 7</td>
<td>6-156</td>
<td>Specifies Power Level.</td>
</tr>
<tr>
<td>RSSI / RSSI?</td>
<td>0 to 7</td>
<td>6-156</td>
<td>Specifies Received Signal Strength in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SEND SETUP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHORT_MESSAGE /</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;string&quot;</td>
<td>See IS-88, Appendix A for string details.</td>
<td>6-157</td>
<td>Specifies Short Message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SHORT MESSAGE?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VMAC / VMAC?</td>
<td>0 to 7</td>
<td>6-157</td>
<td>Specifies Voice Mobile Attenuation Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_MESSAGE /</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;string&quot;</td>
<td>See IS-88, Appendix A for string details.</td>
<td>6-157</td>
<td>Specifies Voice Mail message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE MESSAGE?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOICE_UNANSWERED /</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;nn&quot;</td>
<td>0 to 99</td>
<td>6-157</td>
<td>Specifies number of unanswered messages in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_UNANSWERED?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOICE URGENT /</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>ON or OFF</td>
<td>6-157</td>
<td>Turns on or off Urgent message identifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE URGENT?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORDER: xxx</td>
<td>PAGE, SNDADDR, S_ALERT, AUDIT, MAINTenance, ALERT, RELease, PWRLVL, HANDoff, NAMPS_CH_ASGN, EXTENDED</td>
<td>6-150</td>
<td>Specifies ORDER in Mobile Station Control Message.</td>
</tr>
<tr>
<td>PSCC / PSCC?</td>
<td>0 to 3</td>
<td>6-151</td>
<td>Specifies Present SAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>PWRLVL / PWRLVL?</td>
<td>0 to 7</td>
<td>6-151</td>
<td>Specifies Power Level.</td>
</tr>
<tr>
<td>REPEAT / REPEAT?</td>
<td>ON or OFF</td>
<td>6-148</td>
<td>Turns Repeat on or off.</td>
</tr>
<tr>
<td>SCC / SCC?</td>
<td>0 to 3</td>
<td>6-151</td>
<td>Specifies SAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SEND SETUP</td>
<td></td>
<td>6-148</td>
<td>Configures Test Set for AMPS Forward Voice Channel Message without going to screen.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CELL:GEN:FVC:SHORT_MESSAGE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHORT_MESSAGE &quot;string&quot;</td>
<td>See IS-88,</td>
<td>6-151</td>
<td>Specifies Short Message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td></td>
<td>Appendix A for string details.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHORT_MESSAGE?</td>
<td></td>
<td>6-151</td>
<td>Returns current Short Message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>STOP n</td>
<td>0 to 7</td>
<td>6-148</td>
<td>Stops transmission of Mobile Station Control Message.</td>
</tr>
<tr>
<td>VMAC n</td>
<td></td>
<td>6-151</td>
<td>Specifies Voice Mobile Attenuation Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VMAC?</td>
<td></td>
<td>6-152</td>
<td>Specifies Voice Mail message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_MESSAGE &quot;string&quot;</td>
<td>See IS-88,</td>
<td>6-152</td>
<td>Returns current Voice Mail message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td></td>
<td>Appendix A for string details.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOICE_MESSAGE?</td>
<td></td>
<td>6-152</td>
<td>Specifies number of unanswered messages in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_UNANSWERED &quot;nn&quot;</td>
<td>00 to 99</td>
<td>6-152</td>
<td>Returns current number of unanswered messages in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_UNANSWERED?</td>
<td></td>
<td>6-152</td>
<td>Turns on or off Urgent message identifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_URGENT:x</td>
<td>ON or OFF</td>
<td>6-152</td>
<td>Returns current state of Urgent message identifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>GLACT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACTION:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACCESS b</td>
<td>1 or 0</td>
<td>6-137</td>
<td>Enables/disables Access Attempt in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>BIS b</td>
<td>1 or 0</td>
<td>6-137</td>
<td>Enables/disables Access Type in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>BIS?</td>
<td></td>
<td>6-137</td>
<td>Returns current state of Access Type in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>LOCAL1 b</td>
<td>1 or 0</td>
<td>6-137</td>
<td>Enables/disables Local Control 1 in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>LOCAL1?</td>
<td></td>
<td>6-137</td>
<td>Returns current state of Local Control 1 in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>LOCAL2 b</td>
<td>1 or 0</td>
<td>6-137</td>
<td>Enables/disables Local Control 2 in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>LOCAL2?</td>
<td></td>
<td>6-137</td>
<td>Returns current state of Local Control 2 in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>NEWACC b</td>
<td>1 or 0</td>
<td>6-136</td>
<td>Enables/disables New Access channel set in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>OLC b</td>
<td>1 or 0</td>
<td>6-136</td>
<td>Enables/disables Overload Control in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>REGINCR b</td>
<td>1 or 0</td>
<td>6-136</td>
<td>Enables/disables Registration Increment in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>RESCAN b</td>
<td>1 or 0</td>
<td>6-136</td>
<td>Enables/disables Rescan in Global Action Overhead Message menu.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>BIS b</td>
<td>6-136</td>
<td>Specifies Channel.</td>
<td></td>
</tr>
<tr>
<td>BIS?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>1 to 2047, depending on format</td>
<td>6-136</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOCALcntl n</td>
<td>0 to 65535</td>
<td>6-138</td>
<td>Specifies Local Control in Global Action Overhead Message.</td>
</tr>
<tr>
<td>LOCALcntl?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXBusy:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OTHER n</td>
<td>0 to 15</td>
<td>6-138</td>
<td>Specifies Maximum number of Busy occurrences allowed for Other accesses in Global Action Overhead Message.</td>
</tr>
<tr>
<td>OTHER?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FGR n</td>
<td>0 to 15</td>
<td>6-138</td>
<td>Specifies Maximum number of Busy occurrences allowed for Page Responses in Global Action Overhead Message.</td>
</tr>
<tr>
<td>FGR?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXSitz:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OTHER n</td>
<td>0 to 15</td>
<td>6-139</td>
<td>Specifies Maximum number of Seizure attempts allowed for Other accesses in Global Action Overhead Message.</td>
</tr>
<tr>
<td>OTHER?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FGR n</td>
<td>0 to 15</td>
<td>6-139</td>
<td>Specifies Maximum number of Seizure attempts allowed for Page Responses in Global Action Overhead Message.</td>
</tr>
<tr>
<td>FGR?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NEWACC n</td>
<td>0 to 2047</td>
<td>6-139</td>
<td>Specifies New Access Channel starting point in Global Action Overhead Message.</td>
</tr>
<tr>
<td>NEWACC?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OLC n</td>
<td>0 to 32767</td>
<td>6-139</td>
<td>Specifies Overload Control Class in Global Action Overhead Message.</td>
</tr>
<tr>
<td>OLC?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGINCR n</td>
<td>0 to 4095</td>
<td>6-139</td>
<td>Returns current value of Overload Control Class in Global Action Overhead Message.</td>
</tr>
<tr>
<td>REGINCR?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REPEAT,x</td>
<td>ON or OFF</td>
<td>6-135</td>
<td>Turns Repeat on or off for Global Action Overhead Message.</td>
</tr>
<tr>
<td>REPEAT?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEND</td>
<td></td>
<td>6-135</td>
<td>Returns current state of Repeat for Global Action Overhead Message.</td>
</tr>
<tr>
<td>SETUP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STOP</td>
<td></td>
<td>6-135</td>
<td>Stops Global Action Overhead Message from being sent with System Parameter Overhead Message.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------</td>
<td>-------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>C12 b</td>
<td>1 or 0</td>
<td>6-141</td>
<td>Enables/disables C12 bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>C12?</td>
<td></td>
<td>6-141</td>
<td>Returns current state of C12 bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>C13 b</td>
<td>1 or 0</td>
<td>6-141</td>
<td>Enables/disables C13 bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>C13?</td>
<td></td>
<td>6-141</td>
<td>Returns current state of C13 bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CHAN n</td>
<td>0 to 1024</td>
<td>6-141</td>
<td>Specifies voice channel to which call is assigned in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CHAN?</td>
<td></td>
<td>6-141</td>
<td>Returns current voice channel assignment for call in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>1 to 2047, depending on format</td>
<td>6-140</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td>6-140</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>CHANPos x,y</td>
<td>1 to 6, 0 to 127</td>
<td>6-141</td>
<td>Sets position of a control channel relative to first access channel in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CHANPos? n</td>
<td>1 to 6</td>
<td>6-141</td>
<td>Returns current position of a control channel relative to first access channel in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CLI &quot;string&quot;</td>
<td>0-9, #, * and N (Null), 32 characters, max</td>
<td>6-141</td>
<td>Specifies Call Line Identifier string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CLI?</td>
<td></td>
<td>6-141</td>
<td>Returns current Call Line Identifier string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>DSCC n</td>
<td>0 to 7</td>
<td>6-142</td>
<td>Specifies DSAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>DSCC?</td>
<td></td>
<td>6-142</td>
<td>Returns current value of DSAT Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>EF b</td>
<td>1 or 0</td>
<td>6-142</td>
<td>Enables/disables Extended Protocol Forward Channel Indicator bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>EF?</td>
<td></td>
<td>6-142</td>
<td>Returns current state of Extended Protocol Forward Channel Indicator bit in Mobile Station Control Message.</td>
</tr>
<tr>
<td>LOCAL n</td>
<td>0 to 31</td>
<td>6-142</td>
<td>Specifies LOCAL in Mobile Station Control Message.</td>
</tr>
<tr>
<td>LOCAL?</td>
<td></td>
<td>6-142</td>
<td>Returns current value of LOCAL in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MIN &quot;xxx/xxx-xxxx&quot;</td>
<td>0-9, # and *</td>
<td>6-142</td>
<td>Specifies Mobile Identification Number in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MIN?</td>
<td></td>
<td>6-142</td>
<td>Returns current Mobile Identification Number in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MSL n</td>
<td>0 to 31</td>
<td>6-142</td>
<td>Specifies Message Length in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MSL?</td>
<td></td>
<td>6-142</td>
<td>Returns current value of Message Length in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MST n</td>
<td>0 to 255</td>
<td>6-143</td>
<td>Specifies Message Type in Mobile Station Control Message.</td>
</tr>
<tr>
<td>MST?</td>
<td></td>
<td>6-143</td>
<td>Returns current value of Message Type in Mobile Station Control Message.</td>
</tr>
<tr>
<td>ORDER:xxx</td>
<td>AUDIT, LC, DIR, RTRY, INTRCPT, RELease, REORDER, VC, DES, EXTENDEd</td>
<td>6-143</td>
<td>Specifies Order in Mobile Station Control Message.</td>
</tr>
<tr>
<td>ORDQ n</td>
<td>0 to 7</td>
<td>6-143</td>
<td>Specifies Order Qualifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>ORDQ?</td>
<td></td>
<td>6-143</td>
<td>Returns current value of Order Qualifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>REPEAT:x</td>
<td>ON or OFF</td>
<td>6-140</td>
<td>Sets Repeat on or off in Mobile Station Control Message.</td>
</tr>
<tr>
<td>REPEAT?</td>
<td></td>
<td>6-140</td>
<td>Returns current state of Repeat in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SCC n</td>
<td>0 to 3</td>
<td>6-143</td>
<td>Specifies Supervisory Audio Tone Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SCC?</td>
<td></td>
<td>6-143</td>
<td>Returns current value of Supervisory Audio Tone Color Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>SEND</td>
<td></td>
<td>6-140</td>
<td>Sends Mobile Station Control Message.</td>
</tr>
<tr>
<td>SETUP</td>
<td></td>
<td>6-140</td>
<td>Configures Test Set for Mobile Station Control Message Screen without going to screen.</td>
</tr>
<tr>
<td>SHORT_MESSAGE &quot;string&quot;</td>
<td>See IS-88, Appendix A for string details.</td>
<td>6-143</td>
<td>Specifies Short Message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>STOP</td>
<td></td>
<td>6-140</td>
<td>Stops transmission of Mobile Station Control Message.</td>
</tr>
<tr>
<td>VMAC n</td>
<td>0 to 7</td>
<td>6-144</td>
<td>Specifies Voice Mobile Attenuation Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VMAC?</td>
<td></td>
<td>6-144</td>
<td>Returns current value of Voice Mobile Attenuation Code in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_MESSAGE &quot;string&quot;</td>
<td>See IS-88, Appendix A for string details.</td>
<td>6-144</td>
<td>Specifies Voice Mail message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_MESSAGE?</td>
<td></td>
<td>6-144</td>
<td>Returns current Voice Mail message string in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_UNANSWERED &quot;nn&quot;</td>
<td>00 to 99</td>
<td>6-144</td>
<td>Specifies number of unanswered messages in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_UNANSWERED?</td>
<td></td>
<td>6-144</td>
<td>Returns current number of unanswered messages in Mobile Station Control Message.</td>
</tr>
<tr>
<td>VOICE_URGENT:x</td>
<td>ON or OFF</td>
<td>6-144</td>
<td>Turns on or off Urgent message identifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>RECO:</td>
<td></td>
<td>6-144</td>
<td>Returns current state of Urgent message identifier in Mobile Station Control Message.</td>
</tr>
<tr>
<td>CALLED_ADDRESS &quot;string&quot;</td>
<td>0-9, #, and *, 32 characters, max.</td>
<td>6-145</td>
<td>Specifies Called Address.</td>
</tr>
<tr>
<td>CALLED_ADDRESS?</td>
<td></td>
<td>6-145</td>
<td>Returns current Called Address.</td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>0 to 1023</td>
<td>6-145</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td>6-145</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>DCC n</td>
<td>0 to 3</td>
<td>6-146</td>
<td>Specifies Digital Color Code.</td>
</tr>
<tr>
<td>E b</td>
<td>1 or 0</td>
<td>6-146</td>
<td>Enables/disables Extended Address bit.</td>
</tr>
<tr>
<td>E?</td>
<td></td>
<td>6-146</td>
<td>Enables/disables Extended Address bit.</td>
</tr>
<tr>
<td>EP b</td>
<td>1 or 0</td>
<td>6-146</td>
<td>Enables/disables Extended Protocol bit.</td>
</tr>
<tr>
<td>ER b</td>
<td>1 or 0</td>
<td>6-146</td>
<td>Enables/disables Extended Protocol Reverse Channel bit.</td>
</tr>
<tr>
<td>ER?</td>
<td></td>
<td>6-146</td>
<td>Enables/disables Extended Protocol Reverse Channel bit.</td>
</tr>
<tr>
<td>ESN &quot;string&quot;</td>
<td>0 to 9, 11 digits, max.</td>
<td>6-146</td>
<td>Specifies Electronic Serial Number.</td>
</tr>
<tr>
<td>ESN?</td>
<td></td>
<td>6-146</td>
<td>Returns current Electronic Serial Number.</td>
</tr>
<tr>
<td>LOCAL n</td>
<td>0 to 31</td>
<td>6-146</td>
<td>Specifies LOCAL.</td>
</tr>
<tr>
<td>LOCAL?</td>
<td></td>
<td>6-146</td>
<td>Returns current value of LOCAL.</td>
</tr>
<tr>
<td>LT b</td>
<td>1 or 0</td>
<td>6-146</td>
<td>Enables/disables Last Try bit.</td>
</tr>
<tr>
<td>LT?</td>
<td></td>
<td>6-146</td>
<td>Returns current state of Last Try bit.</td>
</tr>
<tr>
<td>MIN &quot;xxx/xxx-xxxx&quot;</td>
<td>0-9, # and *</td>
<td>6-147</td>
<td>Specifies Mobile Identification Number.</td>
</tr>
<tr>
<td>MIN?</td>
<td></td>
<td>6-147</td>
<td>Returns current Mobile Identification Number.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>MSL n</td>
<td>0 to 31</td>
<td>6-147</td>
<td>Specifies Message Length.</td>
</tr>
<tr>
<td>MSL?</td>
<td></td>
<td>6-147</td>
<td>Returns current value of Message Length.</td>
</tr>
<tr>
<td>MST n</td>
<td>0 to 255</td>
<td>6-147</td>
<td>Specifies Extended Protocol Message Type.</td>
</tr>
<tr>
<td>MST?</td>
<td></td>
<td>6-147</td>
<td>Returns current value of Extended Protocol Message Type.</td>
</tr>
<tr>
<td>ORDER n</td>
<td>0 to 31</td>
<td>6-147</td>
<td>Specifies Order.</td>
</tr>
<tr>
<td>ORDER?</td>
<td></td>
<td>6-147</td>
<td>Returns current value of Order.</td>
</tr>
<tr>
<td>ORDO n</td>
<td>0 to 7</td>
<td>6-147</td>
<td>Specifies Order Qualifier.</td>
</tr>
<tr>
<td>ORDO?</td>
<td></td>
<td>6-147</td>
<td>Returns current value of Order Qualifier.</td>
</tr>
<tr>
<td>REPEAT:x</td>
<td>ON or OFF</td>
<td>6-145</td>
<td>Turns Repeat on or off.</td>
</tr>
<tr>
<td>S b</td>
<td>1 or 0</td>
<td>6-145</td>
<td>Returns current state of Repeat.</td>
</tr>
<tr>
<td>S?</td>
<td></td>
<td>6-147</td>
<td>Enables/disables Serial Number bit.</td>
</tr>
<tr>
<td>SCM n</td>
<td>0 to 15</td>
<td>6-147</td>
<td>Returns current state of Serial Number bit.</td>
</tr>
<tr>
<td>SCM?</td>
<td></td>
<td>6-148</td>
<td>Specifies Station Class Mark.</td>
</tr>
<tr>
<td>SEND</td>
<td></td>
<td>6-148</td>
<td>Returns current value of Station Class Mark.</td>
</tr>
<tr>
<td>SETUP</td>
<td></td>
<td>6-145</td>
<td>Sends Reverse Control Channel message.</td>
</tr>
<tr>
<td>STOP</td>
<td></td>
<td>6-145</td>
<td>Configures Test Set for Reverse Control Channel Screen without going to screen.</td>
</tr>
<tr>
<td>T b</td>
<td>1 or 0</td>
<td>6-145</td>
<td>Stops transmission of Reverse Control Channel message.</td>
</tr>
<tr>
<td>T?</td>
<td></td>
<td>6-148</td>
<td>Enables/disables Type of message bit.</td>
</tr>
<tr>
<td>RVC:</td>
<td></td>
<td>6-148</td>
<td>Returns current state of Type of message bit.</td>
</tr>
<tr>
<td>CALLED_ADDRess &quot;string&quot;</td>
<td>0-9, # and *</td>
<td>32 characters, max</td>
<td>6-159</td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>0 to 1023</td>
<td>6-158</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td>6-158</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>LOCAL n</td>
<td>0 to 31</td>
<td>6-158</td>
<td>Specifies LOCAL in Order Confirmation Message.</td>
</tr>
<tr>
<td>LOCAL?</td>
<td></td>
<td>6-158</td>
<td>Returns current value of LOCAL in Order Confirmation Message.</td>
</tr>
<tr>
<td>MESSAGE:xxx</td>
<td>ORDER_ CONFIRMation, CALLED_ADDRess, EXTENdEd</td>
<td>6-159</td>
<td>Specifies Message to be transmitted.</td>
</tr>
<tr>
<td>MSL n</td>
<td>0 to 31</td>
<td>6-159</td>
<td>Specifies Message Length in Extended Protocol Message.</td>
</tr>
<tr>
<td>MSL?</td>
<td></td>
<td>6-159</td>
<td>Returns current value of Message Length in Extended Protocol Message.</td>
</tr>
<tr>
<td>MST n</td>
<td>0 to 255</td>
<td>6-159</td>
<td>Specifies Message Type in Extended Protocol Message.</td>
</tr>
<tr>
<td>MST?</td>
<td></td>
<td>6-159</td>
<td>Returns current value of Message Type in Extended Protocol Message.</td>
</tr>
<tr>
<td>NAMPS:</td>
<td></td>
<td>6-160</td>
<td>Specifies number of Bit Errors in MRI (Mobile Reported Interference) Order Message.</td>
</tr>
<tr>
<td>BER n</td>
<td>0 to 127</td>
<td>6-160</td>
<td>Specifies number of Bit Errors in MRI (Mobile Reported Interference) Order Message.</td>
</tr>
<tr>
<td>BER?</td>
<td></td>
<td>6-160</td>
<td>Specifies number of Bit Errors in MRI (Mobile Reported Interference) Order Message.</td>
</tr>
<tr>
<td>CALLED_ADDRess &quot;string&quot;</td>
<td>0-9, # and *</td>
<td>32 characters, max</td>
<td>6-160</td>
</tr>
<tr>
<td>CALLED_ADDRess?</td>
<td></td>
<td></td>
<td>6-160</td>
</tr>
<tr>
<td>CHANNEL n</td>
<td>1 to 2047, depending on format</td>
<td>6-160</td>
<td>Specifies Channel.</td>
</tr>
<tr>
<td>CHANNEL:x</td>
<td>LOWer, MIDdle, UPper</td>
<td>6-160</td>
<td>Specifies Band.</td>
</tr>
<tr>
<td>CHANNEL?</td>
<td></td>
<td>6-160</td>
<td>Returns current Channel.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>CELL:GEN:RVC:NAMPS:CONFIRMation b</td>
<td>1 = Order Confirmation Message</td>
<td>6-161</td>
<td>Enables/disables T bit in Reverse Voice Channel Message to specify that message is a Order Confirmation Message or an Order.</td>
</tr>
<tr>
<td>CONFIRMation?</td>
<td></td>
<td></td>
<td>6-161</td>
</tr>
<tr>
<td>DSAT n</td>
<td>0 to 6</td>
<td>6-161</td>
<td>Specifies Digital Supervisory Audio Tone.</td>
</tr>
<tr>
<td>DSAT?</td>
<td>*</td>
<td>6-161</td>
<td>Returns current value of Digital Supervisory Audio Tone.</td>
</tr>
<tr>
<td>LOCAL n</td>
<td>0 to 31</td>
<td>6-161</td>
<td>Specifies LOCAL for Order Message or Order Confirmation Message.</td>
</tr>
<tr>
<td>LOCAL?</td>
<td>*</td>
<td>6-161</td>
<td>Returns current value of LOCAL for Order Message or Order Confirmation Message.</td>
</tr>
<tr>
<td>MESSAGE:xxx</td>
<td>MRI, ORDER, FLASH</td>
<td>6-161</td>
<td>Specifies message to transmitted.</td>
</tr>
<tr>
<td>NEXT</td>
<td>*</td>
<td>6-160</td>
<td>Sends next word of Mobile Station Control Message, if required.</td>
</tr>
<tr>
<td>O_E b</td>
<td>1 or 0</td>
<td>6-161</td>
<td>Enables/disables Odd/Even bit.</td>
</tr>
<tr>
<td>O_E?</td>
<td>*</td>
<td>6-161</td>
<td>Returns current state of Odd/Even bit.</td>
</tr>
<tr>
<td>ORDER n</td>
<td>0 to 31</td>
<td>6-162</td>
<td>Specifies ORDER for order messages.</td>
</tr>
<tr>
<td>ORDER?</td>
<td>*</td>
<td>6-162</td>
<td>Returns current value of ORDER for order messages.</td>
</tr>
<tr>
<td>ORDO n</td>
<td>0 to 7</td>
<td>6-161</td>
<td>Specifies Order Qualifier for order messages.</td>
</tr>
<tr>
<td>ORDO?</td>
<td>*</td>
<td>6-162</td>
<td>Returns current value of Order Qualifier for order messages.</td>
</tr>
<tr>
<td>RSSI n</td>
<td>0 to 7</td>
<td>6-162</td>
<td>Specifies Received Signal Strength in MRI (Mobile Reported Interference) Order Message.</td>
</tr>
<tr>
<td>RSSI?</td>
<td>*</td>
<td>6-162</td>
<td>Returns current value of Received Signal Strength in MRI (Mobile Reported Interference) Order Message.</td>
</tr>
<tr>
<td>SEND</td>
<td>*</td>
<td>6-160</td>
<td>Sends message.</td>
</tr>
<tr>
<td>SETUP</td>
<td>*</td>
<td>6-160</td>
<td>Configures Test Set for NAMPS Reverse Voice Channel Screen without going to screen.</td>
</tr>
<tr>
<td>VMAC n</td>
<td>0 to 7</td>
<td>6-162</td>
<td>Specifies Voice Mobile Attenuation Code in Order Messages.</td>
</tr>
<tr>
<td>ORDER n</td>
<td>0 to 31</td>
<td>6-159</td>
<td>Specifies ORDER in Order Confirmation Message.</td>
</tr>
<tr>
<td>ORDER?</td>
<td>*</td>
<td>6-159</td>
<td>Returns current value of ORDER in Order Confirmation Message.</td>
</tr>
<tr>
<td>ORDO n</td>
<td>0 to 7</td>
<td>6-159</td>
<td>Specifies Order Qualifier in Order Confirmation Message.</td>
</tr>
<tr>
<td>ORDO?</td>
<td>*</td>
<td>6-159</td>
<td>Returns current value of Order Qualifier in Order Confirmation Message.</td>
</tr>
<tr>
<td>REPEAT:x</td>
<td>*</td>
<td>5-158</td>
<td>Turns Repeat on or off.</td>
</tr>
<tr>
<td>REPEAT?</td>
<td>*</td>
<td>5-158</td>
<td>Returns current state of Repeat.</td>
</tr>
<tr>
<td>SEND</td>
<td>*</td>
<td>5-158</td>
<td>Sends RVC message.</td>
</tr>
<tr>
<td>SETUP</td>
<td>*</td>
<td>5-158</td>
<td>Configures Test Set for RVC Reverse Voice Channel Screen without going to screen.</td>
</tr>
<tr>
<td>STOP</td>
<td>*</td>
<td>6-158</td>
<td>Stops transmission of RVC message.</td>
</tr>
<tr>
<td>LOCALCTRL1?</td>
<td>*</td>
<td>6-129</td>
<td>Returns current first position field for Local Control value.</td>
</tr>
<tr>
<td>LOCALCTRL2?</td>
<td>*</td>
<td>6-129</td>
<td>Returns current second position field for Local Control value.</td>
</tr>
<tr>
<td>MAXbusy?</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Busy for Other access.</td>
</tr>
<tr>
<td>OTHER?</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Busy for page response.</td>
</tr>
<tr>
<td>PGR?</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Seizure Tries for Other access.</td>
</tr>
<tr>
<td>MAXSzt:</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Seizure Tries for page response.</td>
</tr>
<tr>
<td>OTHER?</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Seizure Tries for Other access.</td>
</tr>
<tr>
<td>PGR?</td>
<td>*</td>
<td>6-128</td>
<td>Returns current value of Maximum Seizure Tries for page response.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>CELL:</td>
<td></td>
<td>6-125</td>
<td>Returns current MIN value. Returning format is: &quot;xxx/xxx.xxx.&quot;</td>
</tr>
<tr>
<td>MIN?</td>
<td></td>
<td>6-129</td>
<td>Returns current MST value.</td>
</tr>
<tr>
<td>MSL?</td>
<td></td>
<td>6-129</td>
<td>Returns current MST value.</td>
</tr>
<tr>
<td>MST?</td>
<td></td>
<td>6-126</td>
<td>Returns current N-1 value.</td>
</tr>
<tr>
<td>N.1?</td>
<td></td>
<td>6-126</td>
<td>Returns current NAWC value.</td>
</tr>
<tr>
<td>NAWC?</td>
<td></td>
<td>6-127</td>
<td>Returns current NEWACC value.</td>
</tr>
<tr>
<td>NEWACC?</td>
<td></td>
<td>6-127</td>
<td>Returns current OLC value.</td>
</tr>
<tr>
<td>OLC?</td>
<td></td>
<td>6-125</td>
<td>Returns current ORDER value.</td>
</tr>
<tr>
<td>ORDER?</td>
<td></td>
<td>6-128</td>
<td>Returns current PDSGC value.</td>
</tr>
<tr>
<td>PDSGC?</td>
<td></td>
<td>6-128</td>
<td>Returns current PSCO value.</td>
</tr>
<tr>
<td>PSCO?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of RCF bit value.</td>
</tr>
<tr>
<td>RCF?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of REGH bit.</td>
</tr>
<tr>
<td>REGH?</td>
<td></td>
<td>6-127</td>
<td>Returns current REGINCR value.</td>
</tr>
<tr>
<td>REGINCR?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of REGR bit.</td>
</tr>
<tr>
<td>REGR?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of S bit.</td>
</tr>
<tr>
<td>S?</td>
<td></td>
<td>6-125</td>
<td>Returns current SCC value.</td>
</tr>
<tr>
<td>ECC?</td>
<td></td>
<td>6-125</td>
<td>Returns current SID value.</td>
</tr>
<tr>
<td>SID?</td>
<td></td>
<td>6-126</td>
<td>Returns current Voice Channel number.</td>
</tr>
<tr>
<td>VCHAN?</td>
<td></td>
<td>6-126</td>
<td>Returns current VMAC value.</td>
</tr>
<tr>
<td>VMAC?</td>
<td></td>
<td>6-126</td>
<td>Returns current state of WFOM bit.</td>
</tr>
<tr>
<td>WFOM?</td>
<td></td>
<td>6-125</td>
<td>Returns current decoding selection: WORDA, WORDB or BOTH.</td>
</tr>
<tr>
<td>WORDA</td>
<td></td>
<td>6-125</td>
<td>Selects decoding of Stream A words.</td>
</tr>
<tr>
<td>WORDB</td>
<td></td>
<td>6-125</td>
<td>Selects decoding of Stream B words.</td>
</tr>
</tbody>
</table>

**DEVIEATION METER (PEAK) COMMANDS**
See M_DEV:

**DEVIEATION METER (RMS) COMMANDS**
See M_DRMS:

**DIGITAL MULTIMETER COMMANDS**
See M_DMM:

**DISTORTION METER COMMANDS**
See M_DIST:

**DUPLEX COMMANDS**

**DUPlex:**

**INPut:**

<table>
<thead>
<tr>
<th>AGC:</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO</td>
<td></td>
<td>6-41</td>
<td>Sets Automatic Gain Control to automatic.</td>
</tr>
<tr>
<td>MANual n</td>
<td>0 to 255</td>
<td>6-41</td>
<td>Sets Automatic Gain Control to manual and sets level to n.</td>
</tr>
<tr>
<td>USER:xxx</td>
<td>MEASURE, SPEECH, DATA, HIGH, TYPE1, TYPE2, or TYPE3</td>
<td>6-41</td>
<td>Sets Automatic Gain Control to User setting.</td>
</tr>
</tbody>
</table>

**ANTenna**

<table>
<thead>
<tr>
<th>ATTenuation n</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0, 10, 15, 20, 25, 30</td>
<td>6-41</td>
<td>Sets Duplex Input Attenuation in dB.</td>
<td></td>
</tr>
</tbody>
</table>

**LNA**

<table>
<thead>
<tr>
<th>ATTenuation?</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LNA?</td>
<td>6-41</td>
<td></td>
<td>Sets Duplex Input Attenuation to LNA.</td>
</tr>
<tr>
<td></td>
<td>6-41</td>
<td></td>
<td>Returns current state of Low Noise Amplifier. Returns 1 if LNA is selected; 0 otherwise.</td>
</tr>
</tbody>
</table>

**RETURN**

<table>
<thead>
<tr>
<th>ATTenuation?</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6-41</td>
<td></td>
<td>Returns current value of Input Attenuation.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------------------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DUPlex:</td>
<td>CHANnel n</td>
<td>6-42</td>
<td>Sets Duplex Transmitter Frequency cellular channel n.</td>
</tr>
<tr>
<td></td>
<td>CHANnel: BAND?</td>
<td></td>
<td>Returns band for Duplex Transmitter Channel Format. Returns one of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>following bands for the associated channel format: (NADC) U8, U4 or HY;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(ETACS) NOT AVAILABLE; (NAMPS) LOWER, MIDDLE or UPPER.</td>
</tr>
<tr>
<td>FORMat: AMPS:</td>
<td>FOReward</td>
<td>6-42</td>
<td>Selects AMPS (NADC-U8) Forward channels.</td>
</tr>
<tr>
<td></td>
<td>REVerse</td>
<td></td>
<td>Selects AMPS (NADC-U8) Reverse channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects ETACS Forward channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects ETACS Reverse channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NADC Forward. Utilizes current setting of NADC band.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NADC Reverse. Utilizes current setting of NADC band.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NAMPS Narrow Analog channel designator.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NAMPS Forward channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NAMPS Reverse channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NT400 Forward channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selects NT400 Reverse channels.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Returns Duplex Transmitter Channel Format</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(NADC: FORWARD, NADC:REVERSE, ETACS:FORWARD, ETACS:REV,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAMPS:FORWARD or NAMPS:REV).</td>
</tr>
<tr>
<td>FIND: FREQuency?</td>
<td></td>
<td>6-44</td>
<td>Returns frequency of first signal with amplitude larger than Find reference</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>level.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Sets Find reference level in dB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Returns Find reference level in dB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Sets Duplex Transmitter Frequency.</td>
</tr>
<tr>
<td>REFerence n</td>
<td></td>
<td>6-44</td>
<td>Displays Deviation Meter (RMS) when followed by a SCREEN:DUPLEX command.</td>
</tr>
<tr>
<td>REFerence?</td>
<td></td>
<td></td>
<td>Displays Distortion Meter when followed by a SCREEN:DUPLEX command.</td>
</tr>
<tr>
<td>FREQuency n [units]</td>
<td>2500000.0 to 201000000.0 [Hz] 250.0 to 2010000.0 [kHz] 0.250 to 2010.0 [MHz]</td>
<td>6-44</td>
<td>Displays Modulation Meter when followed by a SCREEN:DUPLEX command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Displays Phase Meter (RMS) when followed by a SCREEN:DUPLEX command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Displays SINAD Meter when followed by a SCREEN:DUPLEX command.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MODE type</td>
<td>DiRect, CHAnnel. SCAN, LIST. FLScan</td>
<td>6-45</td>
<td>Selects Duplex Transmitter Mode.</td>
</tr>
<tr>
<td>MODE?</td>
<td></td>
<td>6-45</td>
<td>Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or FREQUENCY LIST SCAN).</td>
</tr>
<tr>
<td>MODulation?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMn</td>
<td>1 or 2</td>
<td>6-45</td>
<td>Selects Amplitude Modulation.</td>
</tr>
<tr>
<td>BFO</td>
<td></td>
<td>6-45</td>
<td>Selects Beat Frequency Oscillation.</td>
</tr>
<tr>
<td>FMn</td>
<td>1, 2, 3 or 4</td>
<td>6-45</td>
<td>Selects Frequency Modulation.</td>
</tr>
<tr>
<td>PM</td>
<td></td>
<td>6-45</td>
<td>Selects Phase Modulation.</td>
</tr>
<tr>
<td>USER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILTer f</td>
<td>3, 30 or 300</td>
<td>6-46</td>
<td>Sets User Defined Modulation IF Filter in kHz.</td>
</tr>
<tr>
<td>MODulation:otype</td>
<td>FM, AM, BFO, PM or DATA (FM DATA)</td>
<td></td>
<td>Selects User Defined Modulation.</td>
</tr>
<tr>
<td>POST</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>APASs</td>
<td></td>
<td>6-46</td>
<td>Selects All Pass Post Detection Filter for User Defined Modulation.</td>
</tr>
<tr>
<td>BPASs tl,th</td>
<td>0.5 to 20.0, 0.1 to 30.0</td>
<td>6-46</td>
<td>Selects Bandpass Post Detection Filter for User Defined Modulation, and specifies lower cutoff frequency (tl) and higher cutoff frequency (th) in kHz.</td>
</tr>
<tr>
<td>CWEEight</td>
<td></td>
<td>6-46</td>
<td>Selects C-Weighted Post Detection Filter for User Defined Modulation.</td>
</tr>
<tr>
<td>HPASs tl</td>
<td>0.5 to 20.0</td>
<td>6-46</td>
<td>Selects High-Pass Post Detection Filter for User Defined Modulation, and specifies cutoff frequency in kHz.</td>
</tr>
<tr>
<td>LPASs th</td>
<td>0.1 to 30.0</td>
<td>6-46</td>
<td>Selects Low-Pass Post Detection Filter for User Defined Modulation, and specifies cutoff frequency in kHz.</td>
</tr>
<tr>
<td>MODulation?</td>
<td></td>
<td>6-47</td>
<td>Returns the current Duplex Transmitter Modulation (FM, AM, BFO, PM, DATA or USER).</td>
</tr>
<tr>
<td>SCAN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ABORT</td>
<td></td>
<td>6-47</td>
<td>Stops frequency scan or frequency list scan depending on present mode.</td>
</tr>
<tr>
<td>CONTInue</td>
<td></td>
<td>6-47</td>
<td>Starts frequency scan or frequency list scan depending on present mode.</td>
</tr>
<tr>
<td>FREQList:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAUSEx n</td>
<td>0.0 to 99.9</td>
<td>6-48</td>
<td>Frequency List Scan Pause Time. Specifies time period in seconds for Duplex Transmitter to sit on a frequency if squelch is broken.</td>
</tr>
<tr>
<td>PAUSEx?</td>
<td></td>
<td>6-48</td>
<td>Returns current value of Frequency List Scan Pause Time.</td>
</tr>
<tr>
<td>RATEx n</td>
<td>0.02 to 99.99</td>
<td>6-48</td>
<td>Frequency List Scan Rate. Specifies time period in seconds for Duplex Transmitter to sit on a frequency unless squelch is broken.</td>
</tr>
<tr>
<td>RATEx?</td>
<td></td>
<td>6-48</td>
<td>Returns current value of Frequency List Scan Rate.</td>
</tr>
<tr>
<td>SQUElch b</td>
<td>1 or 0</td>
<td>6-49</td>
<td>Enables/disables Squelch.</td>
</tr>
<tr>
<td>SQUElch?</td>
<td></td>
<td>6-49</td>
<td>Returns current state of Squelch.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-47</td>
<td>Returns current frequency in kHz being scanned.</td>
</tr>
<tr>
<td>INCrement n</td>
<td>0.0 to 2010250.0</td>
<td>6-47</td>
<td>(Frequency Scan mode only) Specifies increment in kHz between frequencies to be scanned.</td>
</tr>
<tr>
<td>PAUsen</td>
<td>0.0 to 99.9</td>
<td>6-47</td>
<td>(Frequency Scan mode only) Pause Time. Specifies time period in seconds for Duplex Transmitter to sit on a frequency if squelch is broken.</td>
</tr>
<tr>
<td>PAUsen?</td>
<td></td>
<td>6-47</td>
<td>(Frequency Scan mode only) Returns a 1 if scanning is paused (stopped) or 0 if currently scanning.</td>
</tr>
<tr>
<td>RATE x n</td>
<td>0.02 to 99.99</td>
<td>6-48</td>
<td>(Frequency Scan mode only) Scan Rate. Specifies time period in seconds for Duplex Transmitter to sit on a frequency unless squelch is broken.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>DUPlnex:INPut:SCAN:STARt</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCAN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>START f</td>
<td>250.0 to 2010000.0</td>
<td>6-48</td>
<td>(Frequency Scan mode only). Starting Frequency value in kHz. Specifies lower limit frequency for scanning.</td>
</tr>
<tr>
<td>STOP f</td>
<td>250.0 to 2010000.0</td>
<td>6-48</td>
<td>(Frequency Scan mode only). Stop Frequency value in kHz. Specifies upper limit frequency for scanning.</td>
</tr>
<tr>
<td>TO:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUDIO b</td>
<td>1 or 0</td>
<td>6-49</td>
<td>Connects/disconnects demodulated Receiver Input to AUDIO OUT Connector.</td>
</tr>
<tr>
<td>DEMOD b</td>
<td>1 or 0</td>
<td>6-49</td>
<td>Connects/disconnects demodulated Receiver Input to DEMOD OUT Connector.</td>
</tr>
<tr>
<td>SPEAKER b</td>
<td>1 or 0</td>
<td>6-49</td>
<td>Connects/disconnects demodulated Receiver Input to Test Set Speaker.</td>
</tr>
<tr>
<td>TR</td>
<td></td>
<td></td>
<td>6-49</td>
</tr>
<tr>
<td>VOLUME:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO b</td>
<td>1 or 0</td>
<td>6-49</td>
<td>Enables/disables Automatic Volume Control.</td>
</tr>
<tr>
<td>AUTO?</td>
<td></td>
<td></td>
<td>6-49</td>
</tr>
<tr>
<td>METER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DISTortion</td>
<td></td>
<td></td>
<td>6-40</td>
</tr>
<tr>
<td>MODMeter</td>
<td></td>
<td></td>
<td>6-40</td>
</tr>
<tr>
<td>SINAD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTPUT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUDIO b</td>
<td>1 or 0</td>
<td>6-50</td>
<td>Connects/disconnects AF Generator Output to AUDIO OUT Connector.</td>
</tr>
<tr>
<td>CHANnel n</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHANnel:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BAND?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FORMat:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FORward</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ETACS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FORward</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NADC:</td>
<td>BAND:xx</td>
<td>U8, U4 or HYper</td>
<td>6-50</td>
</tr>
<tr>
<td>FORward</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMPS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BAND:x</td>
<td>Lower, Middle or Upper</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NT400:</td>
<td>FORward</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------</td>
<td>------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DUPlex: OUTput: DEMOD b</td>
<td>1 or 0</td>
<td>6-51</td>
<td>Connects/disconnects AF Generator Output to DEMOD OUT Connector.</td>
</tr>
<tr>
<td>DUPlex</td>
<td>1 or 0</td>
<td>6-52</td>
<td>Connects/disconnects RF Generator Output to DUPLEX OUT Connector.</td>
</tr>
<tr>
<td>FREQUENCY n [units]</td>
<td>250000.0 to 20100000000.0 [Hz]</td>
<td>6-52</td>
<td>Sets Duplex Receiver Frequency.</td>
</tr>
<tr>
<td></td>
<td>250.0 to 2010000.0 [kHz]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2500 to 2010.0 [MHz]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

units - Hz, kHz or MHz (KHZ default)

<p>| FREQUENCY?          | 6-52 | Returns Duplex Receiver Frequency in kHz.                                   |
| LEVEL:              |      |                                                                              |
| DBm n               | -137.0 to 0.0 | 6-52 | Sets RF Level in dBm.                                                        |
| DBm?                |       | 6-52 | Returns RF Level in dBm.                                                      |
| METER:              |      |                                                                              |
| AF                  | 6-52 | Displays AF Meter on Duplex Receiver Operation Screen.                      |
| DISTortion          | 6-52 | Displays Distortion Meter on Duplex Receiver Operation Screen.              |
| DMM                 | 6-52 | Displays Digital Multimeter on Duplex Receiver Operation Screen.            |
| SINAD MODE type     | 6-52 | Displays SINAD Meter on Duplex Receiver Operation Screen.                   |
|                     |      | Selects the Duplex Receiver Mode.                                           |
| MODE?               | 6-53 | Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or FREQUENCY LIST SCAN). |
| OFFSET f            | -999749.9 to 999749.9 | 6-53 | Sets Offset Frequency in kHz.                                                |
| OFFSET? SCAN:       |      |                                                                              |
| ABORT               | 6-53 | Returns Offset Frequency in kHz.                                            |
| CONTINUE            | 6-53 | Starts frequency scan or frequency list scan (depending on present mode).  |
| FREQList: RATE n    | 0.02 to 99.99 | 6-54 | Frequency List Scan Rate. Specifies time period in seconds for Duplex Receiver to generate current frequency prior to hopping to next frequency. |
| FREQUENCY? INCrement n | 0.0 to 2010000.0 | 6-53 | (Frequency Scan mode only) Specifies increment in kHz between frequencies to be generated. |
| PAUSE?              |      | 6-54 | (Frequency Scan mode only) Returns a 0 if scanning is paused (stopped) or 0 if currently scanning. |
| RATE n              | 0.02 to 99.99 | 6-54 | (Frequency Scan mode only) Scan Rate. Specifies time period in seconds for each generated frequency. |
| START f             | 250.0 to 2010000.0 | 6-54 | (Frequency Scan mode only) Start Frequency. Specifies lower limit frequency in kHz for first generated frequency. |
| STOP f              | 250.0 to 2010000.0 | 6-54 | (Frequency Scan mode only) Stop Frequency. Specifies upper limit frequency in kHz for last generated frequency. |
| TR                  | 6-54 | Routes Duplex Receiver Output to TR Connector and disconnects DUPLEX OUT Connector. |
| RCL n               | 1 to 9 | 6-40 | Recalls Duplex environment ( routings and settings) stored in memory location n. |</p>
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUPlex:</td>
<td></td>
<td>6-40</td>
<td>Selects Test Set Speaker Input. Command takes effect when RF Generator Operation Screen is updated.</td>
</tr>
<tr>
<td>SPEAKer:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SOURce type</td>
<td>OFF, FGEN (Function Generator), SINAD (SINAD/BER IN Connector) or EXTMOD (EXT MOD IN Connector)</td>
<td>6-40</td>
<td>Stores current Duplex environment ( routings and settings ) in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td></td>
<td>6-40</td>
<td></td>
</tr>
</tbody>
</table>

**FETCH COMMANDS**

<table>
<thead>
<tr>
<th>FETCH</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FETCH:</td>
<td></td>
<td>6-172</td>
<td>Returns Audio Frequency Meter reading in kHz. Must be used with INITiate command.</td>
</tr>
<tr>
<td>AF?</td>
<td></td>
<td>6-172</td>
<td>Returns Digital Multimeter reading. For ACV and DCV Multimeter Function, reading is returned in V. For ACC and DCC Multimeter Function, reading is returned in A. For Ohm Multimeter Function, reading is returned in Ω. Must be used with INITiate command.</td>
</tr>
<tr>
<td>DMM?</td>
<td></td>
<td>6-172</td>
<td>Returns Frequency Error Meter reading in kHz. Must be used with INITiate command.</td>
</tr>
<tr>
<td>RF?</td>
<td></td>
<td>6-172</td>
<td></td>
</tr>
</tbody>
</table>

**FLASH MEMORY COMMANDS**

See MMEMory;

**FREQUENCY ERROR METER COMMANDS**

See M_RF;

**FUNCTION GENERATOR COMMANDS**

<table>
<thead>
<tr>
<th>FGEN:</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOOST b</td>
<td>1 or 0</td>
<td>6-56</td>
<td>Enables/disables Generator #1 Boost.</td>
</tr>
<tr>
<td>BOOST?</td>
<td></td>
<td>6-56</td>
<td>Returns current state of Generator #1 Boost.</td>
</tr>
<tr>
<td>DATA:</td>
<td></td>
<td>6-56</td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0 to 4095</td>
<td>6-56</td>
<td>Specifies audio level for BER Data Generator.</td>
</tr>
<tr>
<td>LEVel?</td>
<td></td>
<td>6-56</td>
<td>Returns current audio level for BER Data Generator.</td>
</tr>
<tr>
<td>MODL n</td>
<td>0.0 to 10.0 (%)*</td>
<td>6-56</td>
<td>Sets Data Generator to n Modulation level.</td>
</tr>
<tr>
<td>MODL?</td>
<td></td>
<td>6-56</td>
<td></td>
</tr>
<tr>
<td>MODulation:</td>
<td>OFF, AM, FM, or PM</td>
<td>6-56</td>
<td>Returns Data Generator Modulation.</td>
</tr>
<tr>
<td>MODulation?</td>
<td>1 or 0</td>
<td>6-56</td>
<td>Enables/disables Data Modulation.</td>
</tr>
<tr>
<td>STATE b</td>
<td></td>
<td>6-57</td>
<td></td>
</tr>
<tr>
<td>EXT:</td>
<td></td>
<td>6-57</td>
<td>Sets External Modulation Proportional output level to n%. Command has no effect if Proportional Mode is off.</td>
</tr>
<tr>
<td>LEVel n</td>
<td>0 to 100</td>
<td>6-57</td>
<td>Returns External Modulation Proportional output level setting in %.</td>
</tr>
<tr>
<td>LEVel?</td>
<td></td>
<td>6-57</td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>FGEN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODL n</td>
<td>AM: 0 to 100 (%)</td>
<td>6-57</td>
<td>Specifies External Modulation level.</td>
</tr>
<tr>
<td></td>
<td>FM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.0 (radians)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODL?</td>
<td></td>
<td>6-57</td>
<td>Returns External Modulation level setting.</td>
</tr>
<tr>
<td>MODulation:</td>
<td>OFF, AM, FM</td>
<td>6-57</td>
<td>Selects External Modulation.</td>
</tr>
<tr>
<td>MODulation?</td>
<td>or FM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-58</td>
<td>Returns External Modulation type (OFF, AM, FM or PM).</td>
</tr>
<tr>
<td>FSK x</td>
<td>1 selects</td>
<td>6-58</td>
<td>Enables/disables External Modulation.</td>
</tr>
<tr>
<td></td>
<td>Generator #1,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 selects</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Generator #2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEN1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQuency f</td>
<td>0.0 to 40000.0</td>
<td>6-59</td>
<td>Sets AF Generator 1 output frequency in Hz.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-59</td>
<td>Returns AF Generator 1 output frequency in Hz.</td>
</tr>
<tr>
<td>LEVel n</td>
<td>0 to 100</td>
<td>6-58</td>
<td>Sets Proportional output level of AF Generator 1 in %.</td>
</tr>
<tr>
<td>LEVel?</td>
<td></td>
<td>6-59</td>
<td>Returns Proportional output level setting of AF Generator 1 in %.</td>
</tr>
<tr>
<td>MODL n</td>
<td>AM: 0 to 100 (%)</td>
<td>6-59</td>
<td>Sets Modulation level of AF Generator 1.</td>
</tr>
<tr>
<td></td>
<td>FM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>100.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.0 (radians)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODL?</td>
<td></td>
<td>6-59</td>
<td>Returns Modulation level setting of AF Generator 1.</td>
</tr>
<tr>
<td>MODulation?</td>
<td>or PM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHAPE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHAPE:</td>
<td></td>
<td>6-60</td>
<td>Sets wave shape to DC level n for AF Generator 1.</td>
</tr>
<tr>
<td>SHAPE:</td>
<td>-1, 0 or 1</td>
<td>6-60</td>
<td>Sets wave shape of AF Generator 1 to Pulse with 50% duty cycle.</td>
</tr>
<tr>
<td>SHAPE: type</td>
<td>SIN (Sine),</td>
<td>6-59</td>
<td>Selects wave shape for AF Generator 1.</td>
</tr>
<tr>
<td></td>
<td>SOU (Square Wave),</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAMP or</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRI (Triangle)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SHAPE?</td>
<td></td>
<td>6-59</td>
<td>Returns wave shape of AF Generator 1.</td>
</tr>
<tr>
<td>STATe b</td>
<td></td>
<td>6-60</td>
<td>Enables/disables AF Generator 1.</td>
</tr>
<tr>
<td>GEN2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQuency f</td>
<td>0.0 to 40000.0</td>
<td>6-59</td>
<td>Sets AF Generator 2 output frequency in Hz.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-59</td>
<td>Returns AF Generator 2 output frequency in Hz.</td>
</tr>
<tr>
<td>LEVel n</td>
<td>0 to 100</td>
<td>6-59</td>
<td>Sets Proportional output level of AF Generator 2 in %.</td>
</tr>
<tr>
<td>LEVel?</td>
<td></td>
<td>6-59</td>
<td>Returns Proportional output level setting of AF Generator 2 in %.</td>
</tr>
<tr>
<td>MODL n</td>
<td>AM: 0 to 100 (%)</td>
<td>6-59</td>
<td>Sets Modulation level of AF Generator 2.</td>
</tr>
<tr>
<td></td>
<td>FM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>100.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.0 (radians)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>FGEN:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>GEN2:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MODL?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODulation::type</td>
<td>OFF, AM, FM or PM</td>
<td>6-59</td>
<td>Returns Modulation level setting of AF Generator 2. Selects Modulation for AF Generator 2.</td>
</tr>
<tr>
<td><strong>SHAPE:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DC n</td>
<td>-1, 0 or 1</td>
<td>6-60</td>
<td>Sets wave shape to DC level n for AF Generator 2.</td>
</tr>
<tr>
<td>PULSE:</td>
<td>DCYCLE 50</td>
<td>6-60</td>
<td>Sets wave shape of AF Generator 2 to Pulse with 50% duty cycle.</td>
</tr>
<tr>
<td>SHAPE::type</td>
<td>SIN (Sine), SQU (Square Wave), RAMP or TRI (Triangle)</td>
<td>6-59</td>
<td>Selects wave shape for AF Generator 2.</td>
</tr>
<tr>
<td><strong>SHAPE?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-60</td>
<td>Returns wave shape of AF Generator 2. Enables/disables AF Generator 2.</td>
</tr>
<tr>
<td><strong>GEN3:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>DIGItal type</strong></td>
<td>DCS, DCSINV, PCSAG, DSAT or DST</td>
<td>6-61</td>
<td>Sets digital type.</td>
</tr>
<tr>
<td><strong>ENCode type</strong></td>
<td>DTMF, TONE, DIGItal or RCC</td>
<td>6-61</td>
<td>Selects signaling format to encode. Must be followed with a SETUP:GEN or SCREEN:GEN command.</td>
</tr>
<tr>
<td><strong>MODL n</strong></td>
<td>AM: 0 to 100.0 (%)</td>
<td>6-61</td>
<td>Sets AF Generator 3 Modulation level.</td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to 10.0 (radians)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FM (Tone/RCC): 0.0 to 100.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FM (DTMF): 0.0 to 10.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FM (Digital): 0.0 to 25.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MODL?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODulation::type</td>
<td>OFF, AM, FM or PM</td>
<td>6-61</td>
<td>Returns AF Generator 3 Modulation level setting. Selects AF Generator 3 Modulation.</td>
</tr>
<tr>
<td><strong>RCC format</strong></td>
<td>IMTS, MTS, SYS805 or TREMote</td>
<td>6-61</td>
<td>Selects Generator RCC signalling format.</td>
</tr>
<tr>
<td><strong>MIC:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVEL n</td>
<td>0 to 100</td>
<td>6-61</td>
<td>Sets Proportional output level of MIC/ACC Connector input in %. Command has no effect if Proportional Mode off.</td>
</tr>
<tr>
<td>LEVEL?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODL n</td>
<td>AM: 0 to 100.0 (%)</td>
<td>6-62</td>
<td>Sets MIC/ACC Connector input Modulation level.</td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to 25.0 (kHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PM: 0.0 to 10.0 (radians)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MODL?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODulation::type</td>
<td>OFF, AM, FM or PM</td>
<td>6-62</td>
<td>Returns MIC/ACC Connector input Modulation level setting. Selects MIC/ACC Connector input Modulation type.</td>
</tr>
<tr>
<td><strong>STATe b</strong></td>
<td>1 or 0</td>
<td>6-62</td>
<td>Enables/disables external modulation through MIC/ACC Connector.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>FGEN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTput</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUDio b</td>
<td>1 or 0</td>
<td>6-63</td>
<td>Connects/disconnects AF Generator Output to AUDIO OUT Connector.</td>
</tr>
<tr>
<td>AUDio?</td>
<td></td>
<td>6-63</td>
<td>Returns 1 if AF Generator Output is routed to the AUDIO OUT Connector; 0 if AUDIO OUT Connector is disconnected.</td>
</tr>
<tr>
<td>DEMOD b</td>
<td>1 or 0</td>
<td>6-63</td>
<td>Connects/disconnects AF Generator Output to DEMOD OUT Connector.</td>
</tr>
<tr>
<td>DEMOD?</td>
<td></td>
<td>6-63</td>
<td>Returns 1 if AF Generator Output is routed to DEMOD OUT Connector; 0 if DEMOD OUT Connector is disconnected.</td>
</tr>
<tr>
<td>LEVel v</td>
<td>0.0000 to 3.1000</td>
<td>6-63</td>
<td>Sets AF Generator output level in volts.</td>
</tr>
<tr>
<td>LEVel?</td>
<td></td>
<td>6-63</td>
<td>Returns AF Generator output level in volts.</td>
</tr>
<tr>
<td>SPEAKER b</td>
<td>1 or 0</td>
<td>6-63</td>
<td>Connects/disconnects AF Generator to Speaker.</td>
</tr>
<tr>
<td>SPEAKER?</td>
<td></td>
<td>6-63</td>
<td>Returns 1 if AF Generator Output is routed to Speaker; 0 if Speaker is disconnected.</td>
</tr>
<tr>
<td>PROP ortional b</td>
<td>1 or 0</td>
<td>6-64</td>
<td>Enables/disables AF Generator Proportional mode.</td>
</tr>
<tr>
<td>PROP ortional?</td>
<td></td>
<td>6-64</td>
<td>Returns 1 if AF Generator is in Proportional mode; 0 otherwise.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-64</td>
<td>Recalls AF Generator environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-64</td>
<td>Stores current AF Generator environment (routings and settings) in memory location n.</td>
</tr>
<tr>
<td>PTT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STAte b</td>
<td>1 = high (mic keyed), 0 = low</td>
<td>6-62</td>
<td>Sets output on the push to talk pin on the MIC/ACC Connector.</td>
</tr>
</tbody>
</table>

**GENERATOR COMMANDS**

**GENerator:**

**AF**

**CHANnel n**

1 to 2047, depending on format

**CHANnel:**

**BAND?**

**FORMat:**

**AMPS:**

FORward

REVerse

**ETACS:**

FORward

REVerse

**NADC:**

BAND:xx

U8, U4 or HYper

FORward

REVerse

**NAMPS:**

BAND:x

Lower, Middle or Upper

**FOReward**

REVerse

**NT400:**

FORward

REVerse

**FORMat?**

6-7 | Displays AF Level Meter on RF Generator Operation Screen when followed by SCREEN:GENerator command. Sets cellular channel in format selected using the GEN:CHAN:FORM command. |

6-7 | Returns band for Generator Channel Format (NADC - U8, U4 or HY; ETACS - NOT AVAILABLE; NAMPS - LOWER, MIDDLE or UPPER). |
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENerator: DCS: INVert</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCS: INVert mnn</td>
<td>000 to 777</td>
<td>6-8</td>
<td>Generates three digit octal Digital Coded Squelch (DCS) code in inverted mode.</td>
</tr>
<tr>
<td>DCS: NORMAL mnn</td>
<td>000 to 777</td>
<td>6-8</td>
<td>Generates three digit octal DCS code.</td>
</tr>
<tr>
<td>DCS: STOP</td>
<td></td>
<td>6-8</td>
<td>Stops generating continuous DCS code.</td>
</tr>
<tr>
<td>DCS: DIAL &quot;sequence&quot;</td>
<td>0 to 9</td>
<td>6-8</td>
<td>Generates sequence as 2805 Pulse code.</td>
</tr>
<tr>
<td></td>
<td>(16 digits max.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCS: DIAL: FREQuency f</td>
<td>0.0 to 40000.0</td>
<td>6-8</td>
<td>Sets 2805 Tone frequency in Hz.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-8</td>
<td>Returns the current 2805 Tone frequency in Hz.</td>
</tr>
<tr>
<td>DCS: DISTortion</td>
<td></td>
<td>6-8</td>
<td>Displays Distortion Meter on RF Generator Operation Screen.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-8</td>
<td>After screen is updated by SCREEN: GENERator command.</td>
</tr>
<tr>
<td>DCS: DMM</td>
<td></td>
<td>6-8</td>
<td>Displays Digital Multimeter on RF Generator Operation Screen.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-8</td>
<td>After screen is updated by SCREEN: GENERator command.</td>
</tr>
<tr>
<td>DCS: DSAT n</td>
<td>0 to 6</td>
<td>6-9</td>
<td>Generates Digital Supervisory Audio Tone (DSAT) code.</td>
</tr>
<tr>
<td>DCS: DSAT: STOP</td>
<td></td>
<td>6-9</td>
<td>Stops generating continuous DSAT code.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-9</td>
<td>Generates Digital Signal Tone (DST) code.</td>
</tr>
<tr>
<td>DCS: DST n</td>
<td>0 to 6</td>
<td>6-9</td>
<td>Generates Digital Signal Tone (DST) code.</td>
</tr>
<tr>
<td>DCS: DST: STOP</td>
<td></td>
<td>6-9</td>
<td>Stops generating continuous DST code.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-9</td>
<td>Generates DTMF coded sequence.</td>
</tr>
<tr>
<td></td>
<td>(16 digits max.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>25 to 9999 ms for</td>
<td>6-9</td>
<td>Sets RF Generator Frequency.</td>
</tr>
<tr>
<td>DCS: FREQuency n [units]</td>
<td>2500000.0 to</td>
<td>6-9</td>
<td>mark or space. (Default for mark</td>
</tr>
<tr>
<td></td>
<td>20100000.0 [Hz]</td>
<td></td>
<td>time is 74 ms. (Default for space</td>
</tr>
<tr>
<td></td>
<td>250.0 to</td>
<td>6-9</td>
<td>time is 67 ms.)</td>
</tr>
<tr>
<td></td>
<td>201000.0 [kHz]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2500 to</td>
<td>6-9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2010.0 [MHz]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>units - Hz, KHZ or MHz (KHZ, default)</td>
<td>6-9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQuency?</td>
<td>6-9</td>
<td>Returns RF Generator Frequency in kHz (250.0 to 2010000.0).</td>
</tr>
<tr>
<td>DCS: IMTS &quot;sequence&quot;</td>
<td>0 to 9</td>
<td>6-9</td>
<td>Generates DCS IMTS code.</td>
</tr>
<tr>
<td></td>
<td>(16 digits max.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCS: LEVeI DBm n</td>
<td>-137.0 to 0.0</td>
<td>6-10</td>
<td>Sets RF Generator Level in dBm.</td>
</tr>
<tr>
<td>DCS: LEVeI DBm?</td>
<td></td>
<td>6-10</td>
<td>Returns RF Generator Level in dBm.</td>
</tr>
</tbody>
</table>
### COMMAND

<table>
<thead>
<tr>
<th>GENERATOR: LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GEN</strong>erator: LEVEL n [units]**</td>
</tr>
<tr>
<td><strong>RANGE</strong></td>
</tr>
<tr>
<td>-137.0 to 0.0 (dBm), 0.031 to 224000.0 (µV), 0.0000031 to 224.0 (mV) or 0.0000000031 to 0.224 (V)</td>
</tr>
<tr>
<td><strong>LEVEL:</strong></td>
</tr>
<tr>
<td><strong>UNIT</strong></td>
</tr>
<tr>
<td><strong>UNIT?</strong></td>
</tr>
<tr>
<td><strong>LEVEL?</strong></td>
</tr>
<tr>
<td><strong>MODE type</strong></td>
</tr>
<tr>
<td><strong>MODE?</strong></td>
</tr>
<tr>
<td><strong>MTS &quot;sequence&quot;</strong></td>
</tr>
<tr>
<td><strong>OUTPUT: AUDIO b</strong></td>
</tr>
<tr>
<td><strong>DEMOD b</strong></td>
</tr>
<tr>
<td><strong>POCSAG:</strong></td>
</tr>
<tr>
<td><strong>ALPHA:</strong></td>
</tr>
<tr>
<td><strong>LOWer cascode</strong></td>
</tr>
<tr>
<td><strong>NUMeric cascode</strong></td>
</tr>
<tr>
<td><strong>SPECial cascode</strong></td>
</tr>
<tr>
<td><strong>UPPer cascode</strong></td>
</tr>
<tr>
<td><strong>BEEP n, cascode</strong></td>
</tr>
<tr>
<td><strong>NUMeric cascode</strong></td>
</tr>
<tr>
<td><strong>RATE</strong></td>
</tr>
<tr>
<td><strong>RATE?</strong></td>
</tr>
<tr>
<td><strong>RCI n</strong></td>
</tr>
<tr>
<td><strong>SCAN:</strong></td>
</tr>
<tr>
<td><strong>ABORT</strong></td>
</tr>
<tr>
<td><strong>CONTINUE</strong></td>
</tr>
<tr>
<td><strong>FREQList:</strong></td>
</tr>
<tr>
<td><strong>RATe n</strong></td>
</tr>
<tr>
<td><strong>RATE?</strong></td>
</tr>
<tr>
<td><strong>FREQuency?</strong></td>
</tr>
<tr>
<td><strong>INCrement n</strong></td>
</tr>
<tr>
<td>COMMAND</td>
</tr>
<tr>
<td>--------------------</td>
</tr>
<tr>
<td>GENerator:SCAN:PAUse?</td>
</tr>
<tr>
<td>RATE n</td>
</tr>
<tr>
<td>START f</td>
</tr>
<tr>
<td>STOP f</td>
</tr>
<tr>
<td>SINAD</td>
</tr>
<tr>
<td>SPEAKER:</td>
</tr>
<tr>
<td>SOURCE type</td>
</tr>
<tr>
<td>STORE n</td>
</tr>
<tr>
<td>TONE &quot;sequence&quot;</td>
</tr>
<tr>
<td>TONE:</td>
</tr>
<tr>
<td>TYPE code</td>
</tr>
<tr>
<td>USER:</td>
</tr>
<tr>
<td>DEFINE &quot;id&quot;, freq, duration</td>
</tr>
<tr>
<td>TREMote</td>
</tr>
<tr>
<td>TREMote:</td>
</tr>
</tbody>
</table>

**GENERIC MEASURE COMMANDS**

See Measure (Generic) Commands.
## INITIATE COMMANDS

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INITiate: AF</td>
<td></td>
<td>6-172</td>
<td>Prepares Audio Frequency Meter for a FETCH: command to take reading.</td>
</tr>
<tr>
<td>DMM</td>
<td></td>
<td>6-172</td>
<td>Prepares Audio Frequency Meter for a FETCH: command to take reading.</td>
</tr>
<tr>
<td>RF</td>
<td></td>
<td>6-172</td>
<td>Prepares Frequency Error Meter for a FETCH: command to take a reading.</td>
</tr>
</tbody>
</table>

## INSTRUMENT COMMAND

<table>
<thead>
<tr>
<th>INSTRument:</th>
<th>SELect type</th>
<th>6-5</th>
<th>Selects default instrument.</th>
</tr>
</thead>
</table>

## AF METER COMMANDS

<table>
<thead>
<tr>
<th>M_AF</th>
<th>ALARM b</th>
<th>6-92</th>
<th>Enables/disables Alarm.</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILTER:</td>
<td>APASSs</td>
<td>6-92</td>
<td>Selects All-Pass Filter for AF Meter.</td>
</tr>
<tr>
<td></td>
<td>APASSs?</td>
<td>6-92</td>
<td>Returns 1 if All-Pass Filter is enabled, 0 if disabled.</td>
</tr>
<tr>
<td></td>
<td>HPASSs:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQUENCY f</td>
<td>6-92</td>
<td>Sets High-Pass cutoff frequency in kHz.</td>
</tr>
<tr>
<td></td>
<td>FREQUENCY?</td>
<td>6-92</td>
<td>Returns High-Pass cutoff frequency in kHz.</td>
</tr>
<tr>
<td></td>
<td>STATE b</td>
<td>6-92</td>
<td>Enables/disables High-Pass Filter</td>
</tr>
<tr>
<td></td>
<td>STATE?</td>
<td>6-92</td>
<td>Returns 1 if High-Pass Filter is enabled, 0 if disabled.</td>
</tr>
<tr>
<td></td>
<td>LPASSs:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FREQUENCY f</td>
<td>6-92</td>
<td>Sets Low-Pass cutoff frequency in kHz.</td>
</tr>
<tr>
<td></td>
<td>FREQUENCY?</td>
<td>6-92</td>
<td>Returns Low-Pass cutoff frequency in kHz.</td>
</tr>
<tr>
<td></td>
<td>STATE b</td>
<td>6-93</td>
<td>Enables/disables Low-Pass Filter.</td>
</tr>
<tr>
<td></td>
<td>STATE?</td>
<td>6-93</td>
<td>Returns 1 if Low-Pass Filter is enabled, 0 if disabled.</td>
</tr>
<tr>
<td>INPUT:type</td>
<td>XAUDIO (Ext Mod), DEMOD (Demod Audio), FGEN (Func Gen Out), SINAD (SINAD/BER), POWer (RF Power)</td>
<td>6-93</td>
<td>Selects Audio Frequency Meter Input.</td>
</tr>
</tbody>
</table>

## LL:

<table>
<thead>
<tr>
<th>LEVel f</th>
<th>6-93</th>
<th>Sets Lower Limit in kHz.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(for an upper range of 0.2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(for upper ranges of 2, 20 and 200)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## STATe b

<table>
<thead>
<tr>
<th>STATE b</th>
<th>6-93</th>
<th>Enables/disables Audio Frequency Meter Lower Limit.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PEAK?</td>
<td>6-93</td>
<td>Returns Audio Frequency Meter Peak reading in Hz</td>
</tr>
<tr>
<td>(0.0 to 200000.0).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## PH b

<table>
<thead>
<tr>
<th>PH b</th>
<th>6-93</th>
<th>Enables/disables Peak Hold Feature.</th>
</tr>
</thead>
</table>
### M_AF: RANGE: AUTO

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>M_AF: RANGe: AUTO</td>
<td></td>
<td>6-94</td>
<td>Sets frequency range to Autorange.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-94</td>
<td>Sets frequency range in kHz.</td>
</tr>
<tr>
<td>RESolution n</td>
<td>0.1 (Gate Time of 10 s) or 1 (Gate Time of 1 s)</td>
<td>6-94</td>
<td>Sets Audio Frequency Meter Resolution (Gate Time) in Hz.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-94</td>
<td>Stores current Audio Frequency Meter environment (routines and settings) in memory location n.</td>
</tr>
<tr>
<td>ULEV n</td>
<td>0.0000 to 0.2000 (for an upper range of 0.2) or 0.000 to 200.000 (for upper ranges of 2, 20 and 200)</td>
<td>6-94</td>
<td>Sets Upper Limit in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-94</td>
<td>Enables/disables Audio Frequency Meter Upper Limit.</td>
</tr>
</tbody>
</table>

### BER (BIT ERROR RATE) METER COMMANDS

**M_BER:**

**PAtttern:**
- FIXED 6-113 Selects Fixed pattern for the BER Meter test data.
- FIXEDUSER? 6-113 Returns specified USER data pattern.
- RANDOM 6-113 Selects Random pattern for BER Meter test data.
- USER nn 6-113 Selects USER data pattern (nn could be a variable or expression) for the BER Meter test data. Patterns entered that are not base 10 are preceded with # character and letter signifying the number base: H (Hexadecimal), B (Binary) or Q (Octal). Pattern is displayed in Hexadecimal.

**POLarity:**
- NEGative 6-113 Selects Negative Polarity for BER Meter.
- POSitive 6-113 Selects Positive Polarity for BER Meter.

**RATEn n** 6-113 Evaluates BER Meter rate.

## RATE?
- 6-113 Returns BER Meter Rate setting.
- RCL n 6-114 Recalls BER Meter environment (routines and settings) stored in memory location n.
- SIZE n 6-114 Sets BER Meter block size in bits.

## SIZE?
- 6-114 Returns BER Meter block size setting.

## STORE n
- 1 to 9 6-114 Stores current BER Meter environment (routines and settings) in memory location n.

**TYPE:xxx**
- GENERator, RECEiver, DUPlex or BASEband 6-114 Selects Bit Error Rate Type.

**M_BER?**
- 6-114 Returns number of errors for last pass.
### Deviation Meter (Peak) Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Range</th>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M_DEV:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-101</td>
<td>Enables/disables Deviation Meter Alarm.</td>
</tr>
<tr>
<td>AVERAGE b</td>
<td>1 or 0</td>
<td>6-101</td>
<td>Enables/disables Deviation Meter Averaging.</td>
</tr>
<tr>
<td>LL: LEVEL f</td>
<td>0.00 to 20.00 (for range values of 2, 5, 10 or 20 kHz) 0 to 100 (for all other ranges)</td>
<td>6-101</td>
<td>Sets Deviation Meter Lower Limit in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-101</td>
<td>Enables/disables Deviation Meter Lower Limit.</td>
</tr>
<tr>
<td>MODE: BOTH</td>
<td></td>
<td></td>
<td>Selects Both Mode for Deviation Meter, reading positive and negative deviation.</td>
</tr>
<tr>
<td>NEGative</td>
<td></td>
<td></td>
<td>Selects Negative Mode for Deviation Meter, reading negative deviation.</td>
</tr>
<tr>
<td>NORMALize</td>
<td></td>
<td></td>
<td>Selects Normalized Mode for Deviation Meter, reading (positive + negative)/2 deviation.</td>
</tr>
<tr>
<td>POSitive</td>
<td></td>
<td></td>
<td>Selects Positive Mode for Deviation Meter, reading positive deviation.</td>
</tr>
<tr>
<td>NEG?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEAK</td>
<td></td>
<td></td>
<td>Return negative Deviation Meter Peak reading as an absolute value in kHz (0.00 to 100.00).</td>
</tr>
<tr>
<td>NEG?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>POS?</td>
<td></td>
<td></td>
<td>Return positive Deviation Meter Peak reading as an absolute value in kHz (0.00 to 100.00).</td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-102</td>
<td>Enables/disables Deviation Meter Peak Hold Feature.</td>
</tr>
<tr>
<td>POS?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RANGE: AUTO</td>
<td></td>
<td></td>
<td>Sets Deviation Meter range to Autorange.</td>
</tr>
<tr>
<td>UPPER f</td>
<td>2, 5, 10, 20, 50 or 100</td>
<td>6-102</td>
<td>Sets Deviation Meter range in kHz.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-103</td>
<td>Recalls Deviation Meter environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-103</td>
<td>Stores current Deviation Meter environment (routings and settings) in memory location n.</td>
</tr>
<tr>
<td>UL: LEVEL f</td>
<td>0.00 to 20.00 (for range values of 2, 5, 10 or 20 kHz) 0 to 100 (for all other ranges)</td>
<td>6-103</td>
<td>Sets Deviation Meter Upper Limit in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-103</td>
<td>Enables/disables Deviation Meter Upper Limit.</td>
</tr>
</tbody>
</table>

### Distortion Meter Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Range</th>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M_DIST:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-106</td>
<td>Enables/disables Distortion Meter Alarm.</td>
</tr>
<tr>
<td>AVERAGE b</td>
<td>1 or 0</td>
<td>6-106</td>
<td>Enables/disables Average Feature.</td>
</tr>
<tr>
<td>FILTER f</td>
<td>600 to 1400</td>
<td>6-106</td>
<td>Sets Notch Filter frequency in Hz.</td>
</tr>
<tr>
<td>INPUT: type</td>
<td>DEMOD (Demod Audio), SINAD (SINAD/BER), XAUD (Ext Mod), or FGEN (Func Gen)</td>
<td>6-106</td>
<td>Sets Distortion Meter Input.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>M_DIST:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.0 to 20.0</td>
<td>6-106</td>
<td>Sets Lower Limit in %.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-106</td>
<td>Enables/disables Distortion Meter Lower Limit.</td>
</tr>
<tr>
<td>PEAK?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PH b</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-107</td>
<td>Recalls Distortion Meter environment (routings and settings) stored at memory location n.</td>
</tr>
<tr>
<td>SElec:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMEssage</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CWEight</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LPASs f</td>
<td>100 to 30000</td>
<td>6-107</td>
<td>Same as M_DIST:SElec:CWEight command.</td>
</tr>
<tr>
<td>STORe n</td>
<td>1 to 9</td>
<td>6-107</td>
<td>Selects the C-Weight Filter.</td>
</tr>
<tr>
<td>UL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.0 to 20.0</td>
<td>6-107</td>
<td>Sets Upper Limit to in %.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-107</td>
<td>Enables/disables Distortion Meter Upper Limit.</td>
</tr>
<tr>
<td>M_DIST?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIGITAL MULTIMETER COMMANDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M_DMM:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FUNCTION:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CURRent:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESistance</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOLTage:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FUNCTION?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INPut:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMPedance n</td>
<td>150, 600 or 1e6</td>
<td>6-115</td>
<td>Sets Input Impedance in ohms. Command ignored if not in AC Voltmeter Function.</td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td></td>
<td>6-116</td>
<td>Sets Multimeter Lower Limit.</td>
</tr>
</tbody>
</table>

**Ammeter** (dc or ac): 0.0000 to 0.19999 (A) (200 and 200 mA range), 0.000 to 19.99 (A) (2 A and 20 A range).

**Voltmeter** (dc or ac): 0.0000 to 0.19999 (V) (200 mV range), 0.00 to 1000.00 (V) (2, 20, 200, 2000 V range).

**Ohmmeter**: 0.0000 to 0.1999 (kΩ) (200 Ω range), 0.00 to 19990 (kΩ) (2 kΩ to 20 MΩ range).

| STATe b | 1 or 0 | 6-116| Enables/disables Multimeter Lower Limit. |
| PH b    | 1 or 0 | 6-116| Enables/disables Multimeter Peak Hold feature. |
### M_DMM: RANGE: AUTO

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>M_DMM:</td>
<td>RANGE: AUTO</td>
<td>6-116</td>
<td>Sets Multimeter range to Autorange.</td>
</tr>
<tr>
<td>UPPER n</td>
<td>Voltmeter</td>
<td>6-116</td>
<td>Sets Multimeter range.</td>
</tr>
<tr>
<td></td>
<td>(dc or ac): 0.2, 2, 20, 200 or 2000 (V)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ammeter (dc or ac): 0.02, 0.2 or 20 (A)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ohmmeter: 0.2, 2, 20, 200, 2000 or 20000 (kΩ)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-117</td>
<td>Recalls Digital Multimeter environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-117</td>
<td>Stores current Digital Multimeter environment (routings and settings) in memory location n.</td>
</tr>
<tr>
<td>UL: LEVEL n</td>
<td>Ammeter</td>
<td>6-117</td>
<td>Returns Multimeter reading in current Function units.</td>
</tr>
<tr>
<td></td>
<td>(dc or ac): 0.00000 to 19.999 (A) (20 and 200 mA range). 0.000 to 199.999 (A) (2 A and 20 A range).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Voltmeter (dc or ac): 0.0000 to 19.999 (V) (200 mV range). 0.00 to 1000.00 (V) (2, 20, 200, 2000 V range).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ohmmeter: 0.0000 to 0.1999 (kΩ) (200 Ω range). 0.000 to 19999 (kΩ) (2 kΩ to 20 MΩ range).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-117</td>
<td>Enables/disables Multimeter Upper Limit.</td>
</tr>
<tr>
<td>M_DMM?</td>
<td></td>
<td>6-117</td>
<td>Enables/disables Deviation Meter (RMS) Upper Limit.</td>
</tr>
</tbody>
</table>

### DEVIATION METER (RMS) COMMANDS

<table>
<thead>
<tr>
<th>M_DRMS:</th>
<th>ALARM b</th>
<th>1 or 0</th>
<th>6-120</th>
<th>Enables/disables Deviation Meter (RMS) Alarm.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVERAGE b</td>
<td>1 or 0</td>
<td>6-120</td>
<td>Enables/disables Deviation Meter (RMS) Averaging.</td>
<td></td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td>6-120</td>
<td>Sets Deviation (RMS) Lower Limit in kHz.</td>
<td></td>
</tr>
<tr>
<td>LEVEL f</td>
<td>0.00 to 10.00</td>
<td>6-120</td>
<td>Enables/disables Deviation Meter (RMS) Lower Limit.</td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-120</td>
<td>Enables/disables Deviation Meter (RMS) Peak Hold feature.</td>
<td></td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-120</td>
<td>Enables/disables Deviation Meter (RMS) Peak Hold feature.</td>
<td></td>
</tr>
<tr>
<td>RANGE: AUTO</td>
<td></td>
<td>6-120</td>
<td>Sets Deviation Meter (RMS) range to Autorange.</td>
<td></td>
</tr>
<tr>
<td>UPPER f</td>
<td>2, 5 or 10</td>
<td>6-120</td>
<td>Sets Deviation Meter (RMS) range in kHz.</td>
<td></td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-120</td>
<td>Recalls Deviation Meter (RMS) environment (routings and settings) stored in memory location n.</td>
<td></td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-120</td>
<td>Stores current Deviation Meter (RMS) environment (routings and settings) in memory location n.</td>
<td></td>
</tr>
<tr>
<td>UL:</td>
<td>LEVEL f</td>
<td>0.00 to 10.00</td>
<td>6-121</td>
<td>Sets Deviation (RMS) Upper Limit in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-121</td>
<td>Enables/disables Deviation Meter (RMS) Upper Limit.</td>
<td></td>
</tr>
<tr>
<td>M_DRMS?</td>
<td></td>
<td>6-121</td>
<td>Returns Deviation Meter (RMS) reading in kHz (0.00 to 10.30).</td>
<td></td>
</tr>
</tbody>
</table>
### MODULATION METER COMMANDS

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M.MOD:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-104</td>
<td>Enables/disables Modulation Meter Alarm.</td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| LEVel n | 0.0 to 100.0 | 6-104 | Sets Modulation Meter Lower Limit in %.
| STATe b | 1 or 0 | 6-104 | Enables/disables Modulation Meter Lower Limit. |
| PEAK? | | | |
| PH b | | | |
| RANGE: | | | |
| AUTO | | | |
| UPPer n | 40 or 100 | 6-104 | Sets Modulation Meter range to Autorange. |
| RCL n | 1 to 9 | 6-104 | Sets Modulation Meter range in %.
| STORe n | 1 to 9 | 6-105 | Stores current Modulation Meter environment (routings and settings) stored in memory location n. |
| UL: | | | |
| LEVel n | 0.0 to 100.0 | 6-105 | Sets Modulation Meter Upper Limit in %.
| STATe b | 1 or 0 | 6-105 | Enables/disables Modulation Meter Upper Limit. |
| **M.MOD?** | | | |

### PHASE METER COMMANDS

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M.PM:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-118</td>
<td>Enables/disables Phase Meter Alarm.</td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.00 to 10.00</td>
<td>6-118</td>
<td>Sets Phase Meter Lower Limit in radians.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-118</td>
<td>Enables/disables Phase Meter Lower Limit.</td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-118</td>
<td>Enables/disables Phase Meter Peak Hold feature.</td>
</tr>
<tr>
<td>RANGE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPPer n</td>
<td>1, 5 or 10</td>
<td>6-118</td>
<td>Sets Phase Meter range to Autorange.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-118</td>
<td>Sets Phase Meter range in radians.</td>
</tr>
<tr>
<td>STORe n</td>
<td>1 to 9</td>
<td>6-119</td>
<td>Stores current Phase Meter environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>UL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.00 to 10.00</td>
<td>6-119</td>
<td>Sets Phase Meter Upper Limit in radians.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-119</td>
<td>Enables/disables Phase Meter Upper Limit.</td>
</tr>
<tr>
<td><strong>M.PM?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### PHASE METER (RMS) COMMANDS

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M.PMRMS:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-122</td>
<td>Enables/disables Phase Meter (RMS) Alarm.</td>
</tr>
<tr>
<td>AVERage b</td>
<td>1 or 0</td>
<td>6-122</td>
<td>Enables/disables Averaging mode.</td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.00 to 10.00</td>
<td>6-122</td>
<td>Sets Phase Meter (RMS) Lower Limit in radians.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-122</td>
<td>Enables/disables Phase Meter (RMS) Lower Limit.</td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-122</td>
<td>Enables/disables Phase Meter (RMS) Peak Hold feature.</td>
</tr>
<tr>
<td>RANGE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPPer n</td>
<td>1, 5 or 10</td>
<td>6-122</td>
<td>Sets Phase Meter (RMS) range to Autorange.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-122</td>
<td>Sets Phase Meter (RMS) range in radians.</td>
</tr>
<tr>
<td>STORe n</td>
<td>1 to 9</td>
<td>6-122</td>
<td>Stores current Phase Meter (RMS) environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>UL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVel n</td>
<td>0.00 to 10.00</td>
<td>6-123</td>
<td>Sets Phase Meter (RMS) Upper Limit in radians.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-123</td>
<td>Enables/disables Phase Meter (RMS) Upper Limit.</td>
</tr>
<tr>
<td><strong>M.PMRMS?</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

5-34
## POWER METER COMMANDS

**M_PWR:**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-98</td>
<td>Enables/disables Power Meter Alarm.</td>
</tr>
<tr>
<td>DBM[:STATE] b</td>
<td>1 or 0</td>
<td>6-98</td>
<td>Enables/disables dBm digital readout. :STATE portion of the command is optional.</td>
</tr>
<tr>
<td>DBM[:STATE]?</td>
<td>6-98</td>
<td>Returns state of dBm digital readout. :STATE portion of the command is optional.</td>
<td></td>
</tr>
<tr>
<td>EX:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OFFSET n</td>
<td>-99.9 to 99.9</td>
<td>6-98</td>
<td>Sets External Loss/Gain Offset value in dBm.</td>
</tr>
<tr>
<td>OFFSET?</td>
<td>6-98</td>
<td>Returns External Loss/Gain Offset value in dBm.</td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-98</td>
<td>Enables/disables External Loss/Gain Offset.</td>
</tr>
<tr>
<td>STATE?</td>
<td>6-98</td>
<td>Returns External Loss/Gain Offset state setting.</td>
<td></td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVEL n</td>
<td>0.0000 to 0.5000</td>
<td>6-98</td>
<td>Sets Power Meter Lower Limit in W.</td>
</tr>
<tr>
<td></td>
<td>(for ranges: 0.02, 0.05, 0.1, 0.2, and 0.5); 0.00 to 200.00 (for all other ranges).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-98</td>
<td>Enables/disables Power Meter Lower Limit.</td>
</tr>
<tr>
<td>PEAK?</td>
<td>6-99</td>
<td>Returns Power Meter Peak reading in mW (0.0 to 200000.0).</td>
<td></td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-99</td>
<td>Enables/disables Power Meter Peak Hold Feature.</td>
</tr>
<tr>
<td>RANGE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO</td>
<td>6-99</td>
<td>Sets Power Meter range to Autorange.</td>
<td></td>
</tr>
<tr>
<td>UPPER n</td>
<td>0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100, 200</td>
<td>6-99</td>
<td>Sets Power Meter range in W.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-99</td>
<td>Recalls Power Meter environment (routines and settings) stored in memory location n.</td>
</tr>
<tr>
<td>RF:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASSUMEd n</td>
<td>250.0 to 2010000.0</td>
<td>6-99</td>
<td>Sets Assumed RF Frequency, in kHz, within 1 MHz of signal measured.</td>
</tr>
<tr>
<td>ASSUMEd?</td>
<td>6-99</td>
<td>Returns Assumed RF Frequency in kHz.</td>
<td></td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-99</td>
<td>Stores current Power Meter environment (routines and settings) in memory location n.</td>
</tr>
<tr>
<td>TYPE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CW</td>
<td>6-100</td>
<td>Selects Average Power Measurement Type for Power Meter.</td>
<td></td>
</tr>
<tr>
<td>FEAK</td>
<td>6-100</td>
<td>Selects Peak Power Measurement Type for Power Meter.</td>
<td></td>
</tr>
<tr>
<td>RMS</td>
<td>6-100</td>
<td>Selects RMS Power Measurement Type for Power Meter.</td>
<td></td>
</tr>
<tr>
<td>UL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVEL n</td>
<td>0.0000 to 0.5000</td>
<td>6-100</td>
<td>Sets Power Meter Upper Limit in W.</td>
</tr>
<tr>
<td></td>
<td>(for upper ranges: 0.02, 0.05, 0.1, 0.2 and 0.5); 0.00 to 200.00 (for all other ranges)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-100</td>
<td>Enables/disables Power Meter Upper Limit.</td>
</tr>
<tr>
<td>M_PWR?</td>
<td>6-100</td>
<td>Returns a Power Meter reading in mW (0.0 to 200000.0).</td>
<td></td>
</tr>
</tbody>
</table>

## FREQUENCY ERROR METER COMMANDS

**M_RF:**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-96</td>
<td>Enables/disables Alarm.</td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVEL f</td>
<td>0.0000 to 0.1000</td>
<td>6-96</td>
<td>Sets Lower Limit in kHz.</td>
</tr>
<tr>
<td></td>
<td>(for an upper range of 0.1); 0.000 to 100.00 (for upper ranges of 1, 10 and 100)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>6-96</td>
<td>Enables/disables Lower Limit.</td>
<td></td>
</tr>
</tbody>
</table>
## M_RF: PEAK?

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>M_RF:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEAK?</td>
<td></td>
<td>6-96</td>
<td>Returns Frequency Error Meter Peak reading in Hz</td>
</tr>
<tr>
<td>PH b</td>
<td></td>
<td>6-96</td>
<td>Enables/disables Peak Hold Feature.</td>
</tr>
<tr>
<td>RANGE:</td>
<td>AUTO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPPER f</td>
<td>0.1, 1, 10 or 100</td>
<td>6-96</td>
<td>Sets Frequency Error Meter range to Autorange.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-96</td>
<td>Sets Frequency Error Meter range in kHz.</td>
</tr>
<tr>
<td>RESolution f</td>
<td>1 (Gate Time of 1 sec) or 10 (Gate Time of 0.1 sec)</td>
<td>6-97</td>
<td>Recalls Frequency Error Meter environment (routines and settings) stored in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-97</td>
<td>Sets Frequency Error Resolution (Gate Time) in Hz.</td>
</tr>
<tr>
<td>UL:</td>
<td>LEVEL f</td>
<td>0.0000 to 0.1000 (for an upper range of 0.1), 0.000 to 100.000 for upper ranges of 1, 10 and 100</td>
<td>6-97</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-97</td>
<td>Enables/disables Upper Limit.</td>
</tr>
<tr>
<td>M_RF?</td>
<td></td>
<td>6-97</td>
<td>Returns Frequency Error Meter reading in Hz (-100000.0 to +100000.0).</td>
</tr>
</tbody>
</table>

## SIGNAL STRENGTH METER COMMANDS

<table>
<thead>
<tr>
<th>M_SIG:</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PEAK?</td>
<td></td>
<td>6-112</td>
<td>Returns a Signal Strength Meter Peak reading (0 to 100).</td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-112</td>
<td>Enables/disables Peak Hold Feature.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-112</td>
<td>Recalls Signal Strength Meter environment (routines and settings) stored in memory location n.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-112</td>
<td>Stores current Signal Strength Meter environment (routines and settings) in memory location n.</td>
</tr>
<tr>
<td>M_SIG?</td>
<td></td>
<td>6-112</td>
<td>Returns a Signal Strength Meter reading (0 to 100).</td>
</tr>
</tbody>
</table>

## SINAD METER COMMANDS

<table>
<thead>
<tr>
<th>M_SINAD:</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ALARM b</td>
<td>1 or 0</td>
<td>6-109</td>
<td>Enables/disables SINAD Meter Alarm.</td>
</tr>
<tr>
<td>AVERAGE b</td>
<td>1 or 0</td>
<td>6-109</td>
<td>Enables/disables Average Feature.</td>
</tr>
<tr>
<td>FILTER f</td>
<td>600 to 1400</td>
<td>6-109</td>
<td>Sets Notch frequency in kHz.</td>
</tr>
<tr>
<td>INPut type</td>
<td>DEMOD</td>
<td>6-109</td>
<td>Selects SINAD Meter Input.</td>
</tr>
<tr>
<td></td>
<td>(Demod Audio), SINAD (SINAD/BER), XAUDIO (Ext Mod) or FGGEN (Func Gen)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEVEL n</td>
<td>3.0 to 40.0</td>
<td>6-109</td>
<td>Sets SINAD Meter Lower Limit in dB.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-109</td>
<td>Enables/disables SINAD Meter Lower Limit.</td>
</tr>
<tr>
<td>PEAK?</td>
<td></td>
<td>6-109</td>
<td>Returns a SINAD Meter Peak reading in dB (3.0 to 40.0).</td>
</tr>
<tr>
<td>PH b</td>
<td>1 or 0</td>
<td>6-109</td>
<td>Enables/disables SINAD Meter Peak Hold Feature.</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-109</td>
<td>Recalls SINAD Meter environment (routines and settings) stored in memory location n.</td>
</tr>
<tr>
<td>RESolution n</td>
<td>0.1 or 0.5</td>
<td>6-110</td>
<td>Sets SINAD Meter readout resolution in dB.</td>
</tr>
<tr>
<td>RESolution?</td>
<td></td>
<td>6-110</td>
<td>Returns current resolution setting in dB (0.1 or 0.5).</td>
</tr>
<tr>
<td>SELECT</td>
<td>CMESegae</td>
<td>6-110</td>
<td>Same as M_SINAD:SELECT:CWEIGHT command.</td>
</tr>
<tr>
<td></td>
<td>CWEight</td>
<td>6-110</td>
<td>Selects G-Weight Filter.</td>
</tr>
<tr>
<td></td>
<td>LPASS f</td>
<td>6-110</td>
<td>Selects Low-Pass Filter with cutoff frequency of f Hz.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-110</td>
<td>Stores current SINAD Meter environment (routines and settings) in memory location n.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>M_SINAD:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UL:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEEEV n</td>
<td>3.0 to 40.0</td>
<td>6-110</td>
<td>Sets SINAD Meter Upper Limit in dB.</td>
</tr>
<tr>
<td>STAts b</td>
<td>1 or 0</td>
<td>6-110</td>
<td>Enables/disables SINAD Meter Upper Limit.</td>
</tr>
<tr>
<td>M_SINAD?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AF LEVEL METER COMMAND</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M_VRMS?</td>
<td></td>
<td>6-123</td>
<td>Returns Voltage RMS reading of received AF Level (0.00 to 10.00).</td>
</tr>
</tbody>
</table>

**MEASURE (GENERIC) COMMANDS**

Generic commands use optional values, e to signify expected value and r to signify resolution. Expected values help determine Meter range. If r is omitted, the last resolution value is used.

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEASure:</td>
<td></td>
<td>6-170</td>
<td>Returns AF Meter frequency counter reading in Hz (0.0 to 200000.0).</td>
</tr>
<tr>
<td>AUDio? [e,r]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CURRent</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC? [e]</td>
<td></td>
<td></td>
<td>Returns DMM ac current reading in amps (0.00000 to 19.990).</td>
</tr>
<tr>
<td>DC? [e]</td>
<td></td>
<td></td>
<td>Returns DMM dc current reading in amps (0.00000 to 19.990).</td>
</tr>
<tr>
<td>FREQuency? [e,r]</td>
<td></td>
<td></td>
<td>Returns signal frequency reading in kHz (250.0 to 2010000.0 kHz).</td>
</tr>
<tr>
<td>M?IC?</td>
<td></td>
<td></td>
<td>Returns 0 if receiving MIC/ACC input; 1 otherwise.</td>
</tr>
<tr>
<td>PHAss? [e]</td>
<td></td>
<td></td>
<td>Returns Phase Meter reading in radians (0.00 to 10.00).</td>
</tr>
<tr>
<td>POWer? [e]</td>
<td></td>
<td></td>
<td>Returns Power Meter reading in mV (0.0 to 100000.0).</td>
</tr>
<tr>
<td>RESistance? [e]</td>
<td></td>
<td></td>
<td>Returns DMM resistance reading in kΩ (0.00000 to 19980).</td>
</tr>
<tr>
<td>SINAD? [r]</td>
<td></td>
<td></td>
<td>Returns SINAD Meter reading in dB (3.0 to 40.0).</td>
</tr>
<tr>
<td>SQUElch?</td>
<td></td>
<td></td>
<td>Returns 1 if squelch broken; 0 otherwise.</td>
</tr>
<tr>
<td>TEMPerature:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMBient?</td>
<td></td>
<td></td>
<td>Returns ambient temperature in °C (0.00000 to 100.00000).</td>
</tr>
<tr>
<td>POWer?</td>
<td></td>
<td></td>
<td>Returns Power Termination temperature in °C (0.00000 to 100.00000).</td>
</tr>
<tr>
<td>VOLTage:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC? [e]</td>
<td></td>
<td></td>
<td>Returns DMM ac voltage reading in volts (0.00000 to 1000.00).</td>
</tr>
<tr>
<td>DC? [e]</td>
<td></td>
<td></td>
<td>Returns DMM dc voltage reading in volts (0.00000 to 1000.00).</td>
</tr>
<tr>
<td>SUPply? n</td>
<td>-15, 5 or 15</td>
<td>6-171</td>
<td>Returns voltage measurement of n power supply in volts (0.00 to 20.00).</td>
</tr>
</tbody>
</table>

**MODULATION METER COMMANDS**

See M_MOD:

**FLASH MEMORY COMMANDS**

MMEMory:

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTribute:</td>
<td></td>
<td>6-166</td>
<td>Sets the File Attribute of the specified Flash Memory file to DELETE. f is Flash Memory file name.</td>
</tr>
<tr>
<td>DELeete &quot;f&quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HIDe &quot;f&quot;</td>
<td></td>
<td></td>
<td>Sets the File Attribute of the specified Flash Memory file to HIDE. f is Flash Memory file name.</td>
</tr>
<tr>
<td>INITialize &quot;f&quot;</td>
<td></td>
<td></td>
<td>Sets the File Attribute of the specified Flash Memory file to INIT. f is Flash Memory file name.</td>
</tr>
<tr>
<td>PACK &quot;f&quot;</td>
<td></td>
<td></td>
<td>Sets the File Attribute of the specified Flash Memory file to PACK. f is Flash Memory file name.</td>
</tr>
<tr>
<td>ATTribute? &quot;f&quot;</td>
<td></td>
<td></td>
<td>Returns the Attribute (DELETE, HIDE, INIT or PACK) of the specified Flash Memory. f is Flash Memory file name.</td>
</tr>
</tbody>
</table>

5-37
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMEMory:CATalog:ENTRY?</td>
<td></td>
<td>6-166</td>
<td>Returns file entry (file name, file type, file size) for given index. Returns $S$ if past end of directory or --- for deleted file. ( n ) is line number (index) in Flash Memory File Directory.</td>
</tr>
<tr>
<td>FREE?</td>
<td></td>
<td>6-166</td>
<td>Returns available file space, in bytes.</td>
</tr>
<tr>
<td>USED?</td>
<td></td>
<td>6-166</td>
<td>Returns file space used, in bytes.</td>
</tr>
<tr>
<td>CATalog?</td>
<td></td>
<td>6-166</td>
<td>Returns Flash Memory status. First number returned is memory space used in bytes. Second number returned is memory space available in bytes. Remainder data is returned in sets of 3 consisting of file name, file type and file size for each file stored in Flash Memory.</td>
</tr>
<tr>
<td>DELete 'f'</td>
<td></td>
<td>6-167</td>
<td>Deletes Flash Memory file, but does not release memory space until Pack operation is done.</td>
</tr>
<tr>
<td>INITIALize</td>
<td></td>
<td>6-167</td>
<td>Erases all files stored in Flash Memory.</td>
</tr>
<tr>
<td>INITIALize?</td>
<td></td>
<td>6-167</td>
<td>Returns 1 if file system has been initialized; otherwise 0.</td>
</tr>
<tr>
<td>LOAD:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CALibration 'f'</td>
<td></td>
<td>6-167</td>
<td>Loads Calibration Data from Flash Memory into Test Set memory. 'f' is Flash Memory file name.</td>
</tr>
<tr>
<td>DATA 'v','f'</td>
<td></td>
<td>6-167</td>
<td>Loads variable stored as 'v' into Test Set memory with name 'v'. 'f' is name of variable; 'f' is Flash Memory file name.</td>
</tr>
<tr>
<td>MACRo 'm','f'</td>
<td></td>
<td>6-167</td>
<td>Loads macro and variables stored as file name 'f' from Flash to Test Set memory, executing macro 'm'. If 'm' is '*', designated macro (See MMEMory:STORE:MACRo) is executed. If 'm' is macro name, that macro is executed. If 'm' is omitted (''''), no macro is executed.</td>
</tr>
<tr>
<td>STATe 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-167</td>
<td>Loads Test Set State stored as 'f' from Flash Memory into Auxiliary Functions &quot;Store Parameters Menu&quot; as entry 'n'. 'n' is number of stored state of Test Set (( n = 0 ) loads current state).</td>
</tr>
<tr>
<td>TRACE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANLZ 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-167</td>
<td>Loads Spectrum Analyzer trace stored as 'f' into Spectrum Analyzer &quot;Store Parameters Menu&quot; as entry 'n'. 'n' is number of stored trace (( n = 0 ) loads live trace).</td>
</tr>
<tr>
<td>SCOPe 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-167</td>
<td>Loads Oscilloscope trace stored as 'f' into Oscilloscope &quot;Store Parameters Menu&quot; as entry 'n'. 'n' is number of stored trace (( n = 0 ) loads live trace).</td>
</tr>
<tr>
<td>PACK</td>
<td></td>
<td>6-168</td>
<td>Packs Flash Memory and frees memory space from deleted files.</td>
</tr>
<tr>
<td>STORE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CALibration 'f'</td>
<td></td>
<td>6-168</td>
<td>Stores Test Set Calibration Data into Flash Memory. 'f' is Flash Memory file name.</td>
</tr>
<tr>
<td>DATA 'v','f'</td>
<td></td>
<td>6-168</td>
<td>Stores variable 'v' into Flash Memory as 'f'. 'v' is name of variable; 'f' is Flash Memory file name.</td>
</tr>
<tr>
<td>MACRo 'm','f'</td>
<td></td>
<td>6-168</td>
<td>Stores all Test Set macros and variables (except free variables) in Flash Memory as 'f' with macro 'm' specified as designated macro. 'm' is name of designated macro; 'f' is Flash Memory file name.</td>
</tr>
<tr>
<td>STATe 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-168</td>
<td>Stores entry 'n' of Auxiliary Functions &quot;Store Parameters Menu&quot; as 'f' in Flash Memory. (( n = 0 ) stores current state.) 'n' is number of stored state of Test Set.</td>
</tr>
<tr>
<td>TRACE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANLZ 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-168</td>
<td>Stores entry 'n' (stored trace) of Spectrum Analyzer &quot;Store Parameters Menu&quot; as 'f' in Flash Memory. (( n = 0 ) stores live trace.) 'n' is number of stored trace.</td>
</tr>
<tr>
<td>SCOPe 'n','f'</td>
<td>0 to 9, flash file name</td>
<td>6-168</td>
<td>Stores entry 'n' (stored trace) of Oscilloscope &quot;Store Parameters Menu&quot; as 'f' in Flash Memory. (( n = 0 ) stores live trace.) 'n' is number of stored trace.</td>
</tr>
<tr>
<td>TYPE? 'f'</td>
<td></td>
<td>6-168</td>
<td>Returns file type. 'f' is Flash Memory file name.</td>
</tr>
</tbody>
</table>
## COMMANDS

### OSCILLOSCOPE COMMANDS
See Scope Commands

### PHASE METER COMMANDS
See M_PM:

### PHASE METER (RMS) COMMANDS
See M_PMRMS:

### POWER METER COMMANDS
See M_PWR:

### PROGRAM COMMANDS

**PROGram:**

**STARTup:**

**DELETE**

<table>
<thead>
<tr>
<th>NAME <em>name</em></th>
</tr>
</thead>
</table>

| NAME? |

**PRESS TO TALK COMMAND**

See Function Generator Commands

### RECEIVER COMMANDS

**RECeiver:**

<table>
<thead>
<tr>
<th>AGC:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO</td>
<td></td>
</tr>
<tr>
<td>MANual n</td>
<td>0 to 255</td>
</tr>
<tr>
<td>USER:xxx</td>
<td>MEASURE, SPEECH, DATA, HIGH, TYPE1, TYPE2 or TYPE3</td>
</tr>
<tr>
<td>CHANnel n</td>
<td>1 to 2047, depending on format</td>
</tr>
<tr>
<td>CHA Nnel:</td>
<td></td>
</tr>
<tr>
<td>BAND?</td>
<td>6-19 Returns band for Receiver Channel Format. Returns one of the following bands for the associated channel format: (NADC) U8, U4 or HY; (ETACS) NOT AVAILABLE; (NAMPS) LOWER, MIDDLE or UPPER.</td>
</tr>
</tbody>
</table>

**FORMat:**

<table>
<thead>
<tr>
<th>AMPS:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FORward</td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
</tr>
<tr>
<td>ETACS:</td>
<td></td>
</tr>
<tr>
<td>FORward</td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
</tr>
<tr>
<td>NADC:</td>
<td></td>
</tr>
<tr>
<td>BAND:xx</td>
<td>U8, U4 or HYper</td>
</tr>
<tr>
<td>FORward</td>
<td></td>
</tr>
<tr>
<td>REVerse</td>
<td></td>
</tr>
</tbody>
</table>

<p>| 6-19 Selects AMPS (NADC-U8) Forward channels. |
| 6-19 Selects AMPS (NADC-U8) Reverse channels. |
| 6-19 Selects ETACS Forward channels. |
| 6-19 Selects ETACS Reverse channels. |
| 6-20 Selects NADC band. |
| 6-20 Selects NADC Forward. Utilizes current setting of NADC band. |
| 6-20 Selects NADC Reverse. Utilizes current setting of NADC band. |</p>
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECeiver: CHANnel: FORMat: NAMPS: BAND: x</td>
<td>Lower, Middle or Upper</td>
<td>6-20</td>
<td>Selects NAMPS Narrow Analog channel designator.</td>
</tr>
<tr>
<td>DCS: INVert?</td>
<td></td>
<td>6-21</td>
<td>Returns three octal digits in inverted mode, decoded from Digital Coded Squelch (DCS) code. Returns -1 if none available or if invalid for inverted DCS.</td>
</tr>
<tr>
<td>NORMat?</td>
<td></td>
<td>6-21</td>
<td>Returns three octal digits decoded from the DCS code. Returns -1 if none available or if invalid for normal DCS.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0, DTMF, TONE or DIGital</td>
<td>6-21</td>
<td>Enables/disables DCS decoding. Sets Receiver decoding type.</td>
</tr>
<tr>
<td>DIGital type</td>
<td>DCS, DCSINV, POCASG, DSAT or DST</td>
<td>6-21</td>
<td>Displays Deviation Meter (RMS) reading on Receiver Operation Screen when followed by SCREEN:RECeiver command. Sets digital type.</td>
</tr>
<tr>
<td>DISTortion</td>
<td></td>
<td>6-22</td>
<td>Displays Distortion Meter reading on Receiver Operation Screen when followed by SCREEN:RECeiver command.</td>
</tr>
<tr>
<td>DMM</td>
<td></td>
<td>6-22</td>
<td>Displays Digital Multimeter reading on Receiver Operation Screen when followed by SCREEN:RECeiver command.</td>
</tr>
<tr>
<td>DSAT: STATE b</td>
<td>1 or 0</td>
<td>6-22</td>
<td>Enables/disables DSAT Decoding Function. Returns DSAT reading. Returns -1 if none available or invalid for normal transmission.</td>
</tr>
<tr>
<td>DSAT?</td>
<td></td>
<td>6-22</td>
<td>Enables/disables DSAT Decoding Function. Returns -1 if none available or invalid for normal transmission.</td>
</tr>
<tr>
<td>DST: STATE b</td>
<td>1 or 0</td>
<td>6-22</td>
<td>Enables/disables DST Decoding Function. Returns DST reading. Returns -1 if none available or invalid for normal transmission.</td>
</tr>
<tr>
<td>DST?</td>
<td></td>
<td>6-22</td>
<td>Enables/disables DST Decoding Function. Returns -1 if none available or invalid for normal transmission.</td>
</tr>
<tr>
<td>DTMF: STATE b</td>
<td></td>
<td>6-22</td>
<td>Enables/disables DTMF Decoding Function. Returns string of decoded digits or -1 if nothing decoded.</td>
</tr>
<tr>
<td>DTMF?</td>
<td></td>
<td>6-22</td>
<td>Returns frequency of first signal with amplitude larger than Find reference level. Returns 0 if no signal is found.</td>
</tr>
<tr>
<td>FIND: FREQuency?</td>
<td>-110 to -5</td>
<td>6-23</td>
<td>Sets Find reference level in dBm.</td>
</tr>
<tr>
<td>REFerence n</td>
<td></td>
<td>6-23</td>
<td>Returns Find reference level in dBm.</td>
</tr>
<tr>
<td>REFerence?</td>
<td>1 to 4</td>
<td>6-23</td>
<td>Zeros FM Deviation Meter, Displays Receiver Operation Screen and selects PM Modulation. Sets Receiver Frequency.</td>
</tr>
<tr>
<td>FREQuency n [units]</td>
<td>250000.0 to 2010000000.0 [Hz]</td>
<td>6-23</td>
<td>Returns Receiver Frequency in kHz (250.0 to 2010000.0).</td>
</tr>
</tbody>
</table>

*units - Hz, KHz or MHz (default units are KHz)*
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECeiver:</td>
<td></td>
<td>6-24</td>
<td>Selects ANTENNA IN Connector as Receiver Input Source. Displays Signal</td>
</tr>
<tr>
<td>INPut:</td>
<td></td>
<td></td>
<td>Strength Meter reading on Receiver Operation Screen.</td>
</tr>
<tr>
<td>ANTenna</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATTenuation n</td>
<td>0, 5, 10, 15, 20,</td>
<td>6-24</td>
<td>Sets Receiver Input Attenuation in dB.</td>
</tr>
<tr>
<td></td>
<td>25, 30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATTenuation: LNA</td>
<td></td>
<td>6-24</td>
<td>Sets Receiver Input Attenuation to LNA.</td>
</tr>
<tr>
<td>LNA?</td>
<td></td>
<td>6-24</td>
<td>Returns current state of Low Noise Amplifier. Returns 1 if LNA is selected;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0 otherwise.</td>
</tr>
<tr>
<td>ATTenuation? TR</td>
<td></td>
<td>6-24</td>
<td>Returns current value of input Attenuation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODE type</td>
<td>DIRECT, CHANnel,</td>
<td>6-25</td>
<td>Selects Receiver Mode.</td>
</tr>
<tr>
<td></td>
<td>SCAN, LIST,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FLScan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODE?</td>
<td></td>
<td>6-25</td>
<td>Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FREQUENCY LIST SCAN).</td>
</tr>
<tr>
<td>MODMeter</td>
<td></td>
<td>6-25</td>
<td>Displays Modulation Meter reading on Receiver Operation Screen when</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>followed by SCREEN:RECeiver command and AM is selected modulation.</td>
</tr>
<tr>
<td>MODulation: AMn</td>
<td>1 or 2</td>
<td>6-25</td>
<td>Selects Amplitude Modulation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-25</td>
<td>Selects Beat Frequency Oscillation.</td>
</tr>
<tr>
<td>MODulation? FMn</td>
<td>1, 2, 3 or 4</td>
<td>6-25</td>
<td>Selects Frequency Modulation.</td>
</tr>
<tr>
<td>MODulation? FM or DATA</td>
<td>FM, AM, BFO, PM</td>
<td>6-26</td>
<td>Selects User Defined Modulation.</td>
</tr>
<tr>
<td>MODulation? type</td>
<td>or DATA [FM DATA]</td>
<td>6-26</td>
<td></td>
</tr>
<tr>
<td>USER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILTER f</td>
<td>3, 30 or 300</td>
<td>6-26</td>
<td>Sets User Defined Modulation IF Filter in kHz.</td>
</tr>
<tr>
<td>FILTER?</td>
<td></td>
<td>6-26</td>
<td>Returns current setting of User Defined Modulation IF Filter in kHz.</td>
</tr>
<tr>
<td>POST:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>APASs</td>
<td></td>
<td>6-26</td>
<td>Selects All Pass Post Detection Filter for User Defined Modulation.</td>
</tr>
<tr>
<td>BPASs &lt;f&gt;</td>
<td>0.5 to 20, 0.1 to 30</td>
<td>6-26</td>
<td>Selects Bandpass Post Detection Filter for User Defined Modulation, and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>sets lower cutoff frequency (fL) and higher cutoff frequency (fH) in kHz.</td>
</tr>
<tr>
<td>BPASs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HIGH f</td>
<td>0.5 to 20.0</td>
<td>6-26</td>
<td>Sets Lower cutoff frequency of Band-Pass Post Detection Filter in kHz.</td>
</tr>
<tr>
<td>HIGH?</td>
<td></td>
<td>6-26</td>
<td>Returns Lower cutoff frequency of Band-Pass Post Detection Filter in kHz.</td>
</tr>
<tr>
<td>LOW f</td>
<td>0.1 to 30.0</td>
<td>6-27</td>
<td>Sets Upper cutoff frequency of Band-Pass Post Detection Filter in kHz.</td>
</tr>
<tr>
<td>LOW?</td>
<td></td>
<td>6-27</td>
<td>Returns Upper cutoff frequency of Band-Pass Post Detection Filter in kHz.</td>
</tr>
<tr>
<td>CWEight</td>
<td></td>
<td>6-27</td>
<td>Selects C-Weighted Post Detection Filter (C-message noise weighting curve</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>response) for User Defined Modulation.</td>
</tr>
<tr>
<td>HPASs &lt;f&gt;</td>
<td>0.5 to 20</td>
<td>6-27</td>
<td>Selects High-Pass Post Detection Filter for User Defined Modulation, and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>specifies cutoff frequency in kHz.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RECeiver:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODulation:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>POST:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LPASs lh</td>
<td>0.1 to 30</td>
<td>6-28</td>
<td>Selects Low-Pass Post Detection Filter for User Defined Modulation, and specifies cutoff frequency in kHz.</td>
</tr>
<tr>
<td>POST?</td>
<td></td>
<td>6-28</td>
<td>Returns current setting of User Defined Post Detection Filter (APAS, LPAS, HPAS, BPAS or CWE).</td>
</tr>
<tr>
<td>MODulation?</td>
<td></td>
<td>6-28</td>
<td>Returns Receiver Modulation (FM, AM, BFO, PM, DATA or USER).</td>
</tr>
<tr>
<td>OUTput:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUDIO b</td>
<td>1 or 0</td>
<td>6-28</td>
<td>Connects/disconnects Demodulated Audio to AUDIO OUT Connector.</td>
</tr>
<tr>
<td>DEMOD b</td>
<td>1 or 0</td>
<td>6-28</td>
<td>Connects/disconnects Demodulated Audio to DEMOD OUT Connector.</td>
</tr>
<tr>
<td>SPEAKER b</td>
<td>1 or 0</td>
<td>6-28</td>
<td>Connects/disconnects Demodulated Audio to Speaker.</td>
</tr>
<tr>
<td>PMRms</td>
<td></td>
<td>6-28</td>
<td>Displays Phase Meter (RMS) reading when followed by SCREEN:REceiver command.</td>
</tr>
<tr>
<td>POCSAG:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAPcode?</td>
<td></td>
<td>6-29</td>
<td>Returns received capcode or -1 if a capcode is not received.</td>
</tr>
<tr>
<td>MESSAGE?</td>
<td></td>
<td>6-29</td>
<td>Returns message from decoded POCSAG signal or -1 if none available.</td>
</tr>
<tr>
<td>RATE b</td>
<td>high = 1, low = 0</td>
<td>6-29</td>
<td>Sets POCSAG rate.</td>
</tr>
<tr>
<td>RATE?</td>
<td></td>
<td>6-29</td>
<td>Returns 1 if POCSAG rate is high, 0 if rate is low.</td>
</tr>
<tr>
<td>STARe b</td>
<td>1 or 0</td>
<td>6-29</td>
<td>Enables/disables POCSAG decoding.</td>
</tr>
<tr>
<td>TYPE?</td>
<td></td>
<td>6-29</td>
<td>Returns POCSAG Function Type (TONE 1 BEEP, TONE 2 BEEPS, TONE 3 BEEPS, TONE 4 BEEPS, NUMERIC, ALPHANUMERIC, NO MESSAGE).</td>
</tr>
<tr>
<td>RCL n</td>
<td>1 to 9</td>
<td>6-29</td>
<td>Recalls Receiver environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>SCAN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ABORT</td>
<td></td>
<td>6-30</td>
<td>Stops frequency scan or frequency list scan depending on present mode.</td>
</tr>
<tr>
<td>CONTinue</td>
<td></td>
<td>6-30</td>
<td>Starts frequency scan or frequency list scan depending on present mode.</td>
</tr>
<tr>
<td>FREQLlist:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAUSE n</td>
<td>0.0 to 99.9</td>
<td>6-31</td>
<td>Frequency List Scan Pause Time. Specifies time period in seconds for Receiver to sit on a frequency if squelch is broken.</td>
</tr>
<tr>
<td>PAUSE?</td>
<td></td>
<td>6-31</td>
<td>Returns current value of Frequency List Scan Pause Time.</td>
</tr>
<tr>
<td>RATE n</td>
<td>0.02 to 99.999</td>
<td>6-31</td>
<td>Frequency List Scan Rate. Specifies time period in seconds for Receiver to sit on a frequency unless squelch is broken.</td>
</tr>
<tr>
<td>RATE?</td>
<td></td>
<td>6-31</td>
<td>Returns current value of Frequency List Scan Rate.</td>
</tr>
<tr>
<td>SQUElch b</td>
<td>1 or 0</td>
<td>6-31</td>
<td>Enables/disables Squelch.</td>
</tr>
<tr>
<td>SQUElch?</td>
<td></td>
<td>6-31</td>
<td>Returns current state of Squelch.</td>
</tr>
<tr>
<td>FREQuency?</td>
<td></td>
<td>6-31</td>
<td>Returns frequency currently being scanned in kHz.</td>
</tr>
<tr>
<td>INCrement f</td>
<td>0.0 to 201000.0</td>
<td>6-30</td>
<td>(Frequency Scan mode only) Sets Receiver Scan Increment in kHz.</td>
</tr>
<tr>
<td>PAUSE t</td>
<td>0.0 to 99.9</td>
<td>6-30</td>
<td>(Frequency Scan mode only) Receiver Pause Rate in sec. Receiver Scan pause time is length of time Scan Function pauses at frequency with broken squelch. Scan Function stops permanently on squelch broken frequency if pause set to 0.0.</td>
</tr>
<tr>
<td>PAUSE?</td>
<td></td>
<td>6-30</td>
<td>(Frequency Scan mode only) Returns 1 if scanning is paused (stopped); 0 if currently scanning.</td>
</tr>
<tr>
<td>RATE t</td>
<td>0.00 to 99.99</td>
<td>6-30</td>
<td>Sets Receiver Scan Rate in sec. Receiver Scan rate is time each frequency is scanned with squelch unbroken.</td>
</tr>
<tr>
<td>START f</td>
<td>250.0 to 201000.0</td>
<td>6-30</td>
<td>Sets Receiver Scan starting frequency in kHz.</td>
</tr>
<tr>
<td>STOP f</td>
<td>250.0 to 201000.0</td>
<td>6-31</td>
<td>Sets Receiver Scan stopping frequency in kHz.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RECeiver:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SINAD</td>
<td></td>
<td>6-32</td>
<td>Displays SINAD Meter reading when followed by a</td>
</tr>
<tr>
<td>SQUElch n</td>
<td>0.0 to 1.0</td>
<td>6-32</td>
<td>Sets squelch to n.</td>
</tr>
<tr>
<td>SQUElch?</td>
<td></td>
<td>6-32</td>
<td>Returns squelch level.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-32</td>
<td>Stores current Receiver environment (routings and settings) in memory location n.</td>
</tr>
<tr>
<td>TONE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DURation? <em>id</em></td>
<td>0 to 9 and A to T</td>
<td>6-32</td>
<td>Returns Duration in ms (20 to 9999) of specified User Defined Audio Tone. <em>id</em> is character associated with the defined tone.</td>
</tr>
<tr>
<td>FREQuency? <em>id</em></td>
<td>0 to 9 and A to T</td>
<td>6-32</td>
<td>Returns frequency setting in Hz of specified User Defined Audio Tone <em>id</em>.</td>
</tr>
<tr>
<td>INPUT n</td>
<td>0 - Demod Audio, 1 - SINAD/BER, 2 - Ext Mod</td>
<td>6-32</td>
<td>Selects decode signal input.</td>
</tr>
<tr>
<td>STATe b TYPE xxx</td>
<td>CCIR, EEA, EIA (U.S.), ZVEI, DDZVEI, DZVEI, NATEL, EURO, TONE56, CCIRH, CCIRH4, USER</td>
<td>6-33</td>
<td>Enables/disables Audio Tone decoding.</td>
</tr>
<tr>
<td>USER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEFINE &quot;id&quot;,f,d</td>
<td>0 to 9 and A to T; 0.0 to 9999.9; 20 to 9999</td>
<td>6-32</td>
<td>Defines a User Defined Audio Tone and duration to be received; where id is character associated with defined tone, f is frequency in Hz of defined tone and d is duration of defined tone in ms.</td>
</tr>
<tr>
<td>TONE?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOLUME n</td>
<td>0.0 to 1.0</td>
<td>6-33</td>
<td>Returns decoded Audio Tone sequence or -1 if not available.</td>
</tr>
<tr>
<td>VOLUME:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO b</td>
<td>1 or 0</td>
<td>6-33</td>
<td>Enables/disables Automatic Volume Control.</td>
</tr>
<tr>
<td>AUTO?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOLUME?</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**RF GENERATOR COMMANDS**
See Generator Commands

**SCOPE COMMANDS**

**SCOPe:**

| ARM                      | 6-66 | |
|--------------------------|------||
| AVERAGE n                | 1 to 100 (default is 100) | 6-66 | Selects Oscilloscope Average Mode using n samples. |
| COMPare n                | 1 to 9 | 6-66 | Selects Compare Mode for Oscilloscope. Trace stored at n memory location is compared to the current Live Trace. |
| COUpling type FULL       | AC, DC or GROUND | 6-66 | Selects full size Oscilloscope display for RF Generator, Receiver and Duplex Operation Screens. |
| HORIZontal n             | -12 to 12 | 6-66 | Sets Horizontal Time Offset to n major divisions. (-12 to -1 are major divisions before the trigger; 1 to 12 are major divisions after the trigger.) |

5-43
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOPE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INPUT:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILTER:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CWEight:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-66</td>
<td>Enables/disables Internal C-Weight filter.</td>
</tr>
<tr>
<td>HPASS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQUENCY f</td>
<td>0.2 to 100</td>
<td>6-66</td>
<td>Sets Internal High-Pass frequency in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-66</td>
<td>Enables/disables Internal High-Pass filter.</td>
</tr>
<tr>
<td>LPASS:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQUENCY f</td>
<td>0.2 to 50</td>
<td>6-67</td>
<td>Sets Internal Low-Pass frequency in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-67</td>
<td>Enables/disables Internal Low-Pass filter.</td>
</tr>
<tr>
<td>NOTch:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREQUENCY f</td>
<td>0.5 to 1.5</td>
<td>6-67</td>
<td>Sets Notch center frequency in kHz.</td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-67</td>
<td>Enables/disables Internal Notch Filter.</td>
</tr>
<tr>
<td>INTernal type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IF (Recr IF), DEMOD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Demod Audio), POWer (RF-Fwr Lvl), SINAD (SINAD/BER), FUNCTION (Func Gen), XAUDIO (Ext Med)</td>
<td>6-67</td>
<td>Selects Internal Oscilloscope Input.</td>
<td></td>
</tr>
<tr>
<td>LEVEL n</td>
<td>0 to 255</td>
<td>6-67</td>
<td>Specifies Trigger level. (0 corresponds to bottom of Oscilloscope Display; 255 corresponding to top.)</td>
</tr>
<tr>
<td>LIVE b</td>
<td>1 = SCREEN.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MARKer:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AOFF</td>
<td></td>
<td>6-68</td>
<td>Disables both Markers.</td>
</tr>
<tr>
<td>DELTA:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLitude?</td>
<td></td>
<td>6-68</td>
<td>Returns voltage difference of the Trace Marker 1 and Trace Marker 2 crossings in volts. Valid only for Oscilloscope Inputs AC, DC and GND.</td>
</tr>
<tr>
<td>POINT?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRACK b</td>
<td>1 or 0</td>
<td>6-68</td>
<td>Returns voltage of live Trace at Marker 1 crossing in volts. Valid only for Oscilloscope Inputs AC, DC and GND.</td>
</tr>
<tr>
<td>MARKER1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLitude?</td>
<td></td>
<td>6-68</td>
<td>Sets Marker 1 position in graticules (100 graticules is equal to Oscilloscope display width).</td>
</tr>
<tr>
<td>POINT n</td>
<td>1 to 100</td>
<td>6-68</td>
<td>Returns Marker 1 position in graticules.</td>
</tr>
<tr>
<td>POINT?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-68</td>
<td>Enables/disables Marker 1.</td>
</tr>
<tr>
<td>STATE?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME?</td>
<td></td>
<td>6-68</td>
<td>Returns Marker 1 position in ms from left edge of Oscilloscope display.</td>
</tr>
<tr>
<td>MARKER2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPLitude?</td>
<td></td>
<td>6-68</td>
<td>Returns voltage of live Trace at Marker 2 crossing in volts. Valid only for Oscilloscope Inputs AC, DC and GND.</td>
</tr>
<tr>
<td>POINT n</td>
<td>1 to 100</td>
<td>6-68</td>
<td>Sets Marker 2 position in graticules (100 graticules is equal to Oscilloscope display width).</td>
</tr>
<tr>
<td>POINT?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATE b</td>
<td>1 or 0</td>
<td>6-68</td>
<td>Enables/disables Marker 2.</td>
</tr>
<tr>
<td>STATE?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIME?</td>
<td></td>
<td>6-69</td>
<td>Returns Marker 2 position in ms from left edge of Oscilloscope display.</td>
</tr>
</tbody>
</table>

5-44
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOPe:</td>
<td></td>
<td>6-70</td>
<td>Draws Oscilloscope grid on attached plotter.</td>
</tr>
<tr>
<td>PLOT:</td>
<td></td>
<td>6-70</td>
<td>Draws Oscilloscope trace on attached plotter.</td>
</tr>
<tr>
<td>GRID</td>
<td></td>
<td>6-70</td>
<td>Draws Oscilloscope units on attached plotter.</td>
</tr>
<tr>
<td>TRACE</td>
<td></td>
<td>6-70</td>
<td>Selects 1/4 size Oscilloscope display for RF Generator, Receiver and Duplex Operation Screens.</td>
</tr>
<tr>
<td>UNITS</td>
<td></td>
<td></td>
<td>Recalls Oscilloscope Trace and environment (routings and settings) stored in memory location n.</td>
</tr>
<tr>
<td>QTR</td>
<td>1 to 9</td>
<td>6-70</td>
<td>Sets Oscilloscope scale.</td>
</tr>
<tr>
<td>RCL n</td>
<td>(AC, DC or GND Input) 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000, 20000, 50000, 100000 (mV/div).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCALE n</td>
<td>(Demod Audio Input with FM) 2, 4, 10, 20 (kHz/div)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Func Gen or Ext Mod Input) 500, 1000, 2500 (mV/div)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCALE?</td>
<td></td>
<td>6-71</td>
<td>Returns Oscilloscope scale in mV/div if input is AC, DC, GND, SINAD/BER, Func Gen or Ext Mod. Returns Oscilloscope scale in kHz/div if input is Demod Audio. Returns Oscilloscope scale in W if input is RF Pwr Lvl.</td>
</tr>
<tr>
<td>SOURce EXTERNAL</td>
<td></td>
<td>6-71</td>
<td>Routes Oscilloscope Input from the SCOPE IN Connector. Selects AC Ext for Oscilloscope Input.</td>
</tr>
<tr>
<td>SOURce INTernal</td>
<td></td>
<td>6-71</td>
<td>Disconnects Oscilloscope Input from the SCOPE IN Connector. Selects Demod Audio for Oscilloscope Input.</td>
</tr>
<tr>
<td>STATe b</td>
<td>1 or 0</td>
<td>6-71</td>
<td>Enables/disables Oscilloscope Audio for Oscilloscope Input. Enables/disables Oscilloscope display in RF Generator, Receiver and Duplex Operation Screens.</td>
</tr>
<tr>
<td>STORE n</td>
<td>1 to 9</td>
<td>6-71</td>
<td>Stores current Oscilloscope Trace and environment (routings and settings) in memory location n.</td>
</tr>
<tr>
<td>SWEep n</td>
<td>1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000, 20000, 50000, 100000</td>
<td>6-71</td>
<td>Sets Oscilloscope sweep rate in μs/div.</td>
</tr>
<tr>
<td>SWEep n MS</td>
<td>1, 2, 5, 10, 20, 50, 100</td>
<td>6-71</td>
<td>Sets Oscilloscope sweep rate in ms/div.</td>
</tr>
<tr>
<td>SWEep n US</td>
<td>1, 2, 5, 10, 20, 50, 100, 200, 500</td>
<td>6-72</td>
<td>Sets Oscilloscope sweep rate in μs/div.</td>
</tr>
<tr>
<td>SWEep?</td>
<td></td>
<td>6-72</td>
<td>Returns Oscilloscope sweep rate in μs/div.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>SCOPE:TRACE:DATA</td>
<td></td>
<td></td>
<td>For the following SCOPE:TRACE commands, the Oscilloscope display is divided into 400 positions horizontally (0 signifying the left edge of the display, 399 signifying the right edge of the display) and 255 values vertically (0 signifying the bottom of the display, 255 signifying the top of the display).</td>
</tr>
<tr>
<td>(DATA)</td>
<td>n,offset,value,.....,value</td>
<td>n = 1 to 9, offset = 0 to 399, value = 0 to 255</td>
<td>6-73 Replaces points of stored Trace n with specified vertical values (value) starting with horizontal offset (offset). Intended for remote GPIB or RS-232 use only; command and data are sent via GPIB or RS-232 from an external application, etc.</td>
</tr>
<tr>
<td>(DATA?)</td>
<td>[[n],offset,points]</td>
<td>n = 0 to 9, offset = 0 to 399 (0 default), points = 1 to 400 (400 default)</td>
<td>6-73 Returns the vertical values for each of the points specified, starting with offset (horizontal offset) of trace n. Intended for remote GPIB or RS-232 use only; data is sent out GPIB or RS-232 for use by an external application, etc.</td>
</tr>
<tr>
<td>(GET)</td>
<td>name,n</td>
<td>n = 0 to 9, 0 = Live Trace</td>
<td>6-73 Assigns vertical values of trace n (in graticules) to each element of declared array name. First element of array corresponds with first point of trace n. If array is less than 400 values in length, the remaining portion of trace is left unassigned to an array. See SCOPE:TRACE:PUT.</td>
</tr>
<tr>
<td>(GET?)</td>
<td>n,offset</td>
<td>n = 0 to 9, 0 = Live Trace</td>
<td>6-73 Returns vertical value of a point in Trace n located at offset (horizontal position from the left edge of display).</td>
</tr>
<tr>
<td>(MAX?)</td>
<td>[[n],offset,points]</td>
<td>n = 0 to 9, offset = 0 to 399 (0 default), points = 1 to 400 (400 default)</td>
<td>6-73 Returns the maximum point of Trace n within specified number of points starting with given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the Live Trace or a Trace stored in memory. Intended for remote GPIB or RS-232 use only; data is sent out GPIB or RS-232 for use by an external application, etc.</td>
</tr>
<tr>
<td>(MIN?)</td>
<td>[[n],offset,points]</td>
<td>n = 0 to 9, offset = 0 to 399 (0 default), points = 1 to 400 (400 default)</td>
<td>6-74 Returns the minimum point of Trace n within specified number of points starting with the given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the Live Trace or a Trace stored in memory. Intended for remote GPIB or RS-232 use only; data is sent via GPIB or RS-232 from an external application, etc.</td>
</tr>
<tr>
<td>(PUT)</td>
<td>name,n</td>
<td>n = 1 to 9</td>
<td>6-74 Assigns values of an array to trace n. name is array name. n signifies stored trace. Each element value represent a vertical value for associated horizontal positions. If array is less than 400 values in length (stored trace length), the array values are assigned to the trace the length of the array starting from horizontal position 0. The remaining portion of the stored trace is left intact. See SCOPE:TRACE:GET.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>RANGE</td>
<td>PAGE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SCOPE:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRIGger:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO</td>
<td>6-74</td>
<td></td>
<td>Sets Trigger to Auto Sweep.</td>
</tr>
<tr>
<td>IMMediate</td>
<td>6-74</td>
<td></td>
<td>Triggers Oscilloscope as soon as command is interpreted or sequenced in a macro.</td>
</tr>
<tr>
<td>NORM</td>
<td>6-74</td>
<td></td>
<td>Sets Trigger to Normal Sweep.</td>
</tr>
<tr>
<td>ONE</td>
<td>6-75</td>
<td></td>
<td>Sets Trigger to One Shot. Causes oscilloscope to sweep once when trigger is received and after setting arm function (SCOPe:ARM command).</td>
</tr>
<tr>
<td>SOURCE type</td>
<td></td>
<td>6-75</td>
<td>Selects source to trigger on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VERTical n</td>
<td>6-75</td>
<td></td>
<td>Sets Vertical Offset voltage to n graticules. Setting is not linear. For AC, DC or GND, middle setting is approximately 150 to 170. For other inputs, middle setting is approximately 160 to 180.</td>
</tr>
<tr>
<td>SCREEN COMMANDS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCREEN:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AF</td>
<td>6-2</td>
<td></td>
<td>Displays Audio Frequency Meter Operation Screen.</td>
</tr>
<tr>
<td>AFLVL</td>
<td>6-2</td>
<td></td>
<td>Displays AF Level (RMS) Meter Operation Screen.</td>
</tr>
<tr>
<td>ANL2</td>
<td>6-2</td>
<td></td>
<td>Displays Spectrum Analyzer Operation Screen.</td>
</tr>
<tr>
<td>BER</td>
<td>6-2</td>
<td></td>
<td>Displays Bit Error Rate Meter Operation Screen.</td>
</tr>
<tr>
<td>BUILD</td>
<td>6-2</td>
<td></td>
<td>Displays blank blue screen with green border.</td>
</tr>
<tr>
<td>CELL</td>
<td>6-2</td>
<td></td>
<td>Displays FOCC Forward Control Channel Screen</td>
</tr>
<tr>
<td>DEViation</td>
<td>6-2</td>
<td></td>
<td>Displays Deviation Meter (Peak) Operation Screen.</td>
</tr>
<tr>
<td>DISTortion</td>
<td>6-3</td>
<td></td>
<td>Displays Distortion Meter Operation Screen.</td>
</tr>
<tr>
<td>DMM</td>
<td>6-3</td>
<td></td>
<td>Displays Digital Multimeter Operation Screen.</td>
</tr>
<tr>
<td>DRMS</td>
<td>6-3</td>
<td></td>
<td>Displays Deviation Meter (RMS) Operation Screen.</td>
</tr>
<tr>
<td>DUPlex</td>
<td>6-3</td>
<td></td>
<td>Displays Duplex Operation Screen.</td>
</tr>
<tr>
<td>DUPRX</td>
<td>6-3</td>
<td></td>
<td>Displays Duplex Receiver Operation Screen.</td>
</tr>
<tr>
<td>DUFTX</td>
<td>6-3</td>
<td></td>
<td>Displays Duplex Transmitter Operation Screen.</td>
</tr>
<tr>
<td>FREquency</td>
<td>6-3</td>
<td></td>
<td>Displays Frequency Error Meter Operation Screen.</td>
</tr>
<tr>
<td>FUNC</td>
<td>6-3</td>
<td></td>
<td>Displays AF Generator Operation Screen.</td>
</tr>
<tr>
<td>GENCELLSETUP</td>
<td></td>
<td></td>
<td>Displays Cellular Settings Menu of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENCELLular</td>
<td></td>
<td>6-3</td>
<td>Displays Main Menu of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENerator</td>
<td>6-3</td>
<td></td>
<td>Displays RF Generator Operation Screen.</td>
</tr>
<tr>
<td>GENFOCC</td>
<td>6-3</td>
<td></td>
<td>Displays Forward Control Channel Screen of the AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENFVC</td>
<td>6-3</td>
<td></td>
<td>Displays the Forward Voice Channel of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENFVCNAMPS</td>
<td></td>
<td>6-4</td>
<td>Displays NAMPS FVC Screen of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENGGLACT</td>
<td>6-4</td>
<td></td>
<td>Displays Global Action Screen for FOCC of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENMSCM</td>
<td>6-4</td>
<td></td>
<td>Displays the Mobile Station Control Message for FOCC of AMPS/NAMPS Cellular Simulation.</td>
</tr>
</tbody>
</table>
### COMMANDS

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>RANGE</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCREEN:</td>
<td>6-4</td>
<td></td>
<td>Displays Reverse Control Channel Simulation Screen of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENRECC</td>
<td>6-124</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GENRVC</td>
<td>6-4</td>
<td></td>
<td>Displays Reverse Voice Channel Simulation Screen of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>GENRVCNAMPS</td>
<td>6-124</td>
<td></td>
<td>Displays NAMPS RVC Screen of AMPS/NAMPS Cellular Simulation.</td>
</tr>
<tr>
<td>MODulation</td>
<td>6-4</td>
<td></td>
<td>Displays Modulation Meter Operation Screen.</td>
</tr>
<tr>
<td>PM</td>
<td>6-4</td>
<td></td>
<td>Displays Phase Meter Operation Screen.</td>
</tr>
<tr>
<td>PMRMS</td>
<td>6-4</td>
<td></td>
<td>Displays Phase Meter (RMS) Operation Screen.</td>
</tr>
<tr>
<td>POWER</td>
<td>6-4</td>
<td></td>
<td>Displays Power Meter Operation Screen.</td>
</tr>
<tr>
<td>RECeiver</td>
<td>6-4</td>
<td></td>
<td>Displays Receiver Operation Screen.</td>
</tr>
<tr>
<td>SCOPe</td>
<td>6-4</td>
<td></td>
<td>Displays Oscilloscope Operation Screen.</td>
</tr>
<tr>
<td>Siganl</td>
<td>6-5</td>
<td></td>
<td>Displays Signal Strength Meter Operation Screen.</td>
</tr>
<tr>
<td>SINAD</td>
<td>6-5</td>
<td></td>
<td>Displays SINAD Meter Operation Screen.</td>
</tr>
<tr>
<td>USER</td>
<td>6-5</td>
<td></td>
<td>Displays blank User Screen without changing routing.</td>
</tr>
</tbody>
</table>

### SETUP COMMANDS

<table>
<thead>
<tr>
<th>SETUP:</th>
<th>PAGE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF</td>
<td>6-1</td>
<td>Configures Test Set routing for AF Meter Operation.</td>
</tr>
<tr>
<td>AFLVL</td>
<td>6-1</td>
<td>Configures Test Set routing for AF Level (RMS) Meter Operation.</td>
</tr>
<tr>
<td>ANLZ</td>
<td>6-1</td>
<td>Configures Test Set routing for Spectrum Analyzer Operation.</td>
</tr>
<tr>
<td>DISTortion</td>
<td>6-1</td>
<td>Configures Test Set routing for Distortion Meter Operation.</td>
</tr>
<tr>
<td>DUPlex</td>
<td>6-1</td>
<td>Configures Test Set routing for Duplex Operation.</td>
</tr>
<tr>
<td>DUPRX</td>
<td>6-1</td>
<td>Configures Test Set routing for Duplex Receiver Operation.</td>
</tr>
<tr>
<td>DUPTX</td>
<td>6-1</td>
<td>Configures Test Set routing for Duplex Transmitter Operation.</td>
</tr>
<tr>
<td>FUNC</td>
<td>6-1</td>
<td>Configures Test Set routing for AF Generator Operation.</td>
</tr>
<tr>
<td>GENerator</td>
<td>6-1</td>
<td>Configures Test Set routing for RF Generator Operation.</td>
</tr>
<tr>
<td>MOnitor</td>
<td>6-2</td>
<td>Configures Test Set routing for Generator/Monitor Operation.</td>
</tr>
<tr>
<td>RECeiver</td>
<td>6-2</td>
<td>Configures Test Set routing for Receiver Operation.</td>
</tr>
<tr>
<td>SCOPe</td>
<td>6-2</td>
<td>Configures Test Set routing for Oscilloscope Operation.</td>
</tr>
<tr>
<td>SINAD</td>
<td>6-2</td>
<td>Configures Test Set routing for SINAD Meter Operation.</td>
</tr>
</tbody>
</table>

### SIGNAL STRENGTH METER COMMANDS

See M_SIG:

### SINAD METER COMMANDS

See M_SINAD:

### SPECTRUM ANALYZER) COMMANDS

See Analyzer Commands.
SECTION 6 - HOST SPECIFIC TMAC COMMANDS

6-1 GENERAL

This section lists HOST specific commands by Operation Mode. Many commands operate only when the Test Set routing is configured for the operation mode of that command. SETUP commands establish routings for selected operation modes (see 6-2). SCREEN commands establish routings and display the operation screen for the selected operation mode (see 6-3).

Commands affect the Test Set when executed; however changes appear on the Operation Screens only when the Screen is renewed using a SCREEN command.

The short form of the command is shown in uppercase letters and the long form is finished in lower case. When entering commands, it is not necessary to use a particular letter case. The TMAC compiler/interpreter is not case sensitive. Brackets ([ ]) indicate optional command items.

6-2 SETUP COMMANDS

SETUP commands establish the routings for various Operation Modes without displaying the Operation Screens.

SETUP:

AF
[SETUP:AF]
Configures Test Set routing for AF Meter Operation.

AFLVL
[SETUP:AFLVL]
Configures Test Set routing for AF Level (RMS) Meter Operation.

ANLZ
[SETUP:ANLZ]
Configures Test Set routing for Spectrum Analyzer Operation.

DISTortion
[SETUP:DISTortion]
Configures Test Set routing for Distortion Meter Operation.

DUPlex
[SETUP:DUPlex]
Configures Test Set routing for Duplex Operation.

DUPRX
[SETUP:DUPRX]
Configures Test Set routing for Duplex Receiver Operation.

DUPTX
[SETUP:DUPTX]
Configures Test Set routing for Duplex Transmitter Operation.

FUNC
[SETUP:FUNC]
Configures Test Set routing for AF Generator Operation.

GENerator
[SETUP:GENerator]
Configures Test Set routing for RF Generator Operation.
SETUP:

MONitor
[SETUP:MONitor]
Configures Test Set routing for Generator/Monitor Operation.

RECeiver
[SETUP:RECiever]
Configures Test Set routing for Receiver Operation.

SCOPe
[SETUP:SCOPe]
Configures Test Set routing for Oscilloscope Operation.

SINAD
[SETUP:SINAD]
Configures Test Set routing for SINAD Meter Operation.

6-3 SCREEN COMMANDS

SCREEN commands renew and display Mode Operation Screen. Many commands operate correctly only when the applicable Mode Operation Screen is displayed.

SCREEN:

AF
[SCREEN:AF]
Displays Audio Frequency Meter Operation Screen.

AFLVL
[SCREEN:AFLVL]
Displays AF Level (RMS) Meter Operation Screen.

ANLZ
[SCREEN:ANLZ]
Displays Spectrum Analyzer Operation Screen.

BER
[SCREEN:BER]
Displays Bit Error Rate Meter Operation Screen.

BUILD
[SCREEN:BUILD]
Displays blank blue screen with green border.

CELL
[SCREEN:CELL]
Displays Forward Control Channel Screen of the AMPS/NAMPS Cell Site Monitor.

DEViation
[SCREEN:DEViation]
Displays Deviation Meter (Peak) Operation Screen.
SCREEN:

DISTortion
[SCREEN:DISTortion]
Displays Distortion Meter Operation Screen.

DMM
[SCREEN:DMM]
Displays Digital Multimeter Operation Screen.

DRMS
[SCREEN:DRMS]
Displays Deviation Meter (RMS) Operation Screen.

DUPlex
[SCREEN:DUPlex]
Displays Duplex Operation Screen.

DUPRX
[SCREEN:DUPRX]
Displays Duplex Receiver Operation Screen.

DUPTX
[SCREEN:DUPTX]
Displays Duplex Transmitter Operation Screen.

FREQuency
[SCREEN:FREQuency]
Displays Frequency Error Meter Operation Screen.

FUNC
[SCREEN:FUNC]
Displays AF Generator Operation Screen.

GENCELSETUP
[SCREEN:GENCELSETUP]
Displays the Cellular Settings Menu of the AMPS/NAMPS Cellular Simulation.

GENCELLular
[SCREEN:GENCELLular]
Displays Main Menu of the AMPS/NAMPS Cellular Simulation.

GENerator
[SCREEN:GENerator]
Displays RF Generator Operation Screen.

GENFOCC
[SCREEN:GENFOCC]
Displays the Forward Control Channel of the AMPS/NAMPS Cellular Simulation.

GENFVC
[SCREEN:GENFVC]
Displays the Forward Voice Channel of the AMPS/NAMPS Cellular Simulation.
SCREEN:

GENFVCNAMPS
[SCREEN:GENFVCNAMPS]
Displays the Narrow (NAMPS) Forward Voice Channel Screen of the AMPS/NAMPS Cellular Simulation.

GENGLACT
[SCREEN:GENGLACT]
Displays the Global Action Screen for the Forward Control Channel (FOCC) of the AMPS/NAMPS Cellular Simulation.

GENMSCM
[SCREEN:GENMSCM]
Displays the Mobile Station Control Message for the FOCC of the AMPS/NAMPS Cellular Simulation.

GENRECC
[SCREEN:GENRECC]
Displays Reverse Control Channel Simulation Screen of AMPS/NAMPS Cellular Simulation.

GENRVC
[SCREEN:GENRVC]
Displays Reverse Voice Channel Simulation Screen of the AMPS/NAMPS Cellular Simulation.

GENRVCNAMPS
[SCREEN:GENRVCNAMPS]
Displays the Narrow (NAMPS) Reverse Voice Channel Screen of the AMPS/NAMPS Cellular Simulation.

MODulation
[SCREEN:MODulation]
Displays Modulation Meter Operation Screen.

PM
[SCREEN:PM]
Displays Phase Meter Operation Screen.

PMRMS
[SCREEN:PMRMS]
Displays Phase Meter (RMS) Operation Screen.

POWer
[SCREEN:POWer]
Displays Power Meter Operation Screen.

RECeiver
[SCREEN:RECeiver]
Displays Receiver Operation Screen.

SCOPe
[SCREEN:SCOPe]
Displays Oscilloscope Operation Screen.
SCREEN:

SIGNAL
[SCREEN:SIGNAL]
Displays Signal Strength Meter Operation Screen.

SINAD
[SCREEN:SINAD]
Displays SINAD Meter Operation Screen.

USER
[SCREEN:USER]
Displays blank User Screen without changing routing. User Screen has no readings to update, decreasing run time when inside a macro. Following this command with other graphic, window and keypad commands creates customized menu screens (see 2-13).

6-4 INSTRUMENT COMMAND

The Instrument command allows the user to select a default "instrument" type (e.g., RF Generator, Receiver, Duplex, etc.). This command allows the user to issue TMAC commands without the "instrument" part (first level) of the command.

**INSTRument:SELeect type**
[INSTRument:SELeect type]
Selects default instrument; eliminates the necessity of the first part of the TMAC command for succeeding commands to that instrument. Enter one of the following for type:

<table>
<thead>
<tr>
<th>GENerator</th>
<th>RECEiver</th>
<th>DUPlex</th>
<th>FGEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOPe</td>
<td>ANLZ</td>
<td>M_AF</td>
<td>M_RF</td>
</tr>
<tr>
<td>M_PWR</td>
<td>M_DEV</td>
<td>M_MOD</td>
<td>M_DIST</td>
</tr>
<tr>
<td>M_SINAD</td>
<td>M_SIG</td>
<td>M_BER</td>
<td>M_DMM</td>
</tr>
<tr>
<td>M_PM</td>
<td>M_DRMS</td>
<td></td>
<td>M_PMRMS</td>
</tr>
</tbody>
</table>

Example:

```plaintext
INST:REC   // Selects Receiver as Default Instrument.
REC:FREQ?  // Returns Receiver freq. setting.
FREQ?      // Also, returns Receiver freq. setting.
GEN:DIST   // This command is unaffected, performs normally.
FREQ?      // Still returns Receiver freq. setting.
INST:GEN   // Selects RF Generator as default.
FREQ?      // Returns RF Generator freq. setting.
```
6-5 RF GENERATOR

6-5-1 RF GENERATOR COMMANDS

GENerator:

**AF**
[GENerator:AF]
Displays AF Level Meter on the RF Generator Operation Screen when followed by SCREEN:GENerator command.

**CHANnel n**
[GENerator:CHANnel n]
Sets Frequency to cellular channel n (1 to 2047, depending on format) in the format selected using the GEN:CHAN:FORM command. RF Generator Operation Screen displays selected cellular channel when RF Generator is in Channel Mode (GEN:MODE CHAN command).

CHANnel:

**FORMat:**

**AMPS:**

**FORward**
[GENerator:CHANnel:FORMat:AMPS:FORward]
Selects AMPS (NADC-U8) Forward channels.

**REVerse**
[GENerator:CHANnel:FORMat:AMPS:REVerse]
Selects AMPS (NADC-U8) Reverse channels.

**ETACS:**

**FORward**
[GENerator:CHANnel:FORMat:ETACS:FORward]
Selects ETACS Forward channels.

**REVerse**
[GENerator:CHANnel:FORMat:ETACS:REVerse]
Selects ETACS Reverse channels.

**NAMPS:**

**FORward**
[GENerator:CHANnel:FORMat:NAMPS:FORward]
Selects NAMPS Forward channels.

**REVerse**
[GENerator:CHANnel:FORMat:NAMPS:REVerse]
Selects NAMPS Reverse channels.

**BAND:**
x
[GENerator:CHANnel:FORMat:NAMPS:BAND:x]
Selects NAMPS Narrow Analog channel designator. Range of x is Lower, Middle or Upper.
GENerator:

CHANnel:

FORMat:

NT400:

FORward
 [GENerator:CHANnel:FORMat:NT400:FORward]
Selects NT400 Forward channels.

REVerse
[GENerator:CHANnel:FORMat:NT400:REVerse]
Selects NT400 Reverse channels.

NADC:

FORward
 [GENerator:CHANnel:FORMat:NADC:FORward]
Selects NADC Forward. Utilizes current setting of NADC band.

REVerse
[GENerator:CHANnel:FORMat:NADC:REVerse]
Selects NADC Reverse. Utilizes current setting of NADC band.

BAND:x
[GENerator:CHANnel:FORMat:NADC:BAND:x]
Selects NADC band. Range of x is U8, U4 or HYper.

FORMat?
[GENerator:CHANnel:FORMat?]  

BAND?
[GENerator:CHANnel:BAND?]  
Returns band for Generator Channel Format. Returns one of the following bands for the associated channel format:

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>BAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC</td>
<td>U8, U4 or HY</td>
</tr>
<tr>
<td>ETACS</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>NAMPS</td>
<td>LOWER, MIDDLE or UPPER</td>
</tr>
</tbody>
</table>
GENerator:

DCS:

INVert nnn
[GENerator:DCS:INVert nnn]
Generates the three digit octal Digital Coded Squelch (DCS) code in inverted mode. Range of nnn is 000 to 777.

NORMAL nnn
[GENerator:DCS:NORMal nnn]
Generates the three digit octal DCS code. Range of nnn is 000 to 777. See 6-5-2G.

STOP
[GENerator:DCS:STOP]
Stops generating the continuous DCS code.
DELY 200 // Waits for 200 ms before next command.
GEN:DCS:STOP // Stops generating the DCS code.

DIAL "sequence"
[GENerator:DIAL "sequence"]
Generates sequence with a maximum of 16 digits (0 through 9 allowed) as 2805 Pulse code. See 6-5-21.

DIAL:

FREQuency f
[GENerator:DIAL:FREQuency f]
Sets 2805 Tone frequency to f Hz. Range of f is 0.0 to 40000.0.

FREQuency?
[GENerator:DIAL:FREQuency?]
Returns the current 2805 Tone frequency in Hz.
Example: GEN:DIAL:FREQ 3000 // Sets 2805 Tone frequency to 3000 Hz.
GEN:DIAL:FREQ? // Returns 2805 Tone frequency setting.
GEN:DIAL "5552424" // Generates a 2805 Tone frequency message // of 5552424.

GENerator:

DISTortion
[GENerator:DISTortion]
Displays Distortion Meter on the RF Generator Operation Screen after the screen is updated by a SCREEN:GENerator command.

DMM
[GENerator:DMM]
Displays Digital Multimeter on the RF Generator Operation Screen after the screen is updated by a SCREEN:GENerator command.
GENerator:

DSAT n
[GENerator:DSAT n]
Generates the Digital Supervisory Audio Tone (DSAT) code specified by n. Range of n is 0 to 6.

DSAT:STOP
[GENerator:DSAT:STOP]
Stops generating the continuous DSAT code.

DST n
[GENerator:DST n]
Generates the Digital Signal Tone (DST) code specified by n. Range of n is 0 to 6.

DST:STOP
[GENerator:DST:STOP]
Stops generating the continuous DST code.

DTMF "sequence",mark,space
[GENerator:DTMF "sequence",mark,space]
Generates a DTMF coded sequence. sequence has a maximum of 16 digits. Range of optional mark and space time is from 25 to 9999 ms. Default for mark time is 74 ms. Default for space time is 67 ms. See 6-5-2D.

FREQuency n [units]
[GENerator:FREQuency n [units]]
Sets RF Generator Frequency. Range of n is 250.0 kHz to 2010.0000 MHz. Select HZ, KHZ, MHZ for units. Default for optional units is KHZ.

FREQuency?
[GENerator:FREQuency?]  
Returns RF Generator Frequency in kHz (250.0 to 2010000.0).
Example: 
GEN:FREQ 100 MHz  // Sets RF Generator Frequency to 100 MHz.
GEN:FREQ?          // Queries the RF Generator Frequency.
// 100000 is returned (100000 kHz).

IMTS "sequence"
[GENerator:IMTS "sequence"]
Generates sequence as a DCS IMTS code. sequence has a maximum of 16 digits (0 through 9 allowed).

LEVel n [units]
[GENerator:LEVel n [units]]
Sets RF Generator Level. Range of n is -137.0 to 0.0 dBm or 0.031 µV to 0.224 V. Select DBm, V, MV (mV) or UV (µV) for units. units is optional with the default being the current units. Specifying units does not change unit selection of Test Set.
GENerator:

LEVel:DBm n
[GENerator:LEVel:DBm n]
Sets RF Generator Level in dBm. Range of n is -137.0 to 0.0.

LEVel:DBm?
[GENerator:LEVel:DBm?]  
Returns RF Generator Level in dBm.

LEVel:UNIT
[GENerator:LEVel:UNIT]
Toggles the RF Generator level units between dBm and Volts.

LEVel:UNIT?
[GENerator:LEVel:UNIT?]  
Returns the current units for the RF Generator Level. Returns DBM or V.

LEVel?
[GENerator:LEVel?]  
Returns the RF Generator Level in the current units.

Example:  
GEN:LEV -65 DB // Sets RF Generator Level to -65 dBm.  
GEN:LEV:UNIT? // Queries the units for the RF Generator  
// Level. DBM is returned.  
GEN:LEV? // Queries the RF Generator Level. -65  
// is returned.  
GEN:LEV -30 // Sets RF Generator Level to -30 dBm (units  
// default to dBm, the current units).

MODE type
[GENerator:MODE type]
Selects the Generator Mode. Channel Mode displays cellular channel frequency according to  
GEN:CHAN commands.

<table>
<thead>
<tr>
<th>MODE</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>DIRect</td>
</tr>
<tr>
<td>Channel</td>
<td>CHANNEL</td>
</tr>
<tr>
<td>Frequency Scan</td>
<td>SCAN</td>
</tr>
<tr>
<td>Frequency List</td>
<td>LIST</td>
</tr>
<tr>
<td>Frequency List Scan</td>
<td>FLScan</td>
</tr>
</tbody>
</table>

MODE?
[GENerator:MODE?]  
Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or  
FREQUENCY LIST SCAN).

MTS "sequence"
[GENerator:MTS "sequence"]  
Generates sequence as a DCS MTS code. sequence has a maximum of 16 digits  
(0 through 9 allowed).
GENerator:

OUTput:

AUDio b
[GENerator:OUTput:AUDio b]
Routes AF Generator Output to the AUDIO OUT Connector if b is 1. Disconnects the
AUDIO OUT Connector if b is 0.

DEMOD b
[GENerator:OUTput:DEMOD b]
Routes AF Generator Output to the DEMOD OUT Connector if b is 1. Disconnects the
DEMOD OUT Connector if b is 0.

POCSAG:

ALPHA:

LOWer capcode
[GENerator:POCSAG:ALPHA:LOWer capcode]
Generates a lower case Alpha message for the capcode specified. Range of the
capcode is 0 to 2097151.

NUMeric capcode
[GENerator:POCSAG:ALPHA:NUMeric capcode]
Generates an Alphanumeric message for the capcode specified. Range of the capcode
is 0 to 2097151.
Example:
GEN:POCSAG:NUMeric 4000 // Generates an Alphanumeric message at a
// capcode of 4000 at the current rate.

SPECial capcode
[GENerator:POCSAG:ALPHA:SPECial capcode]
Generates an Alpha special message for the capcode specified. Range of the capcode
is 0 to 2097151.

UPPer capcode
[GENerator:POCSAG:ALPHA:UPPer capcode]
Generates an upper case Alpha message for the capcode specified. Range of the
capcode is 0 to 2097151. See 6-5-2H.

BEEP n, capcode
[GENerator:POCSAG:BEEP n, capcode]
Generates n Tone Beep POCSAG message for capcode specified. Select 1, 2, 3 or 4
for n. Range of the capcode is 0 to 2097151.

NUMeric capcode
[GENerator:POCSAG:NUMeric capcode]
Generates Numeric message for the capcode specified. Range of capcode is 0 to
2097151.
GENerator:

POCSAG:

RATe

[GENerator:POCSAG:RATe]
Sets POCSAG rate to high if b is 1, low if b is 0.

RATe?

[GENerator:POCSAG:RATe?]
Returns 1 if POCSAG rate is high; returns 0 if rate is low.

Example:

*DMC "POCSAG",BEGIN // Defines a macro named POCSAG.
GEN:POCSAG:RATE 1 // Sets POCSAG rate to High.
FOR X=1 TO 5 // Starts FOR loop of X to loop 5 times.
GEN:POCSAG:BEEP 2,X+5030 // Generates a POCSAG Tone - 2 beeps
"message" // using the capcode specified by X.
DELAY 3000 // Delays 3 seconds (for POCSAG generation).
*WAI // Wait for the last command to be executed.
NEXT X // Loops X to the top of the loop until X=5.
GEN:POCSAG:RATE? // Queries the POCSAG rate. 1 is returned
END // Ends the macro named POCSAG.

RCL n

[GENerator:RCL n]
Recalls RF Generator environment (routings and settings) stored in memory location n.
Range of n is 1 to 9.

SCAN:

Except where noted, the following commands are for RF Generator Frequency Scan or Frequency List Scan modes.

ABORT

[GENerator:SCAN:ABORT]
Stops frequency scan or frequency list scan depending on present mode.

CONTinue

[GENerator:SCAN:CONTinue]
Starts frequency scan or frequency list scan depending on present mode.

FREQuency?

[GENerator:SCAN:FREQuency?]
Returns current frequency in kHz being scanned.

INCrement n

[GENerator:SCAN:INCrement n]
(Frequency Scan mode only) Specifies increment in kHz between frequencies to be scanned. Range of n is 0.0 to 201000.0.
GENerator:

SCAN:

PAUse?
[GENerator:SCAN:PAUse?]
(Frequency Scan mode only) Returns a 1 if scanning is paused (stopped) or 0 if currently scanning.

RATE n
[GENerator:SCAN:RATE n]
(Frequency Scan mode only) Scan Rate. Specifies time period in seconds for RF Generator to sit on a frequency unless squelch is broken. Range of n is 0.02 to 99.99.

START f
[GENerator:SCAN:START f]
(Frequency Scan mode only) Starting Frequency value in kHz. Specifies lower limit frequency for scanning. Range of f is 250.0 to 2010000.0.

STOP f
[GENerator:SCAN:STOP f]
(Frequency Scan mode only) Stop Frequency value in kHz. Specifies upper limit frequency for scanning. Range of f is 250.0 to 2010000.0.

FREQList:
The following commands are for RF Generator Frequency List Scan mode only.

RATE n
[GENerator:SCAN:FREQList:RATE n]
Frequency List Scan Rate. Specifies time period in seconds for RF Generator to sit on a frequency. Range of n is 0.02 to 99.99.

RATE?
[GENerator:SCAN:FREQList:RATE?]
Returns current value of Frequency List Scan Rate.

SINAD
[GENerator:SINAD]
Displays SINAD Meter on the RF Generator Operation Screen when followed by a SCREEN:GENerator command.

SPEAKer:SOURce type
[GENerator:SPEAKer:SOURce type]
Selects Test Set Speaker Input. Select OFF, FGEN (Function Generator), SINAD (SINAD/BER IN Connector) or EXTMOD (EXT MOD IN Connector) for type. This command takes effect when the RF Generator Operation Screen is updated.

Example: GEN:SPEAK:SOUR EXTMOD // Routes EXT MOD IN Connector to the Test Set Speaker.
GENerator:

STORE n
[GENerator:STORE n]
Stores current RF Generator environment (routings and settings) in memory location n.
Range of n is 1 to 9.

TONE "sequence"
[GENerator:TON E "sequence"]
Generates given sequence once the Audio code is selected using the GENerator:TO NE:TYPE command. If selected Audio code is USER, characters contained in the sequence must be previously defined using the GENerator:TON E:USER:DEF ine command. Valid sequence characters are the digits 0 through 9 and characters A, G, R and - (to signify a gap). Valid sequence characters for the USER code are the digits 0 through 9 and characters A through T.
Example: GEN:TO NE:TYPE CCIRH // Selects CCIRH as RF Generator Audio Code.
GEN:TO NE "5553434" // Generates 5552424 as CCIRH Audio Signal.

TONE:

TYPE code
[GENerator:TON E:TYPE code]
Selects the Audio code to generate. Enter one of the following for code:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCIR</td>
<td>EEA</td>
<td>EIA</td>
<td>(U.S.)</td>
</tr>
<tr>
<td>ZVEI</td>
<td>DDZVEI</td>
<td>DZVEI</td>
<td></td>
</tr>
<tr>
<td>NATEL</td>
<td>EURO</td>
<td>TONE56</td>
<td></td>
</tr>
<tr>
<td>CCIRH</td>
<td>CCIRH4</td>
<td>USER</td>
<td></td>
</tr>
</tbody>
</table>

See 6-5-2E and 6-5-2F.

USER:DEF ine "id",freq,duration
[GENerator:TON E:USER:DEF ine "id",freq,duration]
The id character is assigned a freq in Hz and a duration in ms. Range of freq is 0.0 to 9999.9. Range of duration is 20.0 to 9999.9. Valid characters for the id are the digits 0 through 9 and characters A through T. See 6-5-2E and 6-5-2F.

TREMote f
[GENerator:TRE Mote f]
Generates the Tone-remote sequence for the specified function tone frequency given by f in Hz. Select one of the following:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2050</td>
<td></td>
</tr>
<tr>
<td>1750</td>
<td></td>
</tr>
<tr>
<td>1450</td>
<td></td>
</tr>
<tr>
<td>1150</td>
<td></td>
</tr>
<tr>
<td>1950</td>
<td></td>
</tr>
<tr>
<td>1650</td>
<td></td>
</tr>
<tr>
<td>1350</td>
<td></td>
</tr>
<tr>
<td>1050</td>
<td></td>
</tr>
<tr>
<td>1850</td>
<td></td>
</tr>
<tr>
<td>1550</td>
<td></td>
</tr>
<tr>
<td>1250</td>
<td></td>
</tr>
</tbody>
</table>

TRE Mote:STOP
[GENerator:TREMote:STOP]
Stops the Tone remote Guard Tone generated by a previous GENerator:TREMote command.
Example: GEN:TREM 1350 // Generates a Tone Remote 1350 message.
GEN:TREM:STOP // Stops the generating of the Tone Remote // Guard Tone.
6-5-2  REMOTE RF GENERATOR EXAMPLES

A. GENERATING FM MODULATED RF SIGNALS

The following command sequence generates a 1 MHz RF signal FM modulated (5 kHz deviation) with a 1 kHz sine wave and routes the signal to the DEMOD OUT Connector:

SCREEN:GEN // Displays RF Generator Operation Screen.
GEN:FREQ 1000000 // Sets RF Generator Frequency to 1 MHz.
GEN:LEVEL 0 DB // Sets the RF Generator Level to 0 dBm.
GEN:OUTPUT:DEMOD 1 // Routes the RF Generator output to the DEMOD OUT Connector.
FGEN:GEN1:STATE 1 // Activates SOURCE 1 (AF Generator 1).
FGEN:GEN1:FREQ 1000 // Sets SOURCE 1 Frequency to 1000 Hz.
FGEN:GEN1:MODULATION:FM // Sets SOURCE 1 Modulation to FM.
FGEN:GEN1:MODL 5 // Sets SOURCE 1 Deviation to 5 kHz.
FGEN:GEN1:SHAPE:SIN // Sets SOURCE 1 Wave Shape to a sine wave.

B. GENERATING AM MODULATED RF SIGNALS

The following command sequence generates a 10 MHz RF signal, AM modulated (80% modulation), with a 2.5 kHz sine wave and routes the signal to the AUDIO OUT Connector:

SCREEN:GEN // Displays RF Generator Operation Screen.
GEN:FREQ 10 MHZ // Sets RF Generator Frequency to 10 MHz.
GEN:LEVEL 0 DB // Sets the RF Generator Level to 0.0 dBm.
GEN:OUTPUT:AUDIO 1 // Routes the RF Generator output to the AUDIO OUT Connector.
FGEN:GEN1:STATE 1 // Activates SOURCE 1 (AF Generator 1).
FGEN:GEN1:FREQ 2500 // Sets SOURCE 1 Frequency to 2500 Hz.
FGEN:GEN1:MODULATION:AM // Sets SOURCE 1 Modulation to AM.
FGEN:GEN1:MODL 80 // Sets SOURCE 1 Modulation to 80%.
FGEN:GEN1:SHAPE:SIN // Sets SOURCE 1 Wave Shape to a sine wave.

C. GENERATING EXTERNALLY MODULATED RF SIGNALS

The following command sequence generates a RF 15 MHz signal, AM modulated (80% modulation), with an external signal applied to the EXT MOD IN Connector and sends the signal to the AUDIO OUT Connector:

SCREEN:GEN // Displays RF Generator Operation Screen.
GEN:FREQ 15000 // Sets RF Generator Frequency to 15 MHz.
GEN:LEV -20 // Sets the RF Generator Level to -20 dBm.
GEN:OUT:AUD 1 // Routes the RF Generator Output to the AUDIO OUT Connector.
FGEN:EXT:STATE 1 // Activates SOURCE EXT (for external modulation).
FGEN:EXT:MOD:AM // Sets SOURCE EXT Modulation to AM.
FGEN:EXT:MODL 80 // Sets SOURCE EXT Modulation Level to 80%.
D. GENERATING A DTMF CODED SIGNAL

The following command sequence generates a 450 MHz signal FM modulated with a DTMF Code. The modulation level is 4 kHz and the RF Output Level is -60 dBm.

SCREEN:GEN  // Displays RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
GEN:FREQ 450 MHZ  // Sets RF Generator Frequency to 450 MHz.
GEN:LEV -60 DB  // Sets the RF Generator Level to -60 dBm.
FGEN:GEN3:ENCODE DTMF  // Selects DTMF as the signaling format.
SCREEN:GEN  // Renews RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
FGEN:GEN3:MOD:FM  // Sets SOURCE 3 Modulation to FM.
FGEN:GEN3:MODL 4  // Sets SOURCE 3 Modulation Level to 4 kHz.
GEN:DTMF "55523*#", 80, 70  // Generates 55523*# DTMF coded signal with
// 80 ms mark time and a 70 ms space time.

E. GENERATING AUDIO TWO TONE CODING

The following command sequence generates a 150 MHz signal FM modulated with an Audio Two Tone Code. The modulation level is 4 kHz and the RF Output Level is 0 dBm.

SCREEN:GEN  // Displays RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
GEN:FREQ 150 MHZ  // Sets RF Generator Frequency to 150 MHz.
GEN:LEV 0 DB  // Sets the RF Generator Level to 0 dBm.
FGEN:GEN3:ENCODE TONE  // Selects Audio as the signaling format.
SCREEN:GEN  // Renews RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
FGEN:GEN3:MOD:FM  // Sets SOURCE 3 Modulation to FM.
FGEN:GEN3:MODL 4  // Sets SOURCE 3 Modulation Level to 4 kHz.
GEN:TOUE:USER:DEF "0,0,100"  // Defines 0 Tone as 0 Hz, 100 ms.
GEN:TOUE:USER:DEF "1,880,500"  // Defines 1 Tone as 880 Hz, 500 ms.
GEN:TOUE:USER:DEF "2,2200,500"  // Defines 2 Tone as 2200 Hz, 500 ms.
GEN:TOUE 102  // Generates the User Defined Audio Code
// defined in the 3 previous commands.
F. GENERATING A 5/6 AUDIO TONE SEQUENCE

The following command sequence generates a 162 MHz signal FM modulated with an Audio 5/6 Tone Code. The modulation level is 4 kHz and the RF Output Level is 0 dBm.

SCREEN:GEN  // Displays RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
GEN:FREQ 162 MHZ  // Sets RF Generator Frequency to 162 MHz.
GEN:LEV 0 DB  // Sets the RF Generator Level to 0 dBm.
PGEN:GEN3:ENCODE TONE  // Selects Audio as the signaling format.
SCREEN:GEN  // Renews RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
PGEN:GEN3:MOD:FM  // Sets SOURCE 3 Modulation to FM.
PGEN:GEN3:MODL 4  // Sets SOURCE 3 Modulation Level to 4 kHz.
GEN:TONE:USER:DEF 0,900,150  // Defines 0 Tone at 900 Hz and 150 ms.
GEN:TONE:USER:DEF 1,1100,80  // Defines 1 Tone at 1100 Hz and 80 ms.
GEN:TONE:USER:DEF 2,1200,80  // Defines 2 Tone at 1200 Hz and 80 ms.
GEN:TONE:USER:DEF 3,1300,80  // Defines 3 Tone at 1300 Hz and 80 ms.
GEN:TONE:USER:DEF 4,1400,80  // Defines 4 Tone at 1400 Hz and 80 ms.
GEN:TONE:USER:DEF A,0,40  // Defines A Tone at 0 Hz and 40 ms.
GEN:TONE 0A1234  // Generates the User Defined Audio Code defined in the 6 previous commands.

G. GENERATING DCS CODE

The following command sequence generates a 162.450 MHz signal FM modulated with a DCS Code of 456. The modulation level is 1 kHz and the RF Output Level is 0 dBm.

SCREEN:GEN  // Displays RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
GEN:FREQ 162450  // Sets 162.450 MHz RF Generator Frequency.
GEN:LEV 0 DB  // Sets the RF Generator Level to 0 dBm.
PGEN:GEN3:ENCODE DIG  // Selects Digital as the signaling format.
SCREEN:GEN  // Renews RF Generator Operation Screen.
*WAI  // Waits for previous commands to execute.
PGEN:GEN3:MOD:FM  // Sets SOURCE 3 Modulation to FM.
PGEN:GEN3:MODL 1  // Sets SOURCE 3 Modulation Level to 1 kHz.
GEN:DCS:STOP  // Stops the generating of the DCS Code.
H. GENERATING POCSAG CODE

The following command sequence generates a 930 MHz signal FM modulated with a POCSAG Code. The modulation level is 4 kHz and the RF Output Level is 0 dBm.

```
SCREEN:GEN // Displays RF Generator Operation Screen.
*WAI // Waits for previous commands to execute.
GEN:FREQ 450 MHZ // Sets RF Generator Frequency to 450 MHz.
GEN:LEV 0 DB // Sets the RF Generator Level to 0 dBm.
FGEN:GEN3:ENCODE DIG // Selects Digital as the signaling format.
SCREEN:GEN // Renews RF Generator Operation Screen.
*WAI // Waits for previous commands to execute.
FGEN:GEN3:MOD:FM // Sets SOURCE 3 Modulation to FM.
FGEN:GEN3:MODL 4 // Sets SOURCE 3 Modulation Level to 4 kHz.
GEN:POCSAG:RATE 0 // Sets POCSAG rate to Low.
FOR Y=1 TO 5 // Starts loop. Y to be looped 5 times.
  GEN:POCSAG:ALPHA:UPPER Y+5130 // Generates an Alpha upper POCSAG
  // message using the specified capcode.
  DELAY 3000 // Delays 3 seconds (for POCSAG generation).
  *WAI // Waits for the last command to be executed.
NEXT Y // Loops to top of the loop until Y is 5.
```

I. GENERATING 2805 CODE

The following command sequence generates a 155 MHz signal FM modulated with a 2805 Tone (with frequency reset to 1500 Hz). The modulation level is 4 kHz and the RF Output Level is -60 dBm.

```
SCREEN:GEN // Displays RF Generator Operation Screen.
*WAI // Waits for previous commands to execute.
GEN:FREQ 155 MHZ // Sets RF Generator Frequency to 155 MHz.
GEN:LEV -60 DB // Sets the RF Generator Level to -60.0 dBm.
FGEN:GEN3:ENCODE RCC // Selects RCC as the signaling format.
SCREEN:GEN // Renews RF Generator Operation Screen.
*WAI // Waits for previous commands to execute.
FGEN:GEN3:MOD:FM // Sets SOURCE 3 Modulation to FM.
FGEN:GEN3:MODL 4 // Sets SOURCE 3 Modulation Level to 4 kHz.
GEN:DIAL:FREQ 1500 // Sets 2805 Tone Frequency to 1500 Hz.
GEN:DIAL 5551234 // Generates a 5551234 2805 Tone signal.
```
6-6 RECEIVER

6-6-1 RECEIVER COMMANDS

RECeiver:

AGC:

AUTO
[RECeiver:AGC:AUTO]
Sets Automatic Gain Control to automatic setting.

MANual n
[RECeiver:AGC:MANual n]
Sets Automatic Gain Control to manual setting and sets level to n. Range of n is 0 to 255.

Example: REC:AGC:MAN 130 // Sets AGC setting to Manual and sets
          // Manual AGC setting to 130.

USER:xxx
[RECeiver:AGC:USER:xxx]
Sets Automatic Gain Control to User. Select one of the following for xxx: MEASURE, SPEECH, DATA, HIGH, TYPE1, TYPE2 or TYPE3.


CHANnel n
[RECeiver:CHANnel n]
Sets Receive Frequency to cellular channel n (1 to 2047, depending on format) in the format selected using the REC:CHAN:FORMAT command. The Receiver Operation Screen displays selected cellular channel when Receiver is in Channel Mode (REC:MODE CHAN command).

CHANnel:

FORMAT:

AMPS:

FORward
[RECeiver:CHANnel:FORMAT:AMPS:FORward]
Selects AMPS (NADC-U8) Forward channels.

REVerse
[RECeiver:CHANnel:FORMAT:AMPS:REVerse]
Selects AMPS (NADC-U8) Reverse channels.

ETACS:

FORward
[RECeiver:CHANnel:FORMAT:ETACS:FORward]
Selects ETACS Forward channels.

REVerse
[RECeiver:CHANnel:FORMAT:ETACS:REVerse]
Selects ETACS Reverse channels.
RECeiver:

CHANnel:

FORMAT:

NAMPS:

FORward
[RECeiver:CHANnel:FORMAT:NAMPS:FORward]
Selects NAMPS Forward channels.

REVerse
[RECeiver:CHANnel:FORMAT:NAMPS:REVerse]
Selects NAMPS Reverse channels.

BAND:x
[RECeiver:CHANnel:FORMAT:NAMPS:BAND:x]
Selects NAMPS Narrow Analog channel designator. Range of x is Lower, Middle or Upper.

NT400:

FORward
[RECeiver:CHANnel:FORMAT:NT400:FORward]
Selects NT400 Forward channels.

REVerse
[RECeiver:CHANnel:FORMAT:NT400:REVerse]
Selects NT400 Reverse channels.

NADC:

FORward
[RECeiver:CHANnel:FORMAT:NADC:FORward]
Selects NADC Forward. Utilizes current setting of NADC band.

REVerse
[RECeiver:CHANnel:FORMAT:NADC:REverse]
Selects NADC Reverse. Utilizes current setting of NADC band.

BAND:x
[RECeiver:CHANnel:FORMAT:NADC:BAND:x]
Selects NADC band. Range of x is U8, U4 or HYper.

FORMAT?
RECeiver:

CHANnel:

BAND?
[RECeiver:CHANnel:BAND?]
Returns band for Receiver Channel Format. Returns one of the following bands for the associated channel format:

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>BAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC</td>
<td>U8, U4 or HY</td>
</tr>
<tr>
<td>ETACS</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>NAMPS</td>
<td>LOWER, MIDDLE or UPPER</td>
</tr>
</tbody>
</table>

DCS:

INVert?
[RECeiver:DCS:INVert?]
Returns the three octal digits in inverted mode, decoded from the Digital Coded Squelch (DCS) code. Returns -1 if none available or if invalid for inverted DCS.

NORMal?
[RECeiver:DCS:NORMal?]
Returns the three octal digits decoded from the DCS code. Returns -1 if none available or if invalid for normal DCS. See 6-6-2E.

STATe b
[RECeiver:DCS:STATe b]
Enables DCS decoding if $b$ is 1, disables if $b$ is 0. RECeiver:DECode DIGital and RECeiver:DIGital DCS (or DCSINV) commands must be initiated prior to enabling DCS decoding. DCS decoding must be disabled after decoding is finished.

Example:

```
RE:DEC DIG       // Sets Receiver for decoding digital data.
RE:DEC DCS       // Prepares Receiver for decoding DCS.
RE:DCS:STAT 1    // Enables DCS decoding.
RE:DCS:STAT 0    // Disables DCS decoding.
```

DECode type
[RECeiver:DECode type]
Sets Receiver for decoding type. The setting for type is DTMF, TONE or DIGital.

DEVrMs
[RECeiver:DEVrMs]
Displays Deviation Meter (RMS) reading on Receiver Operation Screen when followed by SCREEN:RECeiver command.

DIGital type
[RECeiver:DIGital type]
Sets digital type. Used with the RECeiver:DECode DIGital command to prepare the Receiver for decoding. The setting for type is DCS, DCSINV, POCSAG, DSAT or DST.
RECeiver:

DISTortion
(RECeiver:DISTortion)
Displays Distortion Meter reading on Receiver Operation Screen when followed by
SCREEN:RECeiver command.

DMM
(RECeiver:DMM)
Displays Digital Multimeter reading on Receiver Operation Screen when followed by
SCREEN:RECeiver command.

DSAT:STATE b
(RECeiver:DSAT:STATE b)
Enables DSAT Decoding Function if b is 1, disables if b is 0. RECeiver:DECode DIgital and
RECeiver:DIgital DSAT commands must be initiated prior to enabling DSAT decoding. DSAT
decoding must be disabled after decoding is finished.

DSAT?
(RECeiver:DSAT?)
Returns DSAT reading. Returns -1 if none available or invalid for normal transmission.

DST:STATE b
(RECeiver:DST:STATE b)
Enables DST Decoding Function if b is 1, disables if b is 0. RECeiver:DECode DIgital and
RECeiver:DIgital DST commands must be initiated prior to enabling DST decoding. DST
decoding must be disabled after decoding is finished.

DST?
(RECeiver:DST?)
Returns DST reading. Returns -1 if none available or invalid for normal transmission.

DTMF:STATE b
(RECeiver:DTMF:STATE b)
Enables DTMF Decoding Function if b is 1, disables if b is 0. See 6-6-2C.

DTMF?
(RECeiver:DTMF?)
Returns string of decoded digits or -1 if nothing decoded.

Example:  SCREEN:REC   // Displays Receiver Operation Screen.
          REC:DEC DTMF   // Sets Receiver for decoding DTMF signals.
          REC:DTMF:STAT 1 // Enables DTMF decoding.
          REC:DTMF:STAT 0 // Disables DTMF decoding.
RECeiver:

FIND:

FREQuency?
[RECeiver:FIND:FREQuency?]
Returns frequency of first signal with amplitude larger than Find reference level.
Returns 0 if no signal is found.

REFERENCE n
[RECeiver:FIND:REFERENCE n]
Sets Find reference level to n dBm. Range of n is -110 to -5.

REFERENCE?
[RECeiver:FIND:REFERENCE?]
Returns Find reference level in dBm (-110 to -5).

Example:
RECeiver:FIND:REF -65 // Sets Find Reference Level to -65 dBm.
RECeiver:FIND:FREQ? // Returns the first frequency (in kHz)
// containing a signal greater than -65 dBm.
RECeiver:FIND:REF? // Queries the current Find Reference Level
// (in dBm). Returns -65.

FMZ n
[RECeiver:FMZ n]
Zeros FM Deviation Meter. Displays the Receiver Operation Screen and selects FMn
Modulation. Range of n is from 1 to 4.

FREQuency n [units]
[RECeiver:FREQuency n [units]]
Sets Receiver Frequency. Range of n is 250.0 kHz to 2010.0000 MHz. Select HZ, KHZ or
MHZ for units. units is optional with a default of KHZ.

FREQuency?
[RECeiver:FREQuency?]
Returns Receiver Frequency in kHz (250.0 to 2010000.0).

Example:
RECeiver:FREQ 100 MHZ // Sets Receiver Frequency to 100 MHz.
RECeiver:FREQ? // Queries the Receiver Frequency.
// Returns 100000 (100000 kHz).
RECeiver:

INPut:

ANTenna
[RECeiver:INPut:ANTenna]
Selects ANTEenna IN Connector as the Receiver Input Source. Displays Signal Strength Meter reading on the Receiver Operation Screen.

The following Attenuation commands control the Input Attenuation for the Receiver and selection/deselection of LNA (Low Noise Amplifier). Use RECeiver:INPut:ATTenuation:LNA to select LNA and 0 dB of input attenuation. To deselect LNA and select a specific value of attenuation use RECeiver:INPut:ATTenuation n.

ATTenuation:LNA
[RECeiver:INPut:ATTenuation:LNA]
Sets Receiver Input Attenuation to LNA.

ATTenuation:LNA?
[RECeiver:INPut:ATTenuation:LNA?] Returns current state of the Low Noise Amplifier. Returns 1 if LNA is selected; 0 otherwise.

LNA (Low Noise Amplifier) has the same attenuation as 0 dB (see RECeiver:INPut:ATTenuation n), but LNA has a lower noise figure. LNA is the preferred option for doing off-the-air applications.

ATTenuation n
[RECeiver:INPut:ATTenuation n] Sets Receiver Input Attenuation to n dB. Possible values for n: 0, 5, 10, 15, 20, 25, 30.

ATTenuation?
[RECeiver:INPut:ATTenuation?] Returns the current value of Input Attenuation.

TR
RECeiver:

**MODE type**

(RECeiver:MODE type)

Selects the Receiver Mode. Channel Mode displays cellular channel frequency according to REC:CHAN commands.

<table>
<thead>
<tr>
<th>MODE</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>DIRECT</td>
</tr>
<tr>
<td>Channel</td>
<td>CHANNEL</td>
</tr>
<tr>
<td>Frequency Scan</td>
<td>SCAN</td>
</tr>
<tr>
<td>Frequency List</td>
<td>LIST</td>
</tr>
<tr>
<td>Frequency List Scan</td>
<td>FLScan</td>
</tr>
</tbody>
</table>

         REC:INP:ATT 20  // Sets Receiver Input Attenuation to 20 dB.
         REC:MODE CHAN  // Selects Receiver Channel Mode.

**MODE?**

(RECeiver:MODE?)

Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or FREQUENCY LIST SCAN).

**MODMeter**

(RECeiver:MODMeter)

Displays Modulation Meter reading on the Receiver Operation Screen when followed by a SCREEN:RECeiver command and AM is selected modulation.

**MODulation:**

AMn

(RECeiver:MODulation:AMn)

Selects Amplitude Modulation. Select 1 or 2 for n.

BFO

(RECeiver:MODulation:BFO)

Selects Beat Frequency Oscillation.

FMn

(RECeiver:MODulation:FMn)

Selects Frequency Modulation. Select 1, 2, 3 or 4 for n.

RECeiver:

MODulation:

PM
[RECeiver:MODulation:PM]
Selects Phase Modulation.

USER:

FILTER f
[RECeiver:MODulation:USER:FILTER f]
Sets User Defined Modulation IF Filter to f kHz. Select 3, 30 or 300 for f.

FILTER?
[RECeiver:MODulation:USER:FILTER?]
Returns the current setting of the User Defined Modulation IF Filter in kHz.

MODulation: type
[RECeiver:MODulation:USER:MODulation: type]
Selects User Defined Modulation type (FM, AM, BFO, PM or DATA [FM DATA]).

MODulation?
[RECeiver:MODulation:USER:MODulation?]
Returns User Defined Modulation type.

POST:

APASs
[RECeiver:MODulation:USER:POST:APASs]
Selects All Pass Post Detection Filter for User Defined Modulation.

BPASs fl, fh
[RECeiver:MODulation:USER:POST:BPASs fl, fh]
Selects Bandpass Post Detection Filter for the User Defined Modulation. Lower cutoff frequency is set to fl kHz with a range of 0.5 to 20. Higher cutoff frequency is set to fh kHz with a range of 0.1 to 30.

BPASs:

HIGH f
[RECeiver:MODulation:USER:POST:BPASs:HIGH f]
Sets Lower cutoff frequency of the Band-Pass Post Detection Filter in kHz. Range of f is 0.5 to 20.0.

HIGH?
[RECeiver:MODulation:USER:POST:BPASs:HIGH?]
Returns the Lower cutoff frequency of the Band-Pass Post Detection Filter in kHz.
RECeiver:

MODulation:

USER:

POST:

BPASs:

LOW f
[RECeiver:MODulation:USER:POST:BPASs:LOW f]
Sets Upper cutoff frequency of the Band-Pass Post Detection Filter in kHz.
Range of f is 0.1 to 30.0.

LOW?
[RECeiver:MODulation:USER:POST:BPASs:LOW?]
Returns the Upper cutoff frequency of the Band-Pass Post Detection Filter
in kHz.

CWEight
[RECeiver:MODulation:USER:POST:CWEight]
Selects C-Weighted Post Detection Filter (C-message noise weighting curve
response) for the User Defined Modulation. Refer to MIL-STD-188-200.

CMESsage
[RECeiver:MODulation:USER:POST:CMESsage]
Same as RECeiver:MODulation:USER:POST:CWEight command.

CWT
[RECeiver:MODulation:USER:POST:CWT]
Same as RECeiver:MODulation:USER:POST:CWEight command.

HPASs f/
[RECeiver:MODulation:USER:POST:HPASs f/]
Selects High-Pass Post Detection Filter for the User Defined Modulation. Cutoff
frequency is set to f/ kHz with a range of 0.5 to 20.

HPASs?
[RECeiver:MODulation:USER:POST:HPASs?]
Returns the cutoff frequency for the User Defined Modulation High-Pass Post
Detection Filter.
RECeiver:

MODulation:

USER:

POST:

LPASs \( fh \)

\([\text{RECeiver:MODulation:USER:POST:LPASs} \, fh]\)

Selects Low-Pass Post Detection Filter for the User Defined Modulation. Cutoff frequency is set to \( fh \) kHz with a range of 0.1 to 30.

Example:

\begin{verbatim}
REC:MOD:USER:POST:BPAS 10,20   // Selects a Bandpass Post Detection Filter with 10 kHz lower cutoff and 20 kHz higher cutoff frequencies.
REC:MOD?                 // Queries the Receiver Modulation. USER is returned.
\end{verbatim}

LPASs?

\([\text{RECeiver:MODulation:USER:POST:LPASs}?]\)

Returns the cutoff frequency for the User Defined Modulation Low-Pass Post Detection Filter.

POST?

\([\text{RECeiver:MODulation:USER:POST}?]\)

Returns the current setting of the User Defined Post Detection Filter. Returns one of the following: APAS, LPAS, HPAS, BPAS or CWE.

MODulation?

\([\text{RECeiver:MODulation}?]\)

Returns Receiver Modulation (FM, AM, BFO, PM, DATA or USER).

OUTput:

AUDio \( b \)

\([\text{RECeiver:OUTput:AUDio} \, b]\)

Routes Demodulated Audio to the AUDIO OUT Connector if \( b \) is 1. Disconnects the AUDIO OUT Connector if \( b \) is 0.

DEMOD \( b \)

\([\text{RECeiver:OUTput:DEMOD} \, b]\)

Routes Demodulated Audio to the DEMOD OUT Connector if \( b \) is 1. Disconnects the DEMOD OUT Connector if \( b \) is 0.

SPEAKer \( b \)

\([\text{RECeiver:OUTput:SPEAKer} \, b]\)

Routes Demodulated Audio to the Speaker if \( b \) is 1. Disconnects Speaker if \( b \) is 0.
RECeiver:

PMRms
(RECeiver:PMRms)
Displays Phase Meter (RMS) reading when followed by a SCREEN:RECeiver command.

POCSAG:

CAPcode?
(RECeiver:POCSAG:CAPcode?)
Returns the received capcode or -1 if a capcode is not received.

MESSage?
(RECeiver:POCSAG:MESSage?)
Returns message from the decoded POCSAG signal or -1 if none available.

RATe b
(RECeiver:POCSAG:RATe b)
Sets POCSAG rate to high if b is 1, low if b is 0.

RATe?
(RECeiver:POCSAG:RATe?)
Returns 1 if POCSAG rate is high, 0 if rate is low.

STATe b
(RECeiver:POCSAG:STATe b)
Enables POCSAG decoding if b is 1, disables if b is 0. RECeiver:DECode DIGital and RECeiver:DIGital POCSAG commands must be initiated prior to enabling POCSAG decoding. POCSAG decoding must be disabled after decoding is finished.

Changing screens with POCSAG enabled may cause Test Set to lock up.

TYPE?
(RECeiver:POCSAG:TYPE?)
Returns POCSAG Function Type. One of the following is returned:

<table>
<thead>
<tr>
<th>TONE 1 BEEP</th>
<th>TONE 2 BEEPS</th>
<th>TONE 3 BEEPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TONE 4 BEEPS</td>
<td>NUMERIC</td>
<td>ALPHANUMERIC</td>
</tr>
</tbody>
</table>

NO MESSAGE

Example:

REC:DEC DIG
// Sets Receiver for decoding digital data.
REC:DEC DIG POCSAG
// Prepares Receiver for decoding POCSAG.
REC:POCSAG:STAT 1
// Enables POCSAG decoding.
REC:POCSAG:RATE 1
// Selects high rate to decode.
REC:POCSAG:MESS?
// Queries for the decoded message.
REC:POCSAG:CAP?
// Queries for the decoded capcode.
REC:POCSAG:TYPE?
// Queries for the decoded message type.
REC:POCSAG:RATE?
// Queries for the current rate setting.
// 1 is returned (for high).
REC:POCSAG:STAT 0
// Disables POCSAG decoding.

RCL n
(RECeiver:RCL n)
Recalls Receiver environment (routings and settings) stored in memory location n. Range of n is 1 to 9.
RECeiver:

SCAN:

Except where noted, the following commands are for Receiver Frequency Scan or Frequency List Scan modes.

ABORT

[REceiver:SCAN:ABORT]

Stops frequency scan or frequency list scan depending on present mode.

CONTinue

[REceiver:SCAN:CONTinue]

Starts frequency scan or frequency list scan depending on present mode.

FREQuency?

[REceiver:SCAN:FREQuency?]  

Returns frequency currently being scanned in kHz.

INCrement f

[REceiver:SCAN:INCrement f]

(Frequency Scan mode only.) Sets Receiver Scan increment to f kHz. Range of f is 0.0 to 2010000.0.

PAUSE t

[REceiver:SCAN:PAUSE t]

(Frequency Scan mode only.) Sets Receiver Pause rate to t sec. Range of t is 0.0 to 99.9. Receiver Scan pause time is length of time Scan Function pauses at frequency with broken squelch. Scan Function stops permanently on squelch broken frequency if pause set to 0.0.

PAUSE?

[REceiver:SCAN:PAUSE?]  

(Frequency Scan mode only.) Returns a 1 if scanning is paused (stopped) or 0 if currently scanning.

RATe t

[REceiver:SCAN:RATe t]

Sets Receiver Scan rate to t sec. Range of t is 0.00 to 99.99. Receiver Scan rate is the time each frequency is scanned with squelch unbroken.

START f

[REceiver:SCAN:START f]

Sets Receiver Scan starting frequency to f kHz. Range of f is 250.0 to 2010000.0.
RECeiver:

SCAN:

STOP f
[RECeiver:SCAN:STOP f]
Sets Receiver Scan stopping frequency to f kHz. Range of f is 250.0 to 2010000.0.

Example:
REC:SCAN:STAR 1000 // Sets Receiver Scan starting frequency to 1 MHz.
REC:SCAN:STOP 100000 // Sets Receiver Scan stopping frequency to 100 MHz.
REC:SCAN:INC 250 // Sets Receiver Scan increment to 250 kHz.
REC:SCAN:RAT 1.5 // Sets Receiver Scan rate to 1.5 sec.
REC:SCAN:PAUS 10 // Sets Receiver Scan pause time to 10 sec.
REC:SCAN:CONT // Starts Receiver Scan.
REC:SCAN:ABOR // Stops Receiver Scan.

FREQList:

The following commands are for Receiver Frequency List Scan mode only.

PAUSE n
[RECeiver:SCAN:FREQList:PAUSE n]
Frequency List Scan Pause Time. Specifies time period in seconds for Receiver to sit on a frequency if squelch is broken. Range of n is 0.0 to 99.9.

If 0.0 is specified, Receiver sits on frequency as long as squelch is broken.

PAUSE?
[RECeiver:SCAN:FREQList:PAUSE?]
Returns current value of Frequency List Scan Pause Time.

RATe n
[RECeiver:SCAN:FREQList:RATe n]
Frequency List Scan Rate. Specifies time period in seconds for Receiver to sit on a frequency unless squelch is broken. Range of n is 0.02 to 99.99.

RATe?
[RECeiver:SCAN:FREQList:RATe?]
Returns current value of Frequency List Scan Rate.

SQUElch b
[RECeiver:SCAN:FREQList:SQUElch b]
Enables (b = 1) or disables (b = 0) Squelch.

SQUElch?
[RECeiver:SCAN:FREQList:SQUElch?]
Returns current state of Squelch.
RECeiver:

SINAD
[RECeiver:SINAD]
Displays SINAD Meter reading when followed by a SCREEN:RECeiver command.

SQuelch n
[RECeiver:SQuelch n]
Sets squelch to n. Range of n is 0.0 to 1.0.

SQuelch?
[RECeiver:SQuelch?]
Returns squelch level (0.0 to 1.0).
Example: SQU .3  // Sets Squelch level to 3/10 of total
          // allowable setting.
          SQU?      // Queries Squelch level. .3 is returned.

STORe n
[RECeiver:STORe n]
Stores current Receiver environment (routings and settings) in memory location n. Range of
n is 1 to 9.

TONE:

USER:DEFine "id",f,d
[RECeiver:TO宁:USER:DEFine "id",f,d]
Defines a User Defined Audio Tone and duration to be received; where id is the character
associated with the defined tone, f is the frequency in Hz of the defined tone and d is the
duration of the defined tone in ms. Range of id is 0 to 9 and A to T, range of f is 0.0 to
9999.9 and range of d is 20 to 9999.

Same as GENERator:TO宁:USER:DEFine command. Both commands act upon the same
User Defined Audio Tone database.

DURation? "id"
[RECeiver:TO宁:DURation? "id"]
Returns the Duration in ms (20 to 9999) of the specified User Defined Audio Tone. id is
the character associated with the defined tone. Range of id is 0 to 9 and A to T.

FREQuency? "id"
[RECeiver:TO宁:FREQuency? "id"]
Returns the frequency setting in Hz of the specified User Defined Audio Tone id. Range of
id is 0 to 9 and A to T.

INPut n
[RECeiver:TO宁:INPut n]
Selects the decode signal input as defined below:

<table>
<thead>
<tr>
<th>n</th>
<th>INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Demod Audio</td>
</tr>
<tr>
<td>1</td>
<td>SINAD/BER</td>
</tr>
<tr>
<td>2</td>
<td>Exit Mod</td>
</tr>
</tbody>
</table>

6.32
**RECeiver:**

**TONET**:  

STATe \( b \)  
\( \text{RECeiver:} \text{TONET:STATe} \; b \)  
Enables Audio Tone decoding if \( b \) is 1, disables if \( b \) is 0. **RECeiver:** **DECode TONE** command must be initiated prior to enabling Audio Tone decoding. Audio Tone decoding must be disabled after decoding is finished.

**TYPE \( xxx \)**  
\( \text{RECeiver:} \text{TONET:TYPE} \; xxx \)  
Selects Audio Tone to be decoded. Choose one of the following for \( xxx \):  

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCIR</td>
<td></td>
</tr>
<tr>
<td>ZVEI</td>
<td></td>
</tr>
<tr>
<td>NATEL</td>
<td></td>
</tr>
<tr>
<td>CCIRH</td>
<td></td>
</tr>
<tr>
<td>EEA</td>
<td></td>
</tr>
<tr>
<td>DDZEVEI</td>
<td></td>
</tr>
<tr>
<td>EURO</td>
<td></td>
</tr>
<tr>
<td>CCIRH4</td>
<td></td>
</tr>
<tr>
<td>EIA</td>
<td></td>
</tr>
<tr>
<td>DZVEI</td>
<td></td>
</tr>
<tr>
<td>TONE56</td>
<td></td>
</tr>
<tr>
<td>USER</td>
<td></td>
</tr>
</tbody>
</table>

See 6-6-2D and F.

**TONET?**  
\( \text{RECeiver:} \text{TONET?} \)  
Returns the decoded Audio Tone sequence or -1 if not available.

Example:  
REC:DEC TONE  // Sets Receiver for decoding Audio Tones.  
REC:TONET:STAT 1  // Enables Audio Tone decoding.  
REC:TONET:TYPE CCIR  // Selects CCIR as the Audio Tone type.  
REC:TONET?  // Returns the decoded Audio Tone sequence.  
REC:TONET:STAT 0  // Disables Audio Tone decoding.

**VOLUME \( n \)**  
\( \text{RECeiver:} \text{VOLUME \( n \)} \)  
Sets volume to \( n \). Range of \( n \) is 0.0 to 1.0.

**VOLUME?**  
\( \text{RECeiver:} \text{VOLUME?} \)  
Returns the volume level (0.0 to 1.0).

**VOLUME:**

AUTO \( b \)  
\( \text{RECeiver:} \text{VOLUME:AUTO \( b \)} \)  
Enables Automatic Volume Control if \( b \) is 1, disables if \( b \) is 0.

AUTO?  
\( \text{RECeiver:} \text{VOLUME:AUTO?} \)  
Returns Automatic Volume Control state. 1 is returned if enabled, 0 if disabled.
A. RECEIVING FM SIGNALS

The following command sequence receives a 96 MHz FM signal through the ANTEenna IN Connector and outputs the audio signal through the Test Set Speaker.

```dnc
DMC "REC_FM", BEGIN // Define a macro named REC_FM.
SCREEN:REC // Display the Receiver Operation Screen.
REC:FREQ 96000 // Set Receiver Frequency to 96 MHz.
REC:MODULATION:FM1 // Select FM1 for Receiver Modulation.
REC:INPUT:ANTEenna // Select the ANTEenna IN Connector for Receiver Input.
REC:INPUT:ATTENUATION 0 // Select 0 dB for the Receiver Input Attenuation Level.
N=1, X=0 // Set variables to initial values.
WHILE X=0 // Start WHILE loop to loop as long as
// squelch is unbroken.
  REC:SQU N // Set squelch 0.02 lower than last setting.
  DELAY 25 // Allow time for squelch to settle.
  N=N-0.02 // Decrement variable N (use to set squelch).
  X:=MEAS:SQU? // Set X to 1 when squelch breaks (see 6-15).
WEND // End of WHILE loop. X loops to the top as
// long as X=0 (squelch unbroken).
REC:OUTPUT: SPEAKER 1 // Routes Receiver Output to the Test Set Speaker.
END // End of macro REC_FM.
```

B. RECEIVING AM SIGNALS

The following command sequence receives a 1240 kHz AM signal through the ANTEenna IN Connector and outputs the audio signal through the DEMOD OUT Connector.

```dnc
DMC "REC_AM", BEGIN // Define a macro named REC_AM.
SCREEN:REC // Display the Receiver Operation Screen.
REC:FREQ 1240 // Set Receiver Frequency to 96 MHz.
REC:MODULATION:AM1 // Select AM1 for Receiver Modulation.
REC:INPUT:ANTEenna // Select the ANTEenna IN Connector for Receiver Input.
REC:INPUT:ATTENUATION 0 // Select 0 dB for the Receiver Input Attenuation Level.
N=1, X=0 // Set variables initially.
WHILE X=0 // Start WHILE loop to loop as long as
// squelch is unbroken.
  REC:SQU N // Set squelch 0.02 lower than last setting.
  DELAY 25 // Allow time for squelch to settle.
  N=N-0.02 // Decrement variable N (use to set squelch).
  X:=MEAS:SQU? // Set X to 1 when squelch breaks (see 6-15).
WEND // End of WHILE loop. X loops to the top as
// long as X=0 (squelch unbroken).
REC:OUTPUT:DEMOD 1 // Route Receiver Output to the DEMOD OUT Connector.
END // End of macro REC_AM.
```
C. DECODING DTMF CODED SIGNALS

The following command sequence receives a 450 MHz FM modulated DTMF signal and routes the demodulated signal to the AUDIO OUT Connector and the Test Set Speaker.

*DMC "REC_DTMF",BEGIN // Define a macro named REC_DTMF.
SCREEN:REC // Display Receiver Operation Screen.
REC:FREQ 450000 // Set Receiver Frequency to 450 MHz.
REC:MODULATION:FM1 // Select FM1 for Receiver Modulation.
REC:INPUT:ATTENUATION 0 // Select 0 dB for Receiver Input
// Attenuation Level.
N=1,X=0 // Set variables initially.
WHILE X=0
  REC:SQU N // Set squelch 0.02 lower than last setting.
  DELAY 25 // Allow time for squelch to settle.
  N=N-0.02 // Decrement N (use to set squelch).
  X:=MEAS:SQU? // Set X to 1 when squelch breaks (see 6-16).
WEND // End of WHILE loop. X loops to the top as
// long as X=0 (squelch unbroken).
REC:OUTPUT:AUDIO 1 // Route the demodulated signal to the
// AUDIO OUT Connector.
REC:OUTPUT:SPEAKER 1 // Route Receiver Output to the Test Set
// Speaker.
REC:DEC DTMF // Set Receiver for decoding DTMF signals.
REC:DTMF:STAT 1 // Enable DTMF decoding.
END // End of macro REC_DTMF.

D. DECODING CCIR CODED SIGNALS

The following command sequence receives a 450 MHz FM modulated Audio signal and routes the demodulated signal to the AUDIO OUT Connector and the Test Set Speaker.

*DMC "REC_CCIR",BEGIN // Define a macro named REC_CCIR.
SCREEN:REC // Display the Receiver Operation Screen.
REC:FREQ 450000 // Set Receiver Frequency to 450 MHz.
REC:MODULATION:FM1 // Select FM1 for Receiver Modulation.
REC:INPUT:ANT // Select the ANTENNA IN Connector for the
// Receiver Input.
REC:INPUT:ATTENUATION 0 // Select 0 dB for the Receiver Input
// Attenuation Level.
N=1,X=0 // Set variables initially.
WHILE X=0
  REC:SQU N // Set squelch 0.02 lower than last setting.
  DELAY 25 // Allow time for squelch to settle.
  N=N-0.02 // Decrement N (use to set squelch).
  X:=MEAS:SQU? // Set X to 1 when squelch breaks (see 6-16).
WEND // End of WHILE loop. X loops to the top as
// long as X=0 (squelch unbroken).

/* (macro continues on next page) */
E. DECODING DCS CODED SIGNALS

The following command sequence receives a 450 MHz FM modulated DCS signal and decodes the signal until Soft Function Key F1 is pressed. The decoded DCS digits are printed to the Host.

*DMC "REC_DCS",BEGIN
VAR CODE
SCREEN:REC
RECFREQ 450000
RECMODULATION:FM1
RECINPUT:ANT
RECINPUT:ATTENUATION 0
N=1,X=0
WHILE X=0
   REC:SQU N
   DELAY 25
   N=N-0.02
   X:=MEAS:SQU?
WEND
KEYPAD:CLAIM
REC:DEC DIG
REC:DIG DCS
RECDCS:STAT 1
WHILE (SYSTEM:KEY? != F1)
   TPause
   CODE:=REC:DCS:NORM?)
   IF (CODE != "-1")
      PPRINT CODE
ENDIF
ENDW
REC:DCS:STAT 0
KEYPAD:UNCLAIM
END

// Define a macro named REC_DCS.
// Define variable to hold decoded DCS.
// Displays the Receiver Operation Screen.
// Sets Receiver Frequency to 450 MHZ.
// Selects FM1 for Receiver Modulation.
// Selects the ANTEENNA IN Connector for the Receiver Input.
// Selects 0 dB for the Receiver Input Attenuation Level.
// Set variables initially.
// Start WHILE loop to loop as long as squelch is unbroken.
// Set squelch 0.02 lower than last setting.
// Allows time for squelch to settle.
// Decrement N (used to set squelch).
// Set X to 1 when squelch breaks (see 6-16).
// End of WHILE loop. X loops to the top as long as X=0 (squelch unbroken).
// Directs all Keyboard Input to TMAC so Operation Screen is not changed.
// Sets Receiver for decoding digital data.
// Prepares Receiver for decoding DCS.
// Enable DCS decoding.
// Loop until F1 Key is pressed.
// Allow TMAC to share processor time with Test Set (see 2-12).
// Decode a Normal DCS digits, if received.
// If DCS digits are decoded.
// print them to the Host.
// End of IF statement.
// End of While loop.
// Release Keyboard for normal use.
// End of macro REC_DCS.
F. DECODING AUDIO USER DEFINED CODED SIGNALS

The following command sequence receives a 450 MHz FM modulated Audio signal, decodes the signal using the RF Generator Audio User Defined Tones and routes the demodulated signal to the AUDIO OUT Connector and the Test Set Speaker.

*DMC "REC_AUDIO_USER"  // Define a macro named REC_AUDIO_USER.

// DEFINE THE TONES TO BE DECODED

SCREEN:GEN
GEN:TONE:TYPE:USER
GEN:TONE:USER:DEF 0,900,150
GEN:TONE:USER:DEF 1,1100,80
GEN:TONE:USER:DEF 2,1200,80
GEN:TONE:USER:DEF 3,1300,80
GEN:TONE:USER:DEF 4,1400,80
GEN:TONE:USER:DEF A,0,40

// Displays RF Generator Operation Screen.
// Selects User Defined for the Audio Code.
// Defines 0 Tone at 900 Hz and 150 ms.
// Defines 1 Tone at 1100 Hz and 80 ms.
// Defines 2 Tone at 1200 Hz and 80 ms.
// Defines 3 Tone at 1300 Hz and 80 ms.
// Defines A Tone at 0 Hz and 40 ms.

SCREEN:REC
REC:FREQ 450000
REC:MODULATION:FM1
REC:INPUT:ANT

REC:INPUT:ATTENUATION 0
N=1,X=0
WHILE X=0
  REC:SQU N
  DELAY 25
  N=N-0.02
  X:=MEAS:SQU?
WEND

// Displays the Receiver Operation Screen.
// Sets Receiver Frequency to 450 MHz.
// Selects FM1 for Receiver Modulation.
// Selects the ANTENNA IN Connector for the Receiver Input.
// Selects 0 dB for the Receiver Input
// Attenuation Level.
// Set variables initially.
// Start WHILE loop to loop as long as squelch is unbroken.
// Set squelch 0.02 lower than last setting.
// Allows time for squelch to settle.
// Decrement variable N (used to set squelch).
// Set X to 1 when squelch breaks (see 6-16).
// End of WHILE loop. X loops to the top as long as X=0 (squelch unbroken).
// Routes the demodulated signal to the AUDIO OUT Connector.
// Routes Receiver Output to the Test Set Speaker.
// Selects User Defined for the Audio Code
// Type.
// Sets Receiver for decoding Audio Tones.
// Enables Audio decoding.
// Returns the decoded Audio Tone sequence.
// Disables Audio decoding.
// End of macro REC_AUDIO_USER.
G. DECODING POCSAG CODED SIGNALS

The following command sequence receives a 450 MHz FM modulated POCSAG signal, decodes the signal and prints the POCSAG messages to the Host until Soft Function Key F1 is pressed.

*DMC "REC_POCSAG",BEGIN
STRING CODE // Define a macro named REC_POCSAG.
// Define a string variable to hold decoded
// POCSAG message.
SCREEN:REC // Displays the Receiver Operation Screen.
REC:FREQ 450000 // Sets Receiver Frequency to 450 MHz.
REC:MODULATION:FM1 // Selects FM1 for the Receiver Modulation.
REC:INPUT:ANT // Selects the ANTENNA IN Connector for the
// Receiver Input.
REC:INPUT:ATTENUATION 0 // Selects 0 dB for the Receiver Input
// Attenuation Level.
N=1,X=0 // Set variables initially.
WHILE X=0 // Start WHILE loop to loop as long as
// squelch is unbroken.
   REC:SQU N // Set squelch 0.02 lower than last setting.
   DELAY 25 // Allows time for squelch to settle.
   N=N-0.02 // Decrement variable N (used to set
// squelch).
   X:=MEAS:SQU? // Set X to 1 when squelch breaks (see 6-16).
WEND // End of WHILE loop. X loops to the top as
// long as X=0 (squelch unbroken).
REC:OUTPUT:SPEAKER 1 // Routes Receiver Output to the Test Set
// Speaker.
REC:POCSAG:RAT 1 // Sets POCSAG rate to be decoded to High.
KEYPAD:CLAIM // Directs all Keyboard Input to TMAC so
// Operation Screen is not changed.
REC:DEC DIG // Sets Receiver for decoding digital data.
REC:DIG POCSAG // Prepares Receiver for decoding POCSAG.
REC:POCSAG:STAT 1 // Enable POCSAG decoding.
WHILE (SYSTEM:KEY? != F1) // Loop until F1 Key is pressed.
   TPAUSE // Allow TMAC to share processor time with
// Test Set (see 2-12).
   CODE=STR(REC:POCSAG:MESS?) // Decode a POCSAG message, if received.
   IF (CODE != "-1") // If a POCSAG message was decoded,
   PPRINT CODE // print message to the Host.
   ENDF // End of IF statement.
WEND // End of While loop.
REC:POCSAG:STAT 0 // Disable POCSAG decoding.
KEYPAD:UNCLAIM // Release Keyboard for normal use.
END // End of macro REC_POCSAG.
H. RECEIVER SETUP FOR CELLULAR OPERATION

The following command sequence configures the Receiver Mode for receiving Cellular signals through the ANTENNA IN Connector.

SCREEN:REC // Displays the Receiver Operation Screen.
REC:INPUT:ANTENNA // Selects the Antenna for Receiver Input.
M_DEV:RANG:UPP 10 // Sets Deviation Meter Range to 10 kHz.
M.AF:RES 1 // Sets AF Meter Gate Time to .1 sec (1 Hz).
6-7 DUPLEX

6-7-1 DUPLEX GENERAL COMMANDS

DUPlex:

METER:

DISTortion
[DUplex:METER:DISTortion]
Displays Distortion Meter on Duplex Operation Screen.

MODMeter
[DUplex:METER:MODMeter]
Displays Modulation Meter on Duplex Operation Screen.

OFF
[DUplex:METER:OFF]
Disables Modulation, Distortion and SINAD Meters.

SINAD
[DUplex:METER:SINAD]
Displays SINAD Meter on Duplex Operation Screen.

SPEAKEr:SOURce type
[DUplex:SPEAKEr:SOURce type]
Selects Test Set Speaker Input. Select OFF, FGEN (Function Generator), SINAD
(SINAD/BER IN Connector) or EXTMOD (EXT MOD IN Connector) for type. Command takes
effect when RF Generator Operation Screen is updated.

STORe n
[DUplex:STORe n]
Stores current Duplex environment (routings and settings) in memory location n. Range of n
is 1 to 9.

RCL n
[DUplex:RCL n]
Recalls Duplex environment (routings and settings) stored in memory location n. Range of n
is 1 to 9.
6-7-2 DUPLEX TRANSMITTER COMMANDS

DUPlex:

INPut:

AGC:

AUTO
[DUPlex:INPut:AGC:AUTO]
Sets Automatic Gain Control to automatic setting.

MANual n
[DUPlex:INPut:AGC:MANual n]
Sets Automatic Gain Control to manual setting and sets level to n. Range of n is 0 to 255.

USER:XXX
[DUPlex:INPut:AGC:USER:xxx]
Sets Automatic Gain Control to User setting. Select one of the following for xxx: MEASure, SPEech, DATA, HIGH, TYPE1, TYPE2 or TYPE3.

ANTenna
[DUPlex:INPut:ANTenna]
Selects ANTENNA IN Connector as Transmitter Input Source. Displays Signal Strength Meter reading on the Duplex Transmitter Operation Screen.

The following Attenuation commands control the Input Attenuation for the Duplex Transmitter and selection/deselection of LNA (Low Noise Amplifier). Use DUPlex:INPut:ATTenuation:LNA to select LNA and 0 dB of input attenuation. To deselection LNA and select a specific value of attenuation, use DUPlex:INPut:ATTenuation.

ATTenuation:LNA
[DUPlex:INPut:ATTenuation:LNA]
Sets Duplex Input Attenuation to LNA.

ATTenuation:LNA?
[DUPlex:INPut:ATTenuation:LNA?]
Returns current state of the Low Noise Amplifier. Returns 1 if LNA is selected; 0 otherwise.

LNA (Low Noise Amplifier) has the same attenuation as 0 dB (see DUPlex:INPut:ATTenuation n), but LNA has a lower noise figure. LNA is the preferred option for performing off-the-air applications.

ATTenuation n
[DUPlex:INPut:ATTenuation n]
Sets Duplex Input Attenuation to n dB. Possible values for n: 0, 5, 10, 15, 20, 25, 30.

ATTenuation?
[DUPlex:INPut:ATTenuation?]
Returns the current value of Input Attenuation.
DUplex:

INPut:

CHANnel n
[DUPlex:INPut:CHANnel n]
Sets Duplex Transmitter Frequency to cellular channel n (1 to 2047, depending on format) in the format selected using the DUPlex:INPut:CHANnel:FORMat command. The Duplex Transmitter Operation Screen displays selected cellular channel when Duplex Transmitter is in Channel Mode (DUPlex:INPut:MODE CHAN command).

CHANnel:

FORMat:

AMPS:

FORward
[DUPlex:INPut:CHANnel:FORMat:AMPS:FORward]
Selects AMPS (NADC-U8) Forward channels.

REVerse
[DUPlex:INPut:CHANnel:FORMat:AMPS:REVerse]
Selects AMPS (NADC-U8) Reverse channels.

ETACS:

FORward
[DUPlex:INPut:CHANnel:FORMat:ETACS:FORward]
Selects ETACS Forward channels.

REVerse
[DUPlex:INPut:CHANnel:FORMat:ETACS:REVerse]
Selects ETACS Reverse channels.

NAMPS:

FORward
[DUPlex:INPut:CHANnel:FORMat:NAMPS:FORward]
Selects NAMPS Forward channels.

REVerse
[DUPlex:INPut:CHANnel:FORMat:NAMPS:REVerse]
Selects NAMPS Reverse channels.

BAND:x
[DUPlex:INPut:CHANnel:FORMat:NAMPS:BAND:x]
Selects NAMPS Narrow Analog channel designator. Range of x is Lower, Middle or Upper.
DUplex:

INPut:

CHANnel:

FORMAT:

NT400:

FORward
[DUPlex:INPut:CHANnel:FORMAT:NT400:FORward]
Selects NT400 Forward channels.

REVerse
[DUPlex:INPut:CHANnel:FORMAT:NT400:REVerse]
Selects NT400 Reverse channels.

NADC:

FORward
[DUPlex:INPut:CHANnel:FORMAT:NADC:FORward]
Selects NADC Forward. Utilizes current setting of NADC band.

REVerse
[DUPlex:INPut:CHANnel:FORMAT:NADC:REVerse]
Selects NADC Reverse. Utilizes current setting of NADC band.

BAND:x
[DUPlex:INPut:CHANnel:FORMAT:NADC:BAND:x]
Selects NADC band. Range of x is U8, U4 or HYper.

BAND?
[DUPlex:INPut:CHANnel:BAND?]
Returns band for Duplex Transmitter Channel Format. Returns one of the following bands for the associated channel format:

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>BAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC</td>
<td>U8, U4 or HY</td>
</tr>
<tr>
<td>ETACS</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>NAMPS</td>
<td>LOWER, MIDDLE or UPPER</td>
</tr>
</tbody>
</table>

FORMAT?
[DUPlex:INPut:CHANnel:FORMAT?]
DUPlex:

INPut:

FIND:

FREQuency?
[DUPlex:INPut:FIND:FREQ.?]
Returns frequency of first signal with amplitude larger than the Find reference level.

REFERENCE n
[DUPlex:INPut:FIND:REFERENCE n]
Sets Find reference level to n dB.

REFERENCE?
[DUPlex:INPut:FIND:REFERENCE?]
Returns Find reference level in dB.

Example:
DUPlex:INPut:FIND:REF -55  // Sets Find Reference Level to -55 dBm.
DUPlex:INPut:FIND:FREQ?  // Returns the lowest frequency (in kHz) containing a signal greater than -55 dBm.
DUPlex:INPut:FIND:REF?  // Queries the current Find Reference Level (in dBm). -55 is returned.

FREQuency n [units]
[DUPlex:INPut:FREQuency n [units]]
Sets Duplex Transmitter Frequency. Range of n is 250.0 kHz to 2010.0000 MHz. Select HZ, KHZ, MHZ for units. units is optional with a default of KHZ.

FREQuency?
[DUPlex:INPut:FREQuency?]  
Returns RF Generator Frequency in kHz.

Example: DUPlex:INPut:FREQ 145 MHZ  // Set RF Generator Frequency to 145 MHz.
DUPlex:INPut:FREQ?  // Query the RF Generator Frequency.
                     // 145000 is returned (145000 kHz).

METER:

DEVRms
[DUPlex:INPut:METER:DEVrms]
Displays Deviation Meter (RMS) when followed by a SCREEN:DUPlex command.

DISTortion
[DUPlex:INPut:METER:DISTortion]
Displays Distortion Meter when followed by a SCREEN:DUPlex command.

MODMeter
[DUPlex:INPut:METER:MOdMeter]
Displays Modulation Meter when followed by a SCREEN:DUPlex command.

PMRms
[DUPlex:INPut:METER:PMRms]
Displays Phase Meter (RMS) when followed by a SCREEN:DUPlex command.
DUPlекс:

INPut:

METER:

SINAD
[DUPlex:INPut:METER:SINAD]
Displays SINAD Meter when followed by a SCREEN:DUPlекс command.

MODE type
[DUPlex:INPut:MODE type]
Selects the Duplex Transmitter Mode. Channel Mode displays cellular channel frequency according to DUP:INP:CHAN commands.

<table>
<thead>
<tr>
<th>MODE</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>DIRECT</td>
</tr>
<tr>
<td>Channel</td>
<td>CHANNEL</td>
</tr>
<tr>
<td>Frequency Scan</td>
<td>SCAN</td>
</tr>
<tr>
<td>Frequency List</td>
<td>LIST</td>
</tr>
<tr>
<td>Frequency List Scan</td>
<td>FLScan</td>
</tr>
</tbody>
</table>

MODE?
[DUPlex:INPut:MODE?] Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or FREQUENCY LIST SCAN).

MODulation:

AMn
[DUPlex:INPut:MODulation:AMn]
Selects Amplitude Modulation. Select 1 or 2 for n.
Example: DUP:INP:MOD:AM1 // Selects AM1 for Duplex Transmitter
// Modulation.

BFO
[DUPlex:INPut:MODulation:BFO]
Selects Beat Frequency Oscillation.

FMn
[DUPlex:INPut:MODulation:FMn]
Selects Frequency Modulation. Select 1, 2, 3 or 4 for n.

PM
[DUPlex:INPut:MODulation:PM]
Selects Phase Modulation.
DUplex:

INPut:

MODulation:

USER:

FILTER \( f \)
[DUPlex:INPut:MODulation:USER:FILTER f]
Sets User Defined Modulation IF Filter to \( f \) kHz. Select 3, 30 or 300 for \( f \).

MODulation:type
[DUPlex:INPut:MODulation:USER:MODulation:type]
Selects User Defined Modulation. Enter one of the following for type: FM, AM, BFO, PM or DATA (FM DATA).

POST:

APASs
[DUPlex:INPut:MODulation:USER:POST:APASs]
Selects All Pass Post Detection Filter for User Defined Modulation.

BPASs \( f_l, f_h \)
[DUPlex:INPut:MODulation:USER:POST:BPASs f_l, f_h]
Selects Bandpass Post Detection Filter for User Defined Modulation. Lower cutoff frequency is set to \( f_l \) kHz with range of 0.5 to 20. Higher cutoff frequency is set to \( f_h \) kHz with a range of 0.1 to 30.

CWEight
Selects C-Weighted Post Detection Filter (C-message noise weighting curve response) for the User Defined Modulation. Refer to MIL-STD-188-200.

HPASs \( f_l \)
[DUPlex:INPut:MODulation:USER:POST:HPASs f_l]
Selects High-Pass Post Detection Filter for User Defined Modulation. Cutoff frequency is set to \( f_l \) kHz with a range of 0.5 to 20.0.

LPASs \( f_h \)
[DUPlex:INPut:MODulation:USER:POST:LPASs f_h]
Selects Low-Pass Post Detection Filter for User Defined Modulation. Cutoff frequency is set to \( f_h \) kHz with a range of 0.1 to 30.0.
DUPl ex:

INPut:

MODulation?
[DUPlex:INPut:MODulation?]
Returns the current Duplex Transmitter Modulation (FM, AM, BFO, PM, DATA or USER).
Example:
DUP:INP:MOD:USER:FM // Selects User Defined FM for the
 // Duplex Transmitter Modulation.
 // Filter with 15 kHz cutoff
 // frequency.
 // Modulation. User is returned.

SCAN:

Except where otherwise noted, the following commands are for Duplex Transmitter
Frequency Scan or Frequency List Scan modes.

ABORT
[DUPlex:INPut:SCAN:ABORT]
Stops frequency scan or frequency list scan depending on present mode.

CONTinue
[DUPlex:INPut:SCAN:CONTinue]
Starts frequency scan or frequency list scan depending on present mode.

FREQuency?
[DUPlex:INPut:SCAN:FREQuency?]
Returns current frequency in kHz being scanned.

INCrement n
[DUPlex:INPut:SCAN:INCrement n]
(Frequency Scan mode only.) Specifies increment in kHz between frequencies to be
scanned. Range of n is 0.0 to 2010250.0 kHz.

PAUse n
[DUPlex:INPut:SCAN:PAUse n]
(Frequency Scan mode only.) Pause Time. Specifies time period in seconds for
Duplex Transmitter to sit on a frequency if squelch is broken. Range of n is 0.0
to 99.9.

If 0.0 is selected, the Duplex Transmitter sits on current frequency as long as squelch
is broken.

PAUse?
[DUPlex:INPut:SCAN:PAUse?]'
(Frequency Scan mode only.) Returns a 1 if scanning is paused (stopped) or 0 if
currently scanning.
**DUPlex:**

**INPut:**

**SCAN:**

**RATe** \( n \)
\[ DUX:INPut:SCAN:RATe \ n \]
(Frequency Scan mode only) Scan Rate. Specifies time period in seconds for Duplex Transmitter to sit on a frequency unless squelch is broken. Range of \( n \) is 0.02 to 99.99.

**START** \( f \)
\[ DUX:INPut:SCAN:START \ f \]
(Frequency Scan mode only) Starting Frequency value in kHz. Specifies lower limit frequency for scanning. Range of \( f \) is 250.0 to 2010000.0.

Start frequency must be a valid value and less than Stop frequency for Scan to operate.

**STOP** \( f \)
\[ DUX:INPut:SCAN:STOP \ f \]
(Frequency Scan mode only) Stop Frequency value in kHz. Specifies upper limit frequency for scanning. Range of \( f \) is 250.0 to 2010000.0.

**FREQList:**

The following commands are for Duplex Transmitter Frequency List Scan mode only.

**PAUSE** \( n \)
\[ DUX:INPut:SCAN:FREQList:PAUSE \ n \]
Frequency List Scan Pause Time. Specifies time period in seconds for Duplex Transmitter to sit on a frequency if squelch is broken. Range of \( n \) is 0.0 to 99.9.

If 0.0 is specified, Duplex Transmitter sits on frequency as long as squelch is broken.

**PAUSE?**
\[ DUX:INPut:SCAN:FREQList:PAUSE? \]
Returns current value of Frequency List Scan Pause Time.

**RATe** \( n \)
\[ DUX:INPut:SCAN:FREQList:RATe \ n \]
Frequency List Scan Rate. Specifies time period in seconds for Duplex Transmitter to sit on a frequency unless squelch is broken. Range of \( n \) is 0.02 to 99.99.

**RATe?**
\[ DUX:INPut:SCAN:FREQList:RATe? \]
Returns current value of Frequency List Scan Rate.
**DUPlex:**

**INPut:**

**SCAN:**

**FREQList:**

**SQUElch** \( b \)

\[[DUPlex:INPut:SCAN:FREQList:SQUElch \ b]\]

Enables \( b = 1 \) or disables \( b = 0 \) Squelch.

**SQUElch?**

\[[DUPlex:INPut:SCAN:FREQList:SQUElch?]\]

Returns current state of Squelch.

**TO:**

**AUDio** \( b \)

\[[DUPlex:INPut:TO:AUDio \ b]\]

Routes demodulated Receiver Input to the AUDIO OUT Connector if \( b = 1 \). Disconnects the AUDIO OUT Connector if \( b = 0 \).

**DEMOD** \( b \)

\[[DUPlex:INPut:TO:DEMOD \ b]\]

Routes demodulated Receiver Input to the DEMOD OUT Connector if \( b = 1 \). Disconnects the DEMOD OUT Connector if \( b = 0 \).

**SPEAKer** \( b \)

\[[DUPlex:INPut:TO:SPEAKer \ b]\]

Routes demodulated Receiver Input to the Test Set Speaker if \( b = 1 \). Disconnects the Test Set Speaker if \( b = 0 \).

**TR**

\[[DUPlex:INPut:TR]\]

Selects T/R Connector as Duplex Transmitter Input Source. Displays Power Meter on the Duplex Transmitter Operation Screen.

**VOLUME:**

**AUTO** \( b \)

\[[DUPlex:INPut:VOLUME:AUTO \ b]\]

Enables Automatic Volume Control if \( b = 1 \), disables if \( b = 0 \).

**AUTO?**

\[[DUPlex:INPut:VOLUME:AUTO?]\]

Returns Automatic Volume Control state. 1 is returned if enabled, 0 if disabled.
6-7-3  DUPLEX RECEIVER COMMANDS

DUPlex:

OUTput:

AUDio b
[DUPlex:OUTput:AUDio b]
Routes AF Generator Output to the AUDIO OUT Connector if b is 1. Disconnects the AUDIO OUT Connector if b is 0.

CHANnel n
[DUPlex:OUTput:CHANnel n]
Sets Duplex Receiver Frequency to cellular channel n (1 to 2047, depending on format) in the format selected using the DUPlex:OUTput:CHANnel:FORMAT command. The Duplex Receiver Operation Screen displays selected cellular channel when Duplex Receiver is in Channel Mode (DUPlex:OUTput:MODE CHAN command).

CHANnel:

FORMAT:

AMPS:

FORward
[DUPlex:OUTput:CHANnel:FORMAT:AMPS:FORward]
Selects AMPS (NADC-U8) Forward channels.

REVerse
[DUPlex:OUTput:CHANnel:FORMAT:AMPS:REVerse]
Selects AMPS (NADC-U8) Reverse channels.

ETACS:

FORward
[DUPlex:OUTput:CHANnel:FORMAT:ETACS:FORward]
Selects ETACS Forward channels.

REVerse
[DUPlex:OUTput:CHANnel:FORMAT:ETACS:REVerse]
Selects ETACS Reverse channels.

NADC:

FORward
[DUPlex:OUTput:CHANnel:FORMAT:NADC:FORward]
Selects NADC Forward. Utilizes current setting of NADC band.

REVerse
[DUPlex:OUTput:CHANnel:FORMAT:NADC:REVerse]
Selects NADC Reverse. Utilizes current setting of NADC band.

BAND:x
[DUPlex:OUTput:CHANnel:FORMAT:NADC:BAND:x]
Selects NADC band. Range of x is U8, U4 or HYper.
DUPlex:

OUTput:

CHANnel:

FORMat:

NAMPS:

FORward
[DUPlex:OUTput:CHANnel:FORMat:NAMPS:FORward]
Selects NAMPS Forward channels.

REVerse
[DUPlex:OUTput:CHANnel:FORMat:NAMPS:REVerse]
Selects NAMPS Reverse channels.

BAND:x
[DUPlex:OUTput:CHANnel:FORMat:NAMPS:BAND:x]
Selects NAMPS Narrow Analog channel designator. Range of x is Lower, Middle or Upper.

NT400:

FORward
[DUPlex:OUTput:CHANnel:FORMat:NT400:FORward]
Selects NT400 Forward channels.

REVerse
[DUPlex:OUTput:CHANnel:FORMat:NT400:REVerse]
Selects NT400 Reverse channels.

BAND?
[DUPlex:OUTput:CHANnel:BAND?]
Returns band for Duplex Receiver Channel Format. Returns one of the following bands for the associated channel format:

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>BAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC</td>
<td>U8, U4 or HY</td>
</tr>
<tr>
<td>ETACS</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>NAMPS</td>
<td>LOWER, MIDDLE or UPPER</td>
</tr>
</tbody>
</table>

FORMat?

DEMOD b
[DUPlex:OUTput:DEMOD b]
Routes AF Generator Output to the DEMOD OUT Connector if b is 1. Disconnects the DEMOD OUT Connector if b is 0.
DUPlEx:

OUTput:

DUPlEx

[DUPlEx:OUTput:DUPlEx]
Routes RF Generator Output to the DUplex OUT Connector and disconnects the T/R Connector.

FREQuency n [units]

[DUPlEx:OUTput:FREQuency n [units]]
Sets Duplex Receiver Frequency. Range of n is 250.0 kHz to 2010.0000 MHz. Select HZ, KHz, MHZ for units. units is optional with a default of KHz.

FREQuency?

[DUPlEx:OUTput:FREQuency?] 
Returns Duplex Receiver Frequency in kHz.

Example:

DUP:OUT:FREQ 900 // Set RF Generator Frequency to 900 kHz.
DUP:OUT:FREQ? // Query the RF Generator Frequency.
// 900 is returned (900 kHz).

LEVEL:

DBm n

[DUPlEx:OUTput:LEVEL:DBm n]
Sets RF Level to n dBm. Range of n is 0.0 to -137.0

DBm?

[DUPlEx:OUTput:LEVEL:DBm?] 
Returns RF Level in dBm.

Example:

DUP:OUT:LEV:DB -40(290,631),(419,648) // Sets Duplex Receiver Output Level to
(282,641),(427,656) // -40 dBm.
// -40 is returned.

METER:

AF

[DUPlEx:OUTput:METER:AF]
Displays AF Meter on the Duplex Receiver Operation Screen.

DISTortion

[DUPlEx:OUTput:METER:DISTortion]
Displays Distortion Meter on the Duplex Receiver Operation Screen.

DMM

[DUPlEx:OUTput:METER:DMM]
Displays Digital Multimeter on the Duplex Receiver Operation Screen.

SINAD

[DUPlEx:OUTput:METER:SINAD]
Displays SINAD Meter on the Duplex Receiver Operation Screen.
DUPlx:

OUTput:

MODE type
[DUPlex:OUTput:MODE type]
Selects the Duplex Receiver Mode. Channel Mode displays cellular channel frequency according to DUP:OUT:CHAN commands.

<table>
<thead>
<tr>
<th>MODE</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>DIREc</td>
</tr>
<tr>
<td>Channel</td>
<td>CHAnnel</td>
</tr>
<tr>
<td>Frequency Scan</td>
<td>SCAN</td>
</tr>
<tr>
<td>Frequency List</td>
<td>LIST</td>
</tr>
<tr>
<td>Frequency List Scan</td>
<td>FLScan</td>
</tr>
</tbody>
</table>

MODE?
[DUPlex:OUTput:MODE?]
Returns current mode (DIRECT, CHANNEL, FREQUENCY SCAN, FREQUENCY LIST or FREQUENCY LIST SCAN).

OFFSet /
[DUPlex:OUTput:OFFSet ]
Sets Offset Frequency to f kHz. Range of f is -999749.9 to 999749.9.

OFFSet?
[DUPlex:OUTput:OFFSet?]
Returns Offset Frequency in kHz.

SCAN:

Except where noted, the following commands are for Duplex Receiver Frequency Scan or Frequency List Scan modes.

ABORT
[DUPlex:OUTput:SCAN:ABORT]
Stops frequency scan or frequency list scan (depending on present mode).

CONTinue
[DUPlex:OUTput:SCAN:CONTinue]
Starts frequency scan or frequency list scan (depending on present mode).

FREQuency?
[DUPlex:OUTput:SCAN:FREQuency?]
Returns current frequency in kHz being generated.

INCrement n
[DUPlex:OUTput:SCAN:INCrement n]
(Frequency Scan mode only.) Specifies increment in kHz between frequencies to be generated. Range of n is 0.0 to 2010000.0.
DUplex:

OUTput:

SCAN:

PAUse?
[DUPlex:OUTput:SCAN:PAUse?]
(Frequency Scan mode only) Returns a 1 if scanning is paused (stopped) or 0 if currently scanning.

RATE n
[DUPlex:OUTput:SCAN:RATE n]
(Frequency Scan mode only) Scan Rate. Specifies time period in seconds for each generated frequency. Range of n is 0.02 to 99.99.

START f
[DUPlex:OUTput:SCAN:START f]
(Frequency Scan mode only) Start Frequency. Specifies lower limit frequency in kHz for first generated frequency. Range of f is 250.0 to 2010000.0.

STOP f
[DUPlex:OUTput:SCAN:STOP f]
(Frequency Scan mode only) Stop Frequency. Specifies upper limit frequency in kHz or last generated frequency. Range of f is 250.0 to 2010000.0.

FREQList:
The following applies to Duplex Receiver Frequency List Scan mode only.

RATE n
[DUPlex:OUTput:SCAN:FREQList:RATE n]
Frequency List Scan Rate. Specifies time period in seconds for Duplex Receiver to generate the current frequency prior to hopping to next frequency. Range of n is 0.02 to 99.99.

TR
[DUPlex:OUTput:TR]
Routes Duplex Receiver Output to the T/R Connector and disconnects DUPLEX OUT Connector.
6-7-4  REMOTE DUPLEX EXAMPLES

A. GENERATING AND RECEIVING FM SIGNALS

The following sequence of commands generates a 160 MHz FM modulated signal routed to the DUPLEX OUT Connector and receives a 165 MHz FM modulated signal through the ANTENNA IN Connector:

SCREEN:DUPLEX  // Displays the Duplex Operation Screen
DUP:OUT:FREQ 160000  // Sets 160 MHz Duplex Receiver Frequency.
DUP:OUT:DUP  // Routes the DUPLEX OUT Connector for
               // the Duplex Receiver Output.
DUP:OUT:LEV:DB -30  // Sets Duplex Receiver Output Level to
                     // -30 dBm.
FGEN:GEN1:STATE 1  // Activates SOURCE 1 (AF Generator 1).
FGEN:GEN1:FREQ 2000  // Sets SOURCE 1 AF frequency to 2000 Hz.
FGEN:GEN1:MOD:FM  // Sets SOURCE 1 Modulation to FM.
FGEN:GEN1:MODL 5  // Sets SOURCE 1 Modulation Level to 5 kHz.
FGEN:GEN1:SHAPE:SIN  // Sets SOURCE 1 Wave Shape to a sine wave.
DUP:INP:FREQ 165000  // Sets Duplex Transmitter Frequency to
                      // 165 MHz.
DUP:INP:ANT  // Selects ANTENNA IN Connector for Duplex
              // Transmitter Input.
DUP:INP:ATT 0  // Sets Duplex Transmitter Input Attenuation
                // Level to 0 dB.
DUP:INP:MOD:FM1  // Selects FM1 for the Duplex Transmitter
                 // Modulation type.

B. GENERATING AND RECEIVING AM SIGNALS

The following sequence of commands generates a 1.4 MHz AM signal (50% modulation) modulated with Input from the MIC/ACC IN/OUT Connector and receives a 1 MHz AM modulated signal through the ANTENNA IN Connector:

SCREEN:DUPLEX  // Displays the Duplex Operation Screen
DUP:OUT:FREQ 140000  // Sets 140 MHz Duplex Receiver Frequency.
DUP:OUT:DUP  // Selects the DUPLEX OUT Connector for
               // the Duplex Receiver Output.
DUP:OUT:LEV:DB -30  // Sets Duplex Receiver Output Level to
                     // -30 dBm.
FGEN:MIC:STATE 1  // Activates SOURCE 1 (AF Generator 1).
FGEN:MIC:MODL 50  // Sets SOURCE 1 Modulation Level to 50%.
PTT:STATE 1  // Enables MIC/ACC Connector.
DUP:INP:FREQ 1 MHz  // Sets Duplex Transmitter Frequency to
                    // 1 MHz.
DUP:INP:ANT  // Selects ANTENNA IN Connector for Duplex
              // Transmitter Input.
DUP:INP:ATT 0  // Sets Duplex Transmitter Input Attenuation
                // Level to 0 dB.
DUP:INP:MOD:AM1  // Selects AM1 for the Duplex Transmitter
                 // Modulation type.
6-8 AF GENERATOR

6-8-1 AF GENERATOR COMMANDS

FGEN:

BOOST b
[FGEN:BOOST b]
Enables (b = 1) or disables (b = 0) Generator #1 Boost.

Synchronizes Generator #2 with Generator #1 to allow AUDIO output level to reach 4.0 Vrms.

BOOST?
[FGEN:BOOST?]  
Returns current state of Generator #1 Boost.

DATA:
(Data Generator is used for BER Meter tests.)

LEVEL n
[FGEN:DATA:LEVEL n]
Specifies audio level for BER Data Generator. Range of n is 0 to 4095.

0 corresponds to 0 volts, and 4095 corresponds to 5 volts.

LEVEL?
[FGEN:DATA:LEVEL?]  
Returns current audio level for BER Data Generator.

MODulation: type
[FGEN:DATA:MODulation: type]
Sets Data Generator Modulation to type. Select OFF, AM, FM or PM.

MODulation?
[FGEN:DATA:MODulation?]  
Returns Data Generator Modulation.

MODL n
[FGEN:DATA:MODL n]
Sets Data Generator to n Modulation level. Range of n for AM is 0.0 to 100.0 (%), FM is 0.0 to 25.0 (kHz) and PM is 0.0 to 10.0 (radians).

MODL?
[FGEN:DATA:MODL?]  
Returns Data Modulation level setting.

6-56
FGEN:

DATA:

STATE b
[FGEN:DATA:STATE b]
Sets Data Modulation on if b is 1 or off if b is 0.
Example:
FGEN:DATA:MOD:FM // Sets Data Generator Modulation to FM.
FGEN:DATA:MODL 4 // Sets Data Generator Modulation Level to 4 kHz.
FGEN:DATA:STAT 1 // Enables Data Generator.
FGEN:DATA:MODL? // Returns Data Generator Modulation Level. 4 (kHz) is returned.

EXT:
(External modulation is input through EXT MOD IN Connector.)

LEVEL n
[FGEN:EXT:LEVEL n]
Sets External Modulation Proportional output level to n%. Range of n is 0 to 100.
Sources are adjusted so that proportional output levels of sources combine to equal Function Generator Output Level.

Functions only if Proportional Mode is On.

LEVEL?
[FGEN:EXT:LEVEL?] Returns External Modulation Proportional output level setting in %.

MODL n
[FGEN:EXT:MODL n]
Sets External Modulation level to n. Range of n for AM is 0 to 100 (%), FM is 0.0 to 10.0 (kHz) and PM is 0.0 to 10.0 (radians).

MODL?
[FGEN:EXT:MODL?] Returns External Modulation level setting.

MODulation: type
[FGEN:EXT:MODulation: type] Sets External Modulation to type. Select OFF, AM, FM or PM.

Setting an AF Generator to FM and the other AF Generator to PM, simultaneously, may cause ambiguous readings.

MODulation?
[FGEN:EXT:MODulation?] Returns External Modulation type (OFF, AM, FM or PM).
FGEN:

EXT:

STATe b

[FGEN:EXT:STATe b]
Sets External Modulation on if b is 1 or off if b is 0.
Example:  
FGEN:EXT:MOD:AM  // Sets External (EXT MOD) Modulation Type  
            // to AM.
FGEN:EXT:MODL 50  // Sets External Modulation Level to 50%.
FGEN:PROP 1       // Selects Proportional Mode for the AF  
            // Generator.
FGEN:EXT:LEV 30   // Sets External Output Level to 30%.
FGEN:EXT:MOD?     // Queries External (EXT MOD) Modulation  
            // Type. AM is returned.
FGEN:EXT:MODL?    // Queries External (EXT MOD) Modulation  
            // Level. 50% is returned.
            // 30 is returned.

FSK x

[FGEN:FSK x]
Shifts output (See FGEn:OUTput:) between Generator #1 and Generator #2.  x = 1 selects  
Generator #1.  x = 0 selects Generator #2.
Commands for Generator 1 and Generator 2 are identical and are listed as **FGEN:GENx**. Parameter $x$ is specified as 1 or 2 when the command is entered.

**GENx:**

**FREQuency** $f$

[FGEN:GENx:FREQuency $f$]

Sets AF Generator to $f$ Hz. Range of $f$ is 0.0 to 40000.0.

**FREQuency?**

[FGEN:GENx:FREQuency?]

Returns AF Generator in Hz.

**LEVEL** $n$

[FGEN:GENx:LEVEL $n$]

Sets Proportional output level of AF Generator to $n\%$. Range of $n$ is 0 to 100. Sources are adjusted so that proportional output levels of sources combine to equal Function Generator Output Level.

**Functions only if Proportional Mode is On.**

**LEVEL?**

[FGEN:GENx:LEVEL?]

Returns Proportional output level setting of AF Generator in %.

**MODL** $n$

[FGEN:GENx:MODL $n$]

Sets Modulation level to $n$. Range of $n$ for AM is 0 to 100 ($\%$), FM is 0.0 to 100.0 (kHz) and PM is 0.0 to 10.0 (radians).

**MODL?**

[FGEN:GENx:MODL?]

Returns AF Generator Modulation level setting.

**MODulation:**$type$

[FGEN:GENx:MODulation:$type$]

Sets AF Generator Modulation to $type$. Select OFF, AM, FM or PM for $type$.

**Setting an AF Generator to FM and the other AF Generator to PM, simultaneously, may cause ambiguous readings.**

**MODulation?**

[FGEN:GENx:MODulation?]

Returns AF Generator Modulation Type.

**SHAPE:**$type$

[FGEN:GENx:SHAPE:$type$]

Sets AF Generator wave shape to $type$. Select from SIN (Sine), SQU (Square Wave), RAMP or TRI (Triangle).

**SHAPE?**

[FGEN:GENx:SHAPE?]

Returns AF Generator wave shape.
FGEN:

GENx:

SHAPE:

DC n
[FGEN:GENx:SHAPE:DC n]
Sets wave shape to DC at n level. Select -1, 0 or 1 for n.

PULse:DCYCLE 50
[FGEN:GENx:SHAPE:PULse:DCYCLE 50]
Sets AF Generator wave shape to Pulse with 50% duty cycle.

STATe b
[FGEN:GENx:STATe b]
AF Generator is enabled if b is 1, disabled if b is 0.

Example:

// AF Generator setup as RF Generator
// Modulation Source (AF Generator 1 and 2).
FGEN:GEN1:FREQ 1000  // Sets AF Generator 1 frequency to 1000 Hz.
FGEN:GEN1:MOD:FM    // Selects FM for AF Generator 1 Modulation
                     // Type.
FGEN:GEN1:MODL 5    // Sets AF Generator 1 Modulation Level to
                     // 5 kHz.
FGEN:GEN1:SHAPE:SIN // Selects a sine wave for AF Generator 1
                     // Wave Form.
FGEN:GEN1:STAT 1    // Enables AF Generator 1.
FGEN:GEN2:FREQ 2500  // Sets AF Generator 2 frequency to 2500 Hz.
FGEN:GEN2:MOD:AM    // Selects AM for AF Generator 2 Modulation
                     // Type.
FGEN:GEN2:MODL 25   // Sets AF Generator 2 Modulation Level to
                     // 25%.
FGEN:GEN2:SHAPE:SIN // Selects a sine wave for AF Generator 2
                     // Wave Form.
FGEN:GEN2:STAT 1    // Enables AF Generator 2.

Example:

// AF Generator setup as AF Signal Generator
FGEN:GEN1:FREQ 1000  // Sets AF Generator 1 frequency to 1000 Hz.
FGEN:OUT:PROP 1     // Enables AF Generator Proportional Mode.
FGEN:GEN1:LEV 50    // Sets AF Generator 1 Proportion Level to
                     // 50%.
FGEN:GEN1:SHAPE:SIN // Selects a sine wave for AF Generator 1
                     // Wave Form.
FGEN:GEN1:STAT 1    // Enables AF Generator 1.
FGEN:GEN2:FREQ 2500  // Sets AF Generator 2 frequency to 2500 Hz.
FGEN:GEN2:LEV 100   // Sets AF Generator 2 Proportion Level to
                     // 100%.
FGEN:GEN2:SHAPE:SIN // Selects a sine wave for AF Generator 2
                     // Wave Form.
FGEN:GEN2:STAT 1    // Enables AF Generator 2.
DIGital type
[FGEN:GEN3:DIGital type]
Sets digital type. Used with the FGEN:ENCodE DIGital command to prepare Generator 3 encoding. The setting for type is DCS, DCSINV, POCSAG, DSAT or DST.

ENCodE type
[FGEN:GEN3:ENCodE type]
Selects a signalling format to encode. Select DTMF, TONE, DIGital or RCC for type. Must be followed with a SETUP:GEN or SCREEN:GEN command.

MODL n
[FGEN:GEN3:MODL n]
Sets AF Generator 3 Modulation level. Range of n for AM is 0 to 100.0 (‰), PM is 0.0 to 10.0 (radians). FM is 0.0 to 100.0 (kHz). Tone or RCC signalling format, FM is 0.0 to 10.0 (kHz). DTMF signalling format or FM is 0.0 to 25.0 (kHz). Digital signalling format.

MODL?
[FGEN:GEN3:MODL?] Returns AF Generator 3 Modulation level setting.

MODulatIon: type
[FGEN:GEN3:MODulatIon: type]
Sets AF Generator 3 Modulation to type. Select OFF, AM, FM or PM.

Setting an AF Generator to FM and the other AF Generator to PM, simultaneously, may cause ambiguous readings.

MODulatIon?
[FGEN:GEN3:MODulatIon?] Returns AF Generator 3 Modulation type.


RCC format
[FGEN:GEN3:RCC format]
Sets Generator RCC signaling format. Select IMTS, MTS, SYS2805 or TREMote for format.

MIC:
(External audio modulation is input through MIC/ACC Connector.)

LEVel n
[FGEN:MIC:LEVel n]
Sets Proportional output level of MIC/ACC Connector input to n%. Range of n is 0 to 100. Sources are adjusted so that proportional output levels of sources combine to equal Function Generator Output Level. Functions only if Proportional Mode is On.

LEVel?
[FGEN:MIC:LEVel?] Returns Proportional output level setting for MIC/ACC Connector input.
FGEN:

MIC:

MODL n
[FGEN:MIC:MODL n]
Sets MIC/ACC Connector input Modulation level to n. Range of n for AM is 0 to 100 (%), FM is 0.0 to 25.0 (kHz) and PM is 0.0 to 10.0 (radians).

MODL?
[FGEN:MIC:MODL?]
Returns MIC/ACC Connector input Modulation level setting.

MODulation: type
[FGEN:MIC:MODulation: type]
Sets MIC/ACC Connector input Modulation to type. Select OFF, AM, FM or PM.

Setting an AF Generator to FM and the other AF Generator to PM, simultaneously, may cause ambiguous readings.

MODulation?
[FGEN:MIC:MODulation?]
Returns MIC/ACC Connector input Modulation type.

STATE b
[FGEN:MIC:STATE b]
Enables external modulation through MIC/ACC Connector if b is 1, disables if b is 0.

PTT:STATe b
[PTT:STATE b]
Sets the push to talk pin on the MIC/ACC Connector high (keys the mic) if b is 1, sets the push to talk pin low if b is 0.

Example:
FGEN:MIC:MOD:AM // Sets External (MIC/ACC) Modulation Type
// to AM.
FGEN:MIC:MODL 25 // Sets External (MIC/ACC) Modulation Level
// to 25%.
FGEN:OUT:PROP 1 // Selects Proportional Mode for the AF
// Generator.
FGEN:MIC:LEV 50 // Sets External (MIC/ACC) Output Level
// to 50%.
FGEN:MIC:MOD? // Queries External (MIC/ACC) Modulation Type. AM is returned.
FGEN:MIC:MODL? // Queries External (MIC/ACC) Modulation Level. 25% is returned.
FGEN:MIC:LEV? // Queries External (MIC/ACC) Output Level. // 50 is returned.
FGEN:

OUTput:

AUDio b
[FGEN:OUTput:AUDio b]
Routes AF Generator Output to the AUDIO OUT Connector if b is 1, disconnects AUDIO OUT Connector from Output if b is 0.

AUDio?
[FGEN:OUTput:AUDio?]
Returns 1 if AF Generator Output is routed to the AUDIO OUT Connector, 0 if AUDIO OUT Connector is disconnected.

DEMOD b
[FGEN:OUTput:DEMOD b]
Routes AF Generator Output to DEMOD OUT Connector if b is 1, disconnects DEMOD OUT Connector from Output if b is 0.

DEMOD?
[FGEN:OUTput:DEMOD?]
Returns 1 if AF Generator Output is routed to DEMOD OUT Connector, 0 if DEMOD OUT Connector is disconnected.

LEVel v
[FGEN:OUTput:LEVel v]
Sets AF Generator output level to v volts. Range for v is 0.0000 to 3.1000.

LEVel?
[FGEN:OUTput:LEVel?]
Returns AF Generator output level in volts.

SPEAKer b
[FGEN:OUTput:SPEAKer b]
Routes AF Generator to Speaker if b is 1, disconnects Speaker from Output if b is 0.

SPEAKer?
[FGEN:OUTput:SPEAKer?]
Returns 1 if AF Generator Output is routed to Speaker, 0 if Speaker is disconnected.
FGEN:

**PROPportional b**
([FGEN:PROPportional b])
Selects Proportional mode for AF Generator if b is 1. Takes AF Generator out of Proportional mode if b is 0.

**PROPportional?**
([FGEN:PROPportional?])
Returns 1 if AF Generator is in Proportional mode; 0 otherwise.

Example:

```
FGEN:GEN1:FREQ 500  // Sets AF Generator 1 frequency to 500 Hz.
FGEN:OUT:PROP 1     // Enables Proportional Mode of AF Generator.
FGEN:GEN1:LEV 100    // Sets AF Generator 1 Proportion Level to
                     // 100%.
FGEN:GEN1:SHAPE:RAMP // Selects ramp for the AF Generator 1
                     // Wave Form.
FGEN:GEN1:STAT 1     // Enables AF Generator 1.
FGEN:GEN2:LEV 100     // Sets AF Generator 2 Proportion Level to
                     // 100%.
FGEN:GEN2:SHAPE:SIN   // Selects a sine wave for AF Generator 2
                     // Wave Form.
FGEN:GEN2:STAT 1     // Enables AF Generator 2.
FGEN:OUT:LEV 1.5     // Sets AF Generator Output Level to 1.5 V.
FGEN:OUT:AUDIO 1     // Routes AF Generator Output to AUDIO OUT
                     // Connector.
```

**RCL n**
([FGEN:RCL n])
Recalls AF Generator environment (routings and settings) stored in memory location n.
Range of n is 1 to 9.

**STORE n**
([FGEN:STORE n])
Stores current AF Generator environment (routings and settings) in memory location n.
Range of n is 1 to 9.
6-8-2 REMOTE AF GENERATOR EXAMPLES

The following command sequence generates a 1 kHz sine wave, with 1 V level, and routes the Output to the DEMOD OUT Connector:

SCREEN:FUNC // Displays the AF Generator Operation Screen.
FGEN:GEN1:STATE 1 // Enables AF Generator 1.
FGEN:GEN1:FREQ 1000 // Sets AF Generator 1 frequency to 1000 Hz.
FGEN:GEN1:SHAPE:SIN // Selects a sine wave for the AF Generator 1 wave form.
FGEN:LEVEL 1 // Sets AF Generator 1 Output Level to 1 V.
FGEN:OUTPUT:DEMOD 1 // Routes AF Generator Output to the DEMOD OUT Connector.

The following command sequence generates the sum of a 1 kHz and 2.5 kHz sine wave in equal proportions and routes this Output to the AUDIO OUT Connector:

SCREEN:FUNC // Displays the AF Generator Operation Screen.
FGEN:PROPORTIONAL 1 // Selects Proportional Mode for the AF Generator.
FGEN:OUTPUT:LEVEL 1 // Sets AF Generator 1 Output Level to 1 V.
FGEN:GEN1:STATE 1 // Enables AF Generator 1.
FGEN:GEN1:FREQ 1000 // Sets AF Generator 1 frequency to 1000 Hz.
FGEN:GEN1:SHAPE:SIN // Selects a sine wave for the AF Generator 1 wave form.
FGEN:GEN1:LEVEL 100 // Sets AF Generator 1 Proportional Level to 100%.
FGEN:GEN2:STATE 1 // Enables AF Generator 2.
FGEN:GEN2:FREQ 2500 // Sets AF Generator 2 frequency to 2500 Hz.
FGEN:GEN2:SHAPE:SIN // Selects a sine wave for the AF Generator 2 wave form.
GEN:GEN2:LEVEL 100 // Sets AF Generator 2 Proportional Level to 100%.
FGEN:OUTPUT:AUDIO 1 // Routes AF Generator Output to the AUDIO OUT Connector.
6-9 OSCILLOSCOPE

6-9-1 OSCILLOSCOPE COMMANDS

SCOPe:

ARM
[SCOpe:ARM]
Arms Oscilloscope. Sets Oscilloscope for one sweep. Command ignored unless Trigger is set to One Shot (see SCOpe:TRIGger commands).

AVERAGE n
[SCOpe:AVERAGE n]
Selects Oscilloscope Average Mode using n samples. Range of n is 1 to 100 (default is 100).

COMPARE n
[SCOpe:COMPARE n]
Selects Compare Mode for Oscilloscope. Trace stored at n memory location is compared to the current Live Trace.

Example: SCOP:COMP 2 // Displays the Trace stored in memory location 2 along with the current live Trace. All Oscilloscope settings affect the live Trace only.

COUPLING type
[SCOpe:COUPLing type]
Selects type for External coupling. Select AC, DC or GROUND.

FULL
[SCOpe:FULL]
Selects a full size Oscilloscope display for the RF Generator, Receiver and Duplex Operation Screens.

HORIZONTAL n
[SCOpe:HORIZONTAL n]
Sets Horizontal Time Offset to n major divisions. n is an integer from -12 to 12. -12 to -1 are major divisions before the trigger. 1 to 12 are major divisions after the trigger.

INPut:

FILTER:

CWEight:STATE b
[SCOpe:INPut:FILTER:CWEight:STATE b]
Enables Internal C-Weight filter if b is 1, disables if b is 0.

HPAs:

FREQuency f
[SCOpe:INPut:FILTER:HPAs:FREQuency f]
Sets Internal High-Pass frequency to f kHz. Range of f is 0.2 to 100.

STATE b
[SCOpe:INPut:FILTER:HPAs:STATE b]
Enables Internal High-Pass filter if b is 1, disables if b is 0.
SCOPe:

INPut:

FILTER:

LPASs:

FREQuency f
[SCOPE:INPut:FILTER:LPASs:FREQuency f]
Sets Internal Low-Pass frequency to f kHz. Range of f is 0.2 to 50.

STATE b
[SCOPE:INPut:FILTER:LPASs:STATE b]
Enables Internal Low-Pass filter if b is 1, disables if b is 0.

NOTch:

FREQuency f
[SCOPE:INPut:FILTER:NOTch:FREQuency f]
Sets Notch center frequency to f kHz. Range of f is 0.5 to 1.5.

STATE b
[SCOPE:INPut:FILTER:NOTch:STATE b]
Enables Internal Notch Filter if b is 1, disables if b is 0.

Example:
SCOP:INP:FILT:NOT:FREQ 1.5 // Sets Oscilloscope Input Notch Filter center frequency to 1.5 kHz.

INTernal type
[SCOPE:INTernal type]
Selects type as Internal Oscilloscope Input. Select from: IF (Rcvr IF), DEMOD (Demod Audio), POWER (RF Pwr Lvl), SINAD (SINAD/BER), FUNCTION (Func Gen) or XAUDIO (Ext Mod).

LEVEL n
[SCOPE:LEVEL n]
Sets Trigger level to n. Range of n is 0 to 255 with 0 corresponding to the bottom of the Oscilloscope Display and 255 corresponding to the top.

LIVE b
[SCOPE:LIVE b]
Selects Live Trace Mode for the Oscilloscope. If b is 1, a SCREEN:SCOPe command is performed. If b is 0, no SCREEN:SCOPe command is performed. Default of optional b is 1.
SCOPe:

MARKer:

AOFF
[SCOPe:MARKer:AOFF]
Disables both Markers.

DELTA:

AMPLitude?
[SCOPe:MARKer:DELTA:AMPLitude?]
Returns voltage difference of the Trace Marker 1 and Trace Marker 2 crossings in volts. Valid only for Oscilloscope Inputs AC, DC and GND.

POINT?
[SCOPe:MARKer:DELTA:POINT?]
Returns the difference of Marker positions in graticules with 100 graticules equal to the Oscilloscope display width.

TIME?
[SCOPe:MARKer:DELTA:TIME?]
Returns the difference of the two Marker positions in ms.

TRACK b
[SCOPe:MARKer:TRACK b]
Enables Marker Tracking if b is 1, disables Marker Tracking if b is 0. Tracking feature keeps Markers a constant distance apart.

Commands for Marker 1 and Marker 2 are identical and are listed as SCOPe:MARKER$\text{x}$ commands. Parameter $x$ is specified as 1 or 2 when the command is entered.

SCOPe:

MARKERx:

AMPLitude?
[SCOPe:MARKERx:AMPLitude?]
Returns voltage of live Trace at Marker $x$ crossing in volts. Valid only for Oscilloscope Inputs AC, DC and GND.

POINT n
[SCOPe:MARKERx:POINT n]
Sets Marker $x$ position to $n$ graticules. Range of $n$ is 1 to 100 with 100 graticules equal to Oscilloscope display width.

POINT?
[SCOPe:MARKERx:POINT?]
Returns Marker $x$ position in graticules.
SCOPe:

MARKERx:

STATe b
[SCOPe:MARKERx:STATe b]
Enables Marker x if b is 1, disables Marker x if b is 0.

Both Markers are displayed when one or both are active.

STATe?
[SCOPe:MARKERx:STATe?]
Returns 1 if Marker x is active, 0 if Marker x is not active.

TIME?
[SCOPe:MARKERx:TIME?]
Returns Marker x position in ms from left edge of Oscilloscope display.

Example:

SCOP:MARKER1:STAT 1 // Enables Marker 1.
SCOP:MARKER2:STAT 1 // Enables Marker 2.
SCOP:MARKER1:POIN 20 // Positions Marker 1 two major divisions
                      // from left edge of display screen.
SCOP:MARKER2:POIN 40 // Positions Marker 2 four major divisions
                      // from left edge of display screen.
SCOP:MARK:TRACK 1   // Enables Marker Tracking.
SCOP:MARKER1:POIN 30 // Positions Marker 1 three divisions from
                      // left edge of display screen
                      // (and Marker 2 five divisions from
                      // left edge of display screen).
SCOP:MARK:DELTA:AMPL? // Queries amplitude difference of Trace
                        // Marker crossings in volts.
SCOP:MARK:DELTA:POIN? // Queries position difference of markers
                        // in graticules.
SCOP:MARK:DELTA:TIME? // Queries position difference of markers
                        // in ms.
SCOP:MARK:TRACK 0    // Disables Marker Tracking.
SCOP:MARKER2:AMPL?   // Queries the amplitude of the Trace at
                      // Marker 2 position.
                      // 50 is returned.
SCOP:MARK:AOFF      // Disables both Markers.
SCOPE:

PLOT:

The SCOPE:PLOT:xxx commands send plotter data of the Oscilloscope grid, trace or units out the HOST RS-232 or GPIB Connector. SYSTem:PLOT:GPIB or SYSTem:PLOT:SERial command must be sent to change plotter output connector.

GRID

[SCOPE:PLOT:GRID]
Draws Oscilloscope grid on attached plotter.

TRACE

[SCOPE:PLOT:TRACE]
Draws Oscilloscope trace on attached plotter.

UNITS

[SCOPE:PLOT:UNITS]
Draws Oscilloscope units on attached plotter.

QTR

[SCOPE:QTR]
Selects a 1/4 size Oscilloscope display for the RF Generator, Receiver and Duplex Operation Screens.

RCL n

[SCOPE:RCL n]
Recalls Oscilloscope Trace and environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

Recalled Trace reflects the stored Trace parameters. Changing current settings does not alter the recalled Trace.

Example: SCOP:RCL 4 // Recalls Oscilloscope trace and environment stored in memory location 4.

SCOP:SCAL 500 // Change current scale to 500 mV/div.

// Recalled Trace does not change position.

SCALE n

[SCOPE:SCALE n]
Sets Oscilloscope scale to n mV/div for AC, DC or GND Input. Select from:

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>2000</td>
<td>5000</td>
<td></td>
</tr>
<tr>
<td>10000</td>
<td>20000</td>
<td>50000</td>
<td></td>
</tr>
</tbody>
</table>

Sets Oscilloscope scale to n kHz/div for Demod Audio Input with FM. Select from:

<table>
<thead>
<tr>
<th>n</th>
<th>2</th>
<th>4</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sets Oscilloscope scale to n mV/div for Func Gen or Ext Mod Input. Select from:

<table>
<thead>
<tr>
<th>n</th>
<th>500</th>
<th>1000</th>
<th>2500</th>
</tr>
</thead>
</table>
SCOPe:

SCAl e?
[SCOPe:SCAl e?]
Returns Oscilloscope scale in mV/div if input is AC, DC, GND, SINAD/BER, Func Gen or Ext Mod. Returns Oscilloscope scale in kHz/div if input is Demod Audio. Returns Oscilloscope scale in W if input is RF Pwr Lvl.

SOURce EXTernal
[SCOPe:SOURce EXTernal]
Routes Oscilloscope Input from the SCOPE IN Connector. Selects AC Ext for Oscilloscope Input.

SOURce INTernal
[SCOPe:SOURce INTernal]
 Disconnects Oscilloscope Input from the SCOPE IN Connector. Selects Demod Audio for Oscilloscope Input.

STATe b
[SCOPe:STATe b]
Displays an Oscilloscope display in the RF Generator, Receiver and Duplex Operation Screens if b is 1. Oscilloscope display is not displayed if b is 0.

STORe n
[SCOPe:STORe n]
Stores current Oscilloscope Trace and environment (routings and settings) in memory location n. Range of n is 1 to 9.

The SCOPe:SWEep commands may be represented by the command: SCOPe:SWEep n [units]. For clarification, the commands are described separately.

SWEep n
[SCOPe:SWEep n]
Sets Oscilloscope sweep rate to n µs/div. Select from:

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>20</th>
<th>100</th>
<th>200</th>
<th>1000</th>
<th>2000</th>
<th>10000</th>
<th>20000</th>
<th>100000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>20</td>
<td>50</td>
<td>500</td>
<td>5000</td>
<td>5000</td>
<td>50000</td>
<td>50000</td>
<td>50000</td>
</tr>
</tbody>
</table>

SWEep n MS
[SCOPe:SWEep n MS]
Sets Oscilloscope sweep rate to n ms/div. Select from:

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>10</th>
<th>20</th>
<th>100</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>20</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>
SCOPe:

SWEep n US
[SCOPe:SWEep n US]
Sets Oscilloscope sweep rate to n μs/div. Select from:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

SWEep?
[SCOPe:SWEep?]
Returns Oscilloscope sweep rate in μs/div.

Example:  SCOP:INT FUNC  // Selects AF Generator as the Oscilloscope Input.
          SCOP:LIV       // Selects Live for Oscilloscope Mode.
          SCOP:TRIG:AUTO // Selects Auto for Oscilloscope Trigger Mode.
For the following SCOPe:TRACE commands, the Oscilloscope display is divided into 400 positions horizontally (0 signifying the left edge of the display, 399 signifying the right edge of the display) and 256 values vertically (0 signifying the bottom of the display, 255 signifying the top of the display):

**SCOPe:**

**TRACE:**

```
DATA n, offset, value, value, ..., value
[SCOPe:TRACE:DATA n, offset, value, value, ..., value]
```

Replaces points of a stored Trace n with specified vertical values (value) starting with horizontal offset (offset). offset is given in number of positions from the left edge of the display. Range of n is 1 to 9. Multiple values are separated by commas. One value is displayed for each succeeding horizontal position of the Trace starting with offset. Intended for remote GPIB or RS-232 use only.

```
DATA? [[[n], offset], points]
[SCOPe:TRACE:DATA? [[[n], offset], points]]
```

Sends the specified number of data points of Trace n to the Host. Trace n can be the live Trace or a Trace stored in memory. The affected points start with the optional offset with an offset default of 0. Range of offset is 0 to 399. Range of n is 0 to 9 with 0 signifying the Live Trace (0 default). Range of the number of points is 1 to 400 with a default of 400. Intended for remote GPIB or RS-232 use only.

```
GET name, n
[SCOPe:TRACE:GET name, n]
```

Assigns values of trace n (in graticules) to a declared array. Parameter name is the name of the declared array. Range of n is 0 to 9, 0 signifying the live trace and 1 to 9 signifying stored traces. If array is less than 400 values in length (stored trace length), the array values are assigned for the length of the array, leaving the rest of the stored trace intact.

```
GET? n, offset
[SCOPe:TRACE:GET? n, offset]
```

Returns value of a point in Trace n located the offset number of positions from the left edge of the Display. Trace n can be the live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace. Range of offset is 0 to 399.

```
MAX? [[[n], offset], points]
[SCOPe:TRACE:MAX? [[[n], offset], points]]
```

Returns the maximum point of Trace n within the specified number of points starting with the given offset. Result is returned in x, y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace (0 default). Range of the optional offset is 0 to 399 with a default of 0. Range of optional number of points is 1 to 400 with a default of 400. Intended for remote GPIB or RS-232 use only.
SCOPe:

TRACE:

MIN? [[[n],offset],points]
[SCOPe:TRACE:MIN? [[[n],offset],points]]
Returns the minimum point of Trace n within the specified number of points starting with the given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the Live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace (0 default). Range of the optional offset is 0 to 399 with a default of 0. Range of optional number of points is 1 to 400 with a default of 400. Intended for remote GPIB or RS-232 use only.

PUT name,n
[SCOPe:TRACE:PUT name,n]
Assigns values of an array to trace n (in graticules). Parameter name is the array name. Range of n is 1 to 9, signifying stored traces. If array is less than 400 values in length (stored trace length), the array values are assigned for the length of the array, leaving the rest of the stored trace intact.
Example: VAR POINTS[199]
SCOP:TRACE:DATA 3,40,128 // Changes a single Trace value.
// located one division from the // left edge, to the midway point // (on the axis).
SCOP:TRACE:GET? 0,200 // Queries the value of the live // Trace at the point 200 (center // of oscilloscope display).
POIANTS=SCOP:TRACE:DATA? 3,200,200 // Returns the Trace values // of the right half of the // Trace stored in memory // location 3.
SCOP:TRACE:MAX? 0,200 // Queries the maximum Trace value // of the left half of the live // Trace.
SCOP:TRACE:MIN? 0,40,80 // Queries the minimum Trace value // of the live Trace for the second // and third divisions.

TRIGger:

AUTO
[SCOPe:TRIGger:AUTO]
Sets Trigger to Auto Sweep.

IMMediate
[SCOPe:TRIGger:IMMediate]
Triggers Oscilloscope as soon as command is interpreted or sequenced in a macro.

NORM
[SCOPe:TRIGger:NORM]
Sets Trigger to Normal Sweep.
**SCOPe:**

**TRIGger:**

**ONE**

[SCOPe:TRIGger:ONE]

Sets Trigger to One Shot. Causes oscilloscope to sweep once when trigger is received and after setting arm function (SCOPe:ARM command).

**SOURce type**

[SCOPe:TRIGger:SOURce type]

Selects source to trigger on. Select EXTernal (for external trigger, AC or DC Input only), INTernal (for internal trigger, AC or DC Input only) or BUS (triggered by *TRG or IEEE-488.1 GET [Get Execute Trigger] command).

**VERTical n**

[SCOPe:VERTical n]

Sets Vertical Offset voltage to n graticules. Range of n is 0 to 255. Setting is not linear. For AC, DC or GND, middle setting is approximately 150 to 170. For other Inputs, middle setting is approximately 160 to 180.
6-9-2 REMOTE OSCILLOSCOPE EXAMPLES

The following example is a macro that centers a trace in the middle of the color display. If the trace centered has AC, DC or GND input, GND input should be selected before executing this macro. If the trace centered has other input, RF Pwr Lvl Input should be selected before executing this macro. The Oscilloscope Operation Screen must be displayed for this macro to operate.

*DMC 'Center',BEGIN // Defines macro names Center.
VAR HOLD, CENTER_LINE, X // Declares variables needed.
CENTER_LINE = 0 // Initializes Center_line variable.
FOR X = 0 TO 255 // Loops for every possible vertical trace
  // position.
  SCOPE:VERT X // Moves trace to vertical position x.
  DELAY 50 // Pause allows SCOPE:VERT command to finish.
  HOLD = SCOP:TRACE:GET? 0, 200 // Trace's actual vertical
    // position is stored in HOLD.
  IF (HOLD>=128-3) AND (HOLD<=128+3) // If HOLD is close to center
    // of color display,
    // store x value into
    // CENTER_LINE.
    CENTER_LINE = X
  ENDIF // End of IF statement.
NEXT X // End of FOR loop.
SCOPE:VERT CENTER_LINE // Trace is moved to point previously found
  // that centers trace.
END // End of macro Center.

The following example is a macro that finds the maximum of the live trace starting from a given offset. Range of the offset is 0 to 400. This macro returns the horizontal position of the maximum point. If there are multiple maximum points, the first one is returned. With the RETURN command included, this macro is executed as a function (ex. X = FINDMAX 0).

*DMC "FINDMAX", BEGIN // Define macro named FINDMAX
VAR NEW, OLD=0, OLD_X=0 // Declare and initialize variables: NEW hold
  // highest amplitude reading, OLD holds
  // highest amplitude reading yet found and
  // OLD_X holds position of OLD.
FOR X=51 TO 400 // Loop from offset given to end of color display.
  NEW=SCOP:TRACE:GET? 0, X // Read l point and store in NEW.
  IF NEW > OLD+3 // If current amplitude is greater than last
    // max,
    OLD = NEW
    OLD_X = X
    // store new max into old max and,
    // store new max location into old max
    // location.
  ENDIF // End of IF statement.
NEXT X // End of looping.
RETURN OLD_X // Return location of maximum value found.
END // End of FINDMAX macro.
The following command sequence compares a stored Trace (1 kHz sine wave) with a live Trace (2 kHz square wave). This example assumes the previous macro "Center" already resides in Test Set memory.

// AF Generator is setup to create first signal.
FGEN:OUTPUT:LEVEL 1 // Sets AF Generator Output Level to 1 V.
FGEN:GEN1:STATE 1 // Enables AF Generator 1.
FGEN:GEN1:FREQ 1000 // Sets AF Generator 1 frequency to 1000 Hz.
FGEN:GEN1:SHAPE:SIN // Selects a sine wave for AF Generator 1 Wave Form.

// Oscilloscope parameters are set and first Trace is stored.
SCREEN:SCOP // Displays Oscilloscope Operation Screen.
SCOPE:SOURCE INTERNAL // Sets Oscilloscope to be triggered internally.
SCOPE:TRIGGER:NORM // Selects Normal as Oscilloscope Trigger Mode.

// Sets Trigger Level to the midpoint.
SCOPE:LIVE // Sets Oscilloscope Mode to live.
SCOPE:INTERNAL POWER Center // Executes macro Center which centers trace on display screen (see example above).
SCOPE:INTERNAL FUNC // Selects AF Generator as Oscilloscope Input.
SCOPE:SCALE 1000 // Sets Scale to 1 volt/div.
SCOPE:Sweep 200 // Sets Sweep rate to 200 μV/div.
SCOPE:STORE 1 // Stores Trace in memory location 1.

// Second signal is created.
FGEN:GEN1:FREQ 2000 // Changes AF Generator 1 frequency to 2 kHz.
FGEN:GEN1:SHAPE:SQU // Changes AF Generator 1 Wave Shape to a square wave.

// The first and second signals are compared.
SCOPE:COMPARE 1 // Selects Compare mode for Oscilloscope.
// Trace stored in memory location 1 is displayed on the Oscilloscope display with the current live Trace.
The following command sequence displays the sum of a 1 kHz and 2.5 kHz sine wave on the Oscilloscope display and returns the period of the signals. This example assumes the macro FINDMAX (see second example) is already loaded into Test Set memory.

```
// Setup AF Generator 1 and 2 to generate the signal to be measured.
FGEN:PROPORTIONAL 1  // Enable AP Generator Proportional Mode.
FGEN:OUTPUT:LEVEL 1   // Set AF Generator Output Level to 1 V.
FGEN:GEN1:STATE 1     // Enable AP Generator 1.
FGEN:GEN1:FREQ 1000    // Set AF Generator 1 Frequency to 1000 Hz.
FGEN:GEN1:SHAPE:SIN    // Select a sine wave for the AF
                        // Generator 1 Wave Form.
FGEN:GEN1:LEVEL 100    // Set AF Generator 1 Proportional Level to 100%.
FGEN:GEN2:STATE 1     // Enable AP Generator 2.
FGEN:GEN2:FREQ 2500    // Set AF Generator 2 Frequency to 2500 Hz.
FGEN:GEN2:SHAPE:SIN    // Select a sine wave for the AF
                        // Generator 1 Wave Form.
FGEN:GEN2:LEVEL 100    // Set AF Generator 2 Proportional Level to 100%.

// MEAS_PER macro measures the period of the AF Generator signal.
*DME "MEAS_PER",BEGIN
VAR MAX1,MAX2

SCREEN:SCOP
SCOPE:SOURCE INTERNAL
SCOPE:INTERNAL FUNC
SCOPE:LIVE
SCOPE:SCALE 500
SCOPE:SWEEP 500

SCOPE:TRIGGER:ONE
// Select One Shot Trigger Mode. Hold trace still.
DELAY 600
SCOPE:ARM
// Allow time to set for arming function.
MAX1 = FINDMAX 0
// MAX1 is set to first maximum using macro of second example.
MAX2 = FINDMAX (MAX1+5)
// MAX2 is set to second maximum using macro of second example.
SCOPE:MARKER1:STATE 1
// Enables Marker 1.
SCOPE:MARKER1:POINT FLOOR(MAX1/4)
// Position Marker 1 on the first maximum.
SCOPE:MARKER2:STATE 1
// Enables Marker 2.
SCOPE:MARKER2:POINT FLOOR(MAX2/4)
// Positions Marker 2 on the second maximum.
T=SCOPE:MARKER:DELTA:TIME?
// Query the difference in Marker positions in ms.
PPRINT "PERIOD IS ",1000/T
// Figure and print period to Host.
END
```

6-78
6-10  SPECTRUM ANALYZER

6-10-1  SPECTRUM ANALYZER COMMANDS

ANLZ:

AVErage [n]
[ANLZ:AVErage [n]]
Selects Average Mode for Analyzer using n samples. Range of n is 1 to 100. Default is 100.

CHANnel n
[ANLZ:CHANnel n]
Sets RF Frequency to cellular channel n (1 to 2047, depending on format) in the format selected using the ANLZ:CHANnel:FORMAT command. The Spectrum Analyzer Operation Screen displays selected cellular channel when Analyzer is in Channel Mode (ANLZ:MODE CHAN command).

CHANnel:

FORMat:

AMPS:

FORward
[ANLZ:CHANnel:FORMat:AMPS:FORward]
Selects AMPS (NADC-U8) Forward channels.

REVerse
[ANLZ:CHANnel:FORMat:AMPS:REVerse]
Selects AMPS (NADC-U8) Reverse channels.

ETACS:

FORward
[ANLZ:CHANnel:FORMat:ETACS:FORward]
Selects ETACS Forward channels.

REVerse
[ANLZ:CHANnel:FORMat:ETACS:REVerse]
Selects ETACS Reverse channels.

NADC:

FORward
[ANLZ:CHANnel:FORMat:NADC:FORward]
Selects NADC Forward channels.

REVerse
[ANLZ:CHANnel:FORMat:NADC:REVerse]
Selects NADC Forward channels.

BAND:xx
[ANLZ:CHANnel:FORMat:NADC:BAND:x]
Selects NADC band. Range of x is U8, U4 or HYper.
ANLZ:

CHANnel:

FORMAT:

NAMPS:

FORward
[ANLZ:CHANnel:FORMAT:NAMPS:FORward]
Selects NAMPS Forward channels.

REVere
[ANLZ:CHANnel:FORMAT:NAMPS:REVerse]
Selects NAMPS Reverse channels.

BAND:x
[ANLZ:CHANnel:FORMAT:NAMPS:BAND:x]
Selects NAMPS Narrow Analog channel designator. Range of x is Lower, Middle or Upper.

NT400:

FORward
[ANLZ:CHANnel:FORMAT:NT400:FORward]
Selects NT400 Forward channels.

REVere
[ANLZ:CHANnel:FORMAT:NT400:REVerse]
Selects NT400 Reverse channels.

BAND?
[ANLZ:CHANnel:BAND?]
Returns band for Analyzer Channel Format. Returns one of the following bands for the associated channel format:

<table>
<thead>
<tr>
<th>FORMAT</th>
<th>BAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>NADC</td>
<td>U8, U4 or HY</td>
</tr>
<tr>
<td>ETACS</td>
<td>NOT AVAILABLE</td>
</tr>
<tr>
<td>NAMPS</td>
<td>LOWER, MIDDLE or UPPER</td>
</tr>
</tbody>
</table>

FORMAT?
[ANLZ:CHANnel:FORMAT?]
ANLZ:

COMPar e \( n \)
\[[ANLZ:COMPar e \; n]\]
Selects Compare Mode for Analyzer. Trace stored at \( n \) memory location is compared with Live Trace. Range of \( n \) is 1 to 9.
Example:  \( ANLZ:COM P \; 1 \)  // Selects Compare mode for Analyzer.  // Trace stored in memory location 1 is  // displayed on the Analyzer display  // with the current live Trace.

FIND:

FREQuency?
\[[ANLZ:FIND:FREQuency?]\]
Returns frequency of first signal with amplitude larger than Find reference level. Returns 0 if no signal is found.

REFERence \( n \)
\[[ANLZ:FIND:REFERence \; n]\]
Sets Find reference level to \( n \) dB.

REFERence?
\[[ANLZ:FIND:REFERence?]\]
Returns Find reference level in dB.
Example:  \( ANLZ:FIND:REF \; -65 \)  // Sets Find Reference Level to -65 dBm.  \( ANLZ:FIND:FREQ? \)  // Returns the lowest frequency (in kHz)  // containing a signal greater than -65 dBm.  \( ANLZ:FIND:REF? \)  // Queries the current Find Reference Level  // (in dBm). -65 is returned.

FREQuency \( f \)
\[[ANLZ:FREQuency \; f]\]
Sets Analyzer Frequency to \( f \) kHz. Range for \( f \) is 250.0 to 2010000.0.

FREQuency?
\[[ANLZ:FREQuency?]\]
Returns Analyzer Frequency in kHz.
Example:  \( ANLZ:FREQ \; 135 \; MHZ \)  // Set Analyzer Frequency to 135 MHz.  \( ANLZ:FREQ? \)  // Query the Analyzer Frequency.  // 135000 is returned (135000 kHz).

FULL
\[[ANLZ:FULL]\]
Selects a full size Analyzer display for the RF Generator, Receiver and Duplex Operation Screens.
ANLZ:

INPUT:

\textbf{ANTenna}
\[\text{[ANLZ:INPUT:ANTenna]}\]
Selects ANTENNA IN Connector for the Analyzer Input.

The following Attenuation commands control the Input Attenuation for the Spectrum Analyzer and selection/deselection of LNA (Low Noise Amplifier). Use \textbf{ANLZ:INPUT:ATTenuation:LNA} to select LNA and 0 dB of input attenuation. To deselect LNA and select a specific value of attenuation, use \textbf{ANLZ:INPUT:ATTenuation}.

\textbf{ATTenuation:LNA}
\[\text{[ANLZ:INPUT:ATTenuation:LNA]}\]
Sets Analyzer Input Attenuation to LNA.

\textbf{ATTenuation:LNA?}
\[\text{[ANLZ:INPUT:ATTenuation:LNA?]}\]
Returns current state of the Low Noise Amplifier. Returns 1 if LNA is selected; 0 otherwise.

\textbf{LNA} (Low Noise Amplifier) has the same attenuation as 0 dB (see \textbf{ANLZ:INPUT:ATTenuation} \textit{n}), but LNA has a lower noise figure. LNA is the preferred option for performing off-the-air applications.

\textbf{ATTenuation \textit{n}}
\[\text{[ANLZ:INPUT:ATTenuation \textit{n}]}\]
Sets Analyzer Input Attenuation to \textit{n} dB. Possible values for \textit{n}: 0, 5, 10, 15, 20, 25, 30.

\textbf{ATTenuation?}
\[\text{[ANLZ:INPUT:ATTenuation?]}\]
Returns the current value of Input Attenuation.

\textbf{TR}
\[\text{[ANLZ:INPUT:TR]}\]
Selects T/R Connector for the Analyzer Input.

\textbf{INPUT?}
\[\text{[ANLZ:INPUT?]}\]
Returns Analyzer Input setting.

Example:
\begin{itemize}
  \item \texttt{ANLZ:INP:TR} // Selects T/R Connector for Analyzer Input.
  \item \texttt{ANLZ:INP:ATT 20} // Sets Analyzer Input Attenuation to 20 dB.
  \item \texttt{ANLZ:INP?} // Queries Analyzer Input Connector. TR is returned.
  \item \texttt{ANLZ:INP:ATT?} // Queries Analyzer Input Attenuation. // 20 is returned.
\end{itemize}

\textbf{LIve}
\[\text{[ANLZ:LIVE]}\]
Selects Live Trace mode for the Spectrum Analyzer.
ANLZ:

MARKer:

AOFF
[ANLZ:MARKer:AOFF]
Disables both Markers.

DELTA:

AMPLitude?
[ANLZ:MARKer:DELTA:AMPLitude?]
Returns amplitude difference between the Trace Marker 1 and Trace Marker 2 crossings in dB.

FREquency?
[ANLZ:MARKer:DELTA:FREquency?]
Returns the difference between the two Marker positions in MHz.

POINT?
[ANLZ:MARKer:DELTA:POINT?]
Returns the difference between the Marker positions in graticules with 100 graticules equal to the Analyzer display width.

TRACK b
[ANLZ:MARKer:TRACK b]
Enables Marker Tracking Feature if b is 1, disables Marker Tracking Feature if b is 0. Tracking feature keeps Markers a constant distance apart.
Commands for Marker 1 and Marker 2 are identical and are listed as ANLZ:MARKERx commands. Parameter \( x \) is specified as 1 or 2 when the command is entered.

**ANLZ:**

**MARKERx:**

**AMPLitude?**  
[ANLZ:MARKERx:AMPLitude?]
Returns amplitude of the Trace at the Marker \( x \) crossing. Range and units depend on the current scale settings (ANLZ:SCALe commands).

**FREQuency?**  
[ANLZ:MARKERx:FREQuency?]
Returns Marker \( x \) position in kHz (250 to 2010000.0).

**POINT \( n \)**  
[ANLZ:MARKERx:POINT \( n \)]
Sets Marker \( x \) position to \( n \) graticules. Range of \( n \) is 1 to 100 with 100 graticules equal to the Analyzer display width.

**POINT?**  
[ANLZ:MARKERx:POINT?]
Returns Marker \( x \) position in graticules.

**STATe \( b \)**  
[ANLZ:MARKERx:STATe \( b \)]
Enables Marker \( x \) if \( b \) is 1, disables Marker \( x \) if \( b \) is 0.

**Example:**

ANLZ:MARKER1:STAT 1 // Enables Marker 1.
ANLZ:MARKER1:POIN 10 // Positions Marker 1 one divisions from the left edge of the color display.
ANLZ:MARKER2:POIN 30 // Positions Marker 2 three divisions from the left edge of the color display.
ANLZ:MARK:TRACK // Enables Marker Tracking.
ANLZ:MARKER1:POIN 60 // Positions Marker 1 six divisions from the left edge of the color display (and Marker 2 eight divisions from the left edge of the color display).
ANLZ:MARK:TRACK 1 // Disables Marker Tracking.
ANLZ:MARKER2:AMPL? // Queries the amplitude of the Trace at the Marker 2 position.
ANLZ:MARKER2:POIN? // Queries Marker 2 position in graticules. 80 is returned.
ANLZ:MARKER2:FREQ? // Queries Marker 2 position in kHz.
ANLZ:MARK:AOFF // Disables both Markers.

**STATe?**  
[ANLZ:MARKERx:STATe?]
Returns current state of Marker \( x \).
ANLZ:

**MODE type**
[ANLZ:MODE type]
Selects Analyzer RF Mode (DIREct [Direct Mode] or CHANnel [Channel Mode]). Channel Mode displays cellular channel frequency according to ANLZ:CHANnel commands.

**NORMALize**
[ANLZ:NORMALize]
Normalizes the Analyzer Trace to match the RF Generator Output.

**PEAK**
[ANLZ:PEAK]
Selects Peak Hold Feature for the Analyzer.

**PLOT:**
The ANLZ:PLOT:xxx commands send plotter data of the Analyzer grid, trace or units out the HOST RS-232 or GPIB Connector. SYSTEM:PLOT:GPIB or SYSTEM:PLOT:SERial command must be sent to change plotter output connector.

**GRID**
[ANLZ:PLOT:GRID]
Draws Analyzer grid on attached plotter.

**TRACE**
[ANLZ:PLOT:TRACE]
Draws Analyzer trace on attached plotter.

**UNITS**
[ANLZ:PLOT:UNITS]
Draws Analyzer units on attached plotter.

**QTR**
[ANLZ:QTR]
Selects a 1/4 size Analyzer display for the RF Generator, Receiver and Duplex Operation Screens.

**RCL n**
[ANLZ:RCL n]
Recalls Analyzer Trace and parameters stored in memory location n. Range of n is 1 to 9.

**RLEVEL n**
[ANLZ:RLEVEL n]
Sets reference or offset level in dB for the Analyzer only in 2 dB scale. Range of n is 0 to 64.

**RLEVEL?**
[ANLZ:RLEVEL?]
Returns reference or offset value in dB for the Analyzer in 2 dB scale.

**SCALE n**
[ANLZ:SCALE n]
Sets Analyzer Units/Division Factor to n dB. Select 2 or 10.
ANLZ:

SCALE:

UNIT?:type
[ANLZ:SCALE:UNIT?:type]
Sets Analyzer Scale Units to type. For T/R Analyzer Input, select DBM (dBm) or DBW (dBW). For ANTENNA Analyzer Input, select:

<table>
<thead>
<tr>
<th>DBM (dBm)</th>
<th>DBMV (dBmV)</th>
<th>DBUV (dBμV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBV (dBV)</td>
<td>DBUW (dBμW)</td>
<td></td>
</tr>
</tbody>
</table>

If DBW is selected, the T/R Connector is selected for the Analyzer Input. If DBV, DBMV, DBUV or DBUW is selected, the ANTENNA IN Connector is selected for the Analyzer Input.

UNIT?
[ANLZ:SCALE:UNIT?]
Returns Analyzer Scale Units.

SCALE?
[ANLZ:SCALE?]  
Returns Analyzer Scale in dB.

SCAN n
[ANLZ:SCAN n]
Sets Analyzer Scan Width to n kHz. Select 0 for zero scan or one of the following:

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>500</td>
</tr>
<tr>
<td>1000</td>
<td>2000</td>
<td>5000</td>
</tr>
<tr>
<td>10000</td>
<td>20000</td>
<td>50000</td>
</tr>
</tbody>
</table>

SCAN?
[ANLZ:SCAN?]
Returns Analyzer Scan Width in kHz.

STATE b
[ANLZ:STATE b]
Displays Analyzer display in the RF Generator, Receiver and Duplex Operation Screens if b is 1. Analyzer display is not displayed if b is 0.

STORe n
[ANLZ:STORe n]
Stores the current Analyzer Trace and environment (routings and settings) in memory location n. Range of n is 1 to 9.
ANLZ:

**TOP?**

([ANLZ:TOP?])

Returns the top of screen value in current units. Spectrum Analyzer Operation Screen must be displayed.

Example:

```
ANLZ:SCAL 2 // Selects 2 dB/div for Analyzer
// Units/Division Factor.
ANLZ:SCAN 100 // Sets Analyzer Scan Width to 100 kHz.
ANLZ:SCAL:UNIT:DBM // Selects dBm for Analyzer Scale Units.
ANLZ:TOP? // Queries the amplitude value for the top
// of the Analyzer display (in the current
// units).
ANLZ:SCAN? // Queries Analyzer Scan Width in kHz. 100
// is returned.
ANLZ:SCAL:UNIT? // Queries Analyzer Scale Units.
```

For the following **ANLZ:TRACE** commands, the Analyzer display is divided into 400 positions horizontally (0 signifying the left edge of the display, 399 signifying the right edge of the display) and 255 values vertically (0 signifying the bottom of the display, 255 signifying the top of the display):

**ANLZ**:

**TRACE**:

```
DATA n,offset,value,value,...,value
[ANLZ:TRACE:DATA n,offset,value,value,...,value]
```

Replaces points of a stored Trace n with specified values starting with offset. Offset is given in number of positions from the left edge of the display. Range of n is 1 to 9. Multiple values are separated by commas. Only one value can be displayed for each horizontal position of the Trace. Intended for remote GPIB or RS-232 use only.

```
DATA? [[[n],offset],points]
[ANLZ:TRACE:DATA? [[[n],offset],points]]
```

Sends the specified number of data points of Trace n to the Host. Optional Trace n can be the live Trace or a Trace stored in memory. The affected points start with the optional offset with an offset default of 0. Range of offset is 0 to 399. Range of n is 0 to 9 with 0 signifying the Live Trace (0 default). Range of the number of points is 1 to 400 with a default of 400. Intended for remote GPIB or RS-232 use only.
ANLZ:

TRACE:

GET name,n
[ANLZ:TRACE:GET name,n]
Assigns values of trace n (in graticules) to a declared array. Parameter name is the name of the declared array. Range of n is 0 to 9, 0 signifying the live trace and 1 to 9 signifying stored traces. If array is less than 400 values in length (stored trace length), the array values are assigned for the length of the array, the remaining portion of trace is left unassigned to an array.

GET? n,offset
[ANLZ:TRACE:GET? n,offset]
Returns value of a point in Trace n located the offset number of positions from the left edge of the Display. Trace n can be the live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace. Range of the offset is 0 to 399.

MAX? n,offset,points
[ANLZ:TRACE:MAX? n,offset,points]
Returns the maximum point of Trace n within the specified number of points starting with the given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace. Range of the optional offset is 0 to 399 with a default of 0. Range of optional number of points is 1 to 400 with a default of 400. Intended for remote GPIB or RS-232 use only.

MIN? n,offset,points
[ANLZ:TRACE:MIN? n,offset,points]
Returns the minimum point of Trace n within the specified number of points starting with the given offset. Result is returned in x,y format with x being the number of positions from the left edge and y being the number of values from the bottom. Trace n can be the live Trace or a Trace stored in memory. Range of n is 0 to 9 with 0 signifying the Live Trace. Range of the optional offset is 0 to 399 with a default of 0. Range of optional number of points is 1 to 400 with a default of 400. Intended for remote GPIB and RS-232 use only.
**ANLZ:**

**TRACK:**

**PUT** name, n

[ANLZ:TRACE:PUT name, n]

Assigns values of an array to trace n (in graticules). Parameter name is the array name. Range of n is 1 to 9, signifying stored traces. If array is less than 400 values in length (stored trace length), the array values are assigned for the length of the array, leaving the rest of the stored trace intact.

Example:

```
ANLZ:TRACE:DATA 2, 80, 128  // Changes a single Trace value,
    // located 2 divisions from the left
    // edge, to the midway point (on the
    // axis).

ANLZ:TRACE:GET? 0, 240  // Queries the value of the live
    // Trace at the point 240 (six
    // divisions from the left edge of
    // the display).

ANLZ:TRACE:DATA? 3, 200, 200  // Returns the Trace values of the
    // right half of the Trace stored
    // in memory location 3 (200 values).

ANLZ:TRACE:MAX? 0, 200  // Queries the maximum Trace value of
    // the left half of the live Trace.

ANLZ:TRACE:MIN? 0, 40, 80  // Queries the minimum Trace value
    // of the live Trace for the second
    // and third divisions.
```

**BWIDth f**

[ANLZ:TRACK:BWIDth f]

Sets Tracking Generator Bandwidth to f kHz. Select 3, 30, 300 or 3000.

**BWIDth?**

[ANLZ:TRACK:BWIDth?]

Returns Tracking Generator Bandwidth in kHz.

**LEVEL n**

[ANLZ:TRACK:LEVEL n]

Sets Tracking Generator Level to n dBm. Range of n is 0.0 to -137.0.

**LEVEL?**

[ANLZ:TRACK:LEVEL?]

Returns Tracking Generator Level in dBm.
ANLZ:

TRACK:

OUTput:

TR
[ANLZ:TRACK:OUTput:TR]
Selects T/R Connector as Tracking Generator Output Connector.

DUPlex
[ANLZ:TRACK:OUTput:DUPlex]
Selects DUPLEX OUT Connector as Tracking Generator Output Connector.

OUTput?
[ANLZ:TRACK:OUTput?]  
Returns current Tracking Generator Output Connector.

RESolution:

HIGH
[ANLZ:TRACK:RESolution:HIGH]
Selects high for Tracking Generator Resolution.

LOW
[ANLZ:TRACK:RESolution:LOW]
Selects low for Tracking Generator Resolution.

MED
[ANLZ:TRACK:RESolution:MED]
Selects medium for Tracking Generator Resolution.

RESolution?
[ANLZ:TRACK:RESolution?]  
Returns HIGH if Tracking Resolution is high, LOW if Tracking Resolution is low or MED if Tracking Resolution is medium.
ANLZ:

TRACK:

STATe b
[ANLZ:TRACK:STATe b]
Enables Tracking Generator if b is 1, disables Tracking Generator if b is 0.

STATe?
[ANLZ:TRACK:STATe?]
Returns 1 if Tracking Generator is active; 0 if inactive.

Example:

ANLZ:TRACK:LEV -10 // Sets Analyzer Tracking Generator Level to -10 dBm.
ANLZ:TRACK:RES:HIGH // Sets Analyzer Tracking Generator Resolution to high.
ANLZ:TRACK:BWID 30 // Sets Analyzer Bandwidth to 30 kHz.
ANLZ:TRACK:STAT 1 // Enables the Analyzer Tracking Generator.
ANLZ:TRACK:LEV? // Queries the Analyzer Tracking Generator Level. -10 is returned.
ANLZ:TRACK:RES? // Queries the Analyzer Tracking Generator Resolution. HIGH is returned.
ANLZ:TRACK:BWID? // Queries the Analyzer Bandwidth in kHz. 30 is returned.
ANLZ:TRACK:STAT? // Queries the current status of the Analyzer Tracking Generator. 1 is returned.

6-10-2 REMOTE SPECTRUM ANALYZER EXAMPLE

The following command sequence measures a 96 MHz signal received at the ANTENNA IN Connector and returns the amplitude in dB:

SCREEN:ANLZ // Displays the Analyzer Operation Screen.
ANLZ:SCAN 20 // Sets the Analyzer Scan Width to 20 kHz.
ANLZ:FREQ 96000 // Sets Analyzer RF Frequency to 96 MHz.
ANLZ:LIVE // Selects the Live Analyzer Mode.
ANLZ:SCALE 10 // Selects the 10 Units/Division Factor.
ANLZ:SCALE:UNIT:DBM // Selects dBm for the Analyzer Scale Units.
ANLZ:INPUT:ANTENNA // Selects the ANTENNA IN Connector for the Analyzer Input.
ANLZ:INPUT:ATTENUATION 0 // Sets Analyzer Input Attenuation to 0 dB.
ANLZ:MARKER1:STATE 1 // Enables Marker 1.
ANLZ:MARKER1:POINT 50 // Positions Marker 1 at the Analyzer RF Frequency.
ANLZ:MARKER1:AMPLITUDE? // Queries the amplitude of the Trace at the Marker 1 position.
6-11 METER COMMANDS

6-11-1 AF METER COMMANDS

M_AF:

ALARM b
[M_AF:ALARM b]
Enables Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

FILTER:

APASs
[M_AF:FILTER:APASs]
Selects the All-Pass Filter for the AF Meter.

APASs?
[M_AF:FILTER:APASs?]
Returns 1 if All-Pass Filter is enabled, 0 if disabled.

HPASs:

FREQuency f
[M_AF:FILTER:HPASs:FREQuency f]
Sets High-Pass cutoff frequency to f kHz. Range of f is 0.5 to 20.0.

FREQuency?
[M_AF:FILTER:HPASs:FREQuency?]
Returns High-Pass cutoff frequency in kHz.

STATE b
[M_AF:FILTER:HPASs:STATE b]
Enables High-Pass Filter if b is 1, disables if b is 0.

STATE?
[M_AF:FILTER:HPASs:STATE?]
Returns 1 if High-Pass Filter is enabled, 0 if disabled.

LPASs:

FREQuency f
[M_AF:FILTER:LPASs:FREQuency f]
Set Low-Pass cutoff frequency to f kHz. Range of f is 0.1 to 30.0.

FREQuency?
[M_AF:FILTER:LPASs:FREQuency?]
Returns Low-Pass cutoff frequency in kHz.
M_AF:

FILTER:

LPASs:

STATE b
[M_AF:FILTER:LPASs:STATE b]
Enables Low-Pass Filter if b is 1, disables if b is 0.

STATE?
[M_AF:FILTER:LPASs:STATE?]
Returns 1 if Low-Pass Filter is enabled, 0 if disabled.

Example:
M_AF:FILT:LPAS:FREQ 5 // Sets Low-Pass cutoff frequency to 5 kHz.
M_AF:FILT:LPAS:FREQ? // Queries the Low-Pass cutoff frequency in kHz. 5 is returned.
M_AF:FILT:LPAS:STAT? // Queries the Low-Pass Filter status. 1 is returned.

INPUT: type
[M_AF:INPUT: type]
Selects the Audio Frequency Meter Input. Select from the following for type:
XAUDIO (Ext Mod), DEMOD (Demod Audio), FGEN (Func Gen Out), SINAD (SINAD/BER) or POWER (RF Power).

LL:

LEVEL f
[M_AF:LL:LEVEL f]
Sets Lower Limit to f kHz. Range of f is 0.0000 to 0.2000 for an upper range of 0.2.
Range of f is 0.000 to 200.000 for upper ranges of 2, 20 and 200.

STATE b
[M_AF:LL:STATE b]
Enables Audio Frequency Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding the Lower Limit sets bit 3 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_AF:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

PEAK?
[M_AF:PEAK?]
Returns Audio Frequency Meter Peak reading in Hz (0.0 to 200000.0).

PH b
[M_AF:PH b]
Enables Peak Hold Feature when b is 1, disables Peak Hold Feature when b is 0.
M_AF:

RANGE:

AUTO
[M_AF:RANGE:AUTO]
Sets frequency range to Autorange.

UPPer f
[M_AF:RANGE:UPPer f]
Sets frequency range to f kHz. Select 0.2, 2, 20 or 200.
Example:

M_AF:INP:SINAD // Selects SINAD/BER IN Connector for AF
M_AF:RANG:UPP 20 // Sets AF Meter Range to 20 kHz.
M_AF:PH 1 // Enables AF Meter Peak Hold.
M_AF:LL:LEV 55.5 // Sets a Lower Limit of 55.5 kHz.
M_AF:LL:STAT 1 // Enables Lower Limit.
M_AF:ALARM 1 // Enables Alarm.

RCL n
[M_AF:RCL n]
Recalls Audio Frequency Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

RESolution n
[M_AF:RESolution n]
Sets Audio Frequency Meter Resolution (Gate Time) to n Hz. Select 0.1 (Gate Time of 10 s) or 1 (Gate Time of 1 s).

STORe n
[M_AF:STORe n]
Stores current Audio Frequency Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

UL:

LEVel f
[M_AF:UL:LEVel f]
Sets Upper Limit to f kHz. Range of f is 0.0000 to 0.2000 for an upper range of 0.2. Range of f is 0.000 to 200.000 for upper ranges of 2, 20 and 200.

STATe b
[M_AF:UL:STATe b]
Enables Audio Frequency Meter Upper Limit if b is 1, disables if b is 0. When enabled, exceeding the Upper Limit sets bit 2 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_AF:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.
M.AF?
[M.AF?]
Returns an Audio Frequency Meter reading in Hz (0.0 to 200000.0). See also MEASURE:AUdio?
query (6-16).

Reading is invalid if squelch is unbroken for some internal routings.

Example:  SCREEN:AF     // Displays AF Meter Operation Screen.
          M.AF:RANG:AUTO   // Sets AF Meter Range to Autorange.
          M.AF:UL:LEV 100   // Sets AF Meter Upper Limit to 100 kHz.
          M.AF:UL:STAT 1    // Enables Upper Limit.
          M.AF:LL:LEV 25    // Sets AF Meter Lower Limit to 25 kHz.
          M.AF:LL:STAT 1    // Enables Lower Limit.
          M.AF:INF:XAUDIO   // Selects EXT MOD IN Connector for AF Meter
                               // Input.
          M.AF:FILTER:LPAS:FREQ 10 // Sets Low-Pass Filter cutoff frequency to
                                 // 10 kHz.
          M.AF:ALARM 1         // Enables Alarm.
          M.AF:PH 1            // Enables Peak Hold feature.
          M.AF:PEAK?           // Queries AF Meter Peak reading.
          M.AF?               // Queries AF Meter reading.
6-11-2 FREQUENCY ERROR METER COMMANDS

M_RF:

**ALARM** \( b \)

\[ M_{RF:ALARM} b \]

Enables Alarm if \( b \) is 1, disables if \( b \) is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

**LL:**

**LEVEL** \( f \)

\[ M_{RF:LL:LEVEL} f \]

Sets Lower Limit to \( f \) kHz. Range of \( f \) is 0.0000 to 0.1000 for an upper range of 0.1. Range of \( f \) is 0.000 to 100.000 for upper ranges of 1, 10 and 100.

**STATE** \( b \)

\[ M_{RF:LL:STATE} b \]

Enables Lower Limit if \( b \) is 1, disables if \( b \) is 0. When enabled, exceeding the Lower Limit sets bit 5 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_RF:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

**PEAK?**

\[ M_{RF:PEAK?} \]

Returns Frequency Error Meter Peak reading in Hz (0.0 to 100000.0).

**PH** \( b \)

\[ M_{RF:PH} b \]

Enables Peak Hold Feature if \( b \) is 1, disables Peak Hold Feature if \( b \) is 0.

**RANGE:**

**AUTO**

\[ M_{RF:RANGE:AUTO} \]

Sets Frequency Error Meter range to Autorange.

**UPPer** \( f \)

\[ M_{RF:RANGE:UPPer} f \]

Sets Frequency Error Meter range to \( f \) kHz. Select 0.1, 1, 10 or 100.

**Example:**

\[
M_{RF:RANGE:UPP} 10 \quad // \quad \text{Sets Frequency Error Meter Range to} \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad // \quad 10 \text{ kHz.} \\
M_{RF:PH} 1 \quad // \quad \text{Enables Frequency Error Meter Peak Hold.} \\
M_{RF:LL:LEV} 20.5 \quad // \quad \text{Sets a Lower Limit of 20.5 kHz.} \\
M_{RF:LL:STAT} 1 \quad // \quad \text{Enables Lower Limit.} \\
M_{RF:ALARM} 1 \quad // \quad \text{Enables Alarm.}
\]

**RCL** \( n \)

\[ M_{RF:RCL} n \]

Recalls Frequency Error Meter environment (routings and settings) stored in memory location \( n \). Range of \( n \) is 1 to 9.
**M_RF:**

**RESolution f**

[M_RF:RESolution f]

Sets Frequency Error Resolution (Gate Time) to f Hz. Select 1 (Gate Time of 1 sec) or 10 (Gate Time of 0.1 sec).

**STORe n**

[M_RF:STORe n]

Stores current Frequency Error Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

**UL:**

**LEVel f**

[M_RF:UL:LEVel f]

Sets Upper Limit to f kHz. Range of f is 0.0000 to 0.1000 for an upper range of 0.1. Range of f is 0.000 to 100.000 for upper ranges of 1, 10 and 100.

**STATE b**

[M_RF:UL:STATE b]

Enables Upper Limit if b is 1, disables if b is 0. When enabled, exceeding the Upper Limit sets bit 4 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_RF:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

**M_RF?**

[M_RF?]?

Returns Frequency Error Meter reading in Hz (-100000.0 to +100000.0). Actual signal frequency can be calculated by converting the Receiver Frequency setting (RECeiver:FREQuency? query) from kHz to Hz and adding the Frequency Error Meter reading. For an actual measured frequency reading, see MEASure:FREQuency? query (6-16).

Reading is invalid if squelch is unbroken for some internal routings.

**Example:**

SCREEN:FREQ  // Displays Frequency Error Meter Operation

M_RF:RANG:AUTO  // Sets Frequency Error Meter Range to Autorange.

M_RF:UL:LEV 100  // Sets Frequency Error Meter Upper Limit to
// 100 kHz.

M_RF:UL:STAT 1  // Enables Upper Limit.

M_RF:LL:LEV 25  // Sets Frequency Error Meter Lower Limit to
// 25 kHz.

M_RF:LL:STAT 1  // Enables Lower Limit.

M_RF:ALARM 1  // Enables Alarm.

M_RF:PH 1  // Enables Peak Hold feature.

M_RF:PEAK?  // Queries Frequency Error Meter Peak reading.

M_RF?  // Queries Frequency Error Meter reading.
6-11-3  POWER METER COMMANDS

M_PWR:

ALARM b
[M_PWR:ALARM b]
Enables Power Meter Alarm if b is 1, disables if b is 0. Enabled Power Meter Alarm sounds when Upper or Lower Limit is surpassed.

DBM[:STATE] b
[M_PWR:DBM[:STATE] b]
Enables (b = 1) or disables (b = 0) the dBm digital readout. The :STATE portion of the command is optional.

DBM[:STATE]?
[M_PWR:DBM[:STATE]??]
Returns the state of the dBm digital readout. The :STATE portion of the command is optional.

EXT:

STATE b
[M_PWR:EXT:STATE b]
Enables External Loss/Gain Offset if b is 1, disables if b is 0. External Loss/Gain Offset compensates Power Meter readings for external gains or losses.

STATE?
[M_PWR:EXT:STATE??]
Returns External Loss/Gain Offset state setting.

OFFSet n
[M_PWR:EXT:OFFSet n]
Sets External Loss/Gain Offset value to n dBm. Range of n is -99.9 to 99.9. Positive values lower Power Meter readings and compensate for external gains. Negative values raise Power Meter readings and compensate for external losses.

OFFSet?
[M_PWR:EXT:OFFSet??]
Returns External Loss/Gain Offset value in dBm.

LL:

LEVel n
[M_PWR:LL:LEVel n]
Sets Power Meter Lower Limit to n W. Range of n is 0.0000 to 0.5000 for ranges: 0.02, 0.05, 0.1, 0.2 and 0.5. Range of n is 0.00 to 200.00 otherwise.

STATE b
[M_PWR:LL:STATE b]
Enables Power Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding the Lower Limit sets bit 7 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_PWR:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.
M_PWR:

PEAK?
[M_PWR:PEAK?]
Returns Power Meter Peak reading in mW (0.0 to 200000.0).

PH b
[M_PWR:PH b]
Enables Power Meter Peak Hold Feature if b is 1, disables if b is 0.

RANGE:

AUTO
[M_PWR:RANGE:AUTO]
Sets Power Meter range to Autorange.

UPPer n
[M_PWR:RANGE:UPPer n]
Sets Power Meter range to n W. Select from:

<table>
<thead>
<tr>
<th>Value</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.02</td>
<td>0.05</td>
</tr>
<tr>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>200</td>
<td>100</td>
</tr>
</tbody>
</table>

RCL n
[M_PWR:RCL n]
Recalls Power Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

Example: M_PWR:RANG:UPP 2 // Sets Power Meter Range to 2 W.
M_PWR:PH 1 // Enables Power Meter Peak Hold.
M_PWR:LL:LEV 0.75 // Sets a Lower Limit of 0.75 W.
M_PWR:LL:STAT 1 // Enables Lower Limit.
M_PWR:ALARM 1 // Enables Alarm.

RF:

ASSUMEd n
[M_PWR:RF:ASSUMEd n]
Sets Assumed RF Frequency, in kHz, within 1 MHz of signal measured. Range of n is 250.0 to 2010000.0 kHz.

ASSUMEEd?
[M_PWR:RF:ASSUMEEd?]
Returns Assumed RF Frequency.

STORE n
[M_PWR:STORE n]
Stores current Power Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.
M_PWR:

TYPE:

CW
[M_PWR:TYPE:CW]
Selects Average Power Measurement Type for Power Meter.

PEAK
[M_PWR:TYPE:PEAK]
Selects Peak Power Measurement Type for Power Meter.

RMS
[M_PWR:TYPE:RMS]
Selects RMS Power Measurement Type for Power Meter.

UL:

LEVEL n
[M_PWR:UL:LEVEL n]
Sets Power Meter Upper Limit to n W. Range of n is 0.0000 to 0.5000 for upper ranges: 0.02, 0.05, 0.1, 0.2 and 0.5. Range of n is 0.00 to 200.00 otherwise.

STATE b
[M_PWR:UL:STATE b]
Enables Power Meter Upper Limit if b is 1, disables if b is 0. When enabled, exceeding the Upper Limit sets bit 6 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_PWR:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

M_PWR?
[M_PWR?]
Returns a Power Meter reading in mW (0.0 to 200000.0). See also MEASURE:POWER? query (6-16).

Example:
SCREEN:POW
M_PWR:RANG:AUTO
M_PWR:UL:LEVEL 3
M_PWR:UL:STAT 1
M_PWR:LL:LEVEL .5
M_PWR:LL:STAT 1
M_PWR:ALARM 1
M_PWR:PH 1
M_PWR:PEAK?
M_PWR?
// Displays Power Meter Operation Screen.
// Sets Power Meter Range to Autorange.
// Sets Power Meter Upper Limit to 3 W.
// Enables Upper Limit.
// Sets Power Meter Lower Limit to 0.5 W.
// Enables Lower Limit.
// Enables Alarm.
// Enables Peak Hold feature.
// Queries Power Meter Peak reading.
// Queries Power Meter reading.
6-11-4  DEVIATION METER (PEAK) COMMANDS

M_DEV:

ALARM b
[M_DEV:ALARM b]
Enables Deviation Meter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

AVERAGE b
[M_DEV:AVERAGE b]
Enables Deviation Meter Averaging if b is 1, disables if b is 0.

LL:

LEVEL f
[M_DEV:LL:LEVEL f]
Sets the Deviation Meter Lower Limit to f kHz. f can vary from 0.00 to 20.00 for range values of 2, 5, 10 or 20 kHz. f varies from 0 to 100 otherwise.

STATE b
[M_DEV:LL:STATE b]
Enables Deviation Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding the Lower Limit sets bit 9 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm ([M_DEV:ALARM 1]) are enabled, exceeding the Lower Limit activates an audio alarm.

MODE:

BOTH
[M_DEV:MODE:BOTH]
Selects Both Mode for Deviation Meter, reading positive and negative deviation. Use M_DEV:POS? and M_DEV:NEG? to query meter readings.

NEGATIVE
[M_DEV:MODE:NEGATIVE]
Selects Negative Mode for Deviation Meter, reading negative deviation. Use M_DEV:NEG? to query meter readings.

NORMALIZE
[M_DEV:MODE:NORMALIZE]
Selects Normalized Mode for Deviation Meter, reading (positive + negative)/2 deviation. Use M_DEV:POS? to query meter readings.

POSITIVE
[M_DEV:MODE:POSITIVE]
Selects Positive Mode for Deviation Meter, reading positive deviation. Use M_DEV:POS? to query meter readings.

Deviation Meter Screen must be updated for a Deviation Meter Mode change to take effect.
Example: M_DEV:LL:LEV 10 // Sets a Lower Limit of 10 kHz.
M_DEV:LL:STAT 1 // Enables Lower Limit.
M_DEV:ALARM 1 // Enables Alarm.
M_DEV:MODE:POS // Selects the Positive Mode of the Deviation (Peak) Meter.
SCREEN:DEV // Updates the Deviation (Peak) Meter Operation Screen.

M_DEV:

MODE:NEG?
[M_DEV:NEG?]
Returns a negative Deviation Meter reading as an absolute value in kHz (0.00 to 100.00).

PEAK:

NEG?
[M_DEV:PEAK:NEG?]
Returns a negative Deviation Meter Peak reading as an absolute value in kHz (0.00 to 100.00).

POS?
[M_DEV:PEAK:POS?]
Returns a positive Deviation Meter Peak reading as an absolute value in kHz (0.00 to 100.00).

PH b
[M_DEV:PH b]
Enables the Deviation Meter Peak Hold Feature if b is 1, disables if b is 0.

POS?
[M_DEV:POS?]
Returns a positive Deviation Meter reading as an absolute value in kHz (0.00 to 100.00).

RANGE:

AUTO
[M_DEV:RANGE:AUTO]
Sets Deviation Meter range to Autorange.

UPPer f
[M_DEV:RANGE:UPPer f]
Sets Deviation Meter range to f kHz. Settings for f are: 2, 5, 10, 20, 50 or 100.

Example: M_DEV:MODE:BOOTH // Selects Both Mode.
SCREEN:DEV // Updates the Deviation (Peak) Meter Operation Screen.
M_DEV:RANG:UPP 10 // Sets the Range to 10 kHz.
M_DEV:PH 1 // Enables Peak Hold feature.
M_DEV:

RCL n
[M_DEV:RCL n]
Recalls Deviation Meter environment (routings and settings) stored in memory location n.
Range of n is 1 to 9.

STORe n
[M_DEV:STORe n]
Stores current Deviation Meter environment (routings and settings) in memory location n.
Range of n is 1 to 9.

UL:

LEVel f
[M_DEV:UL:LEVel f]
Sets Deviation Meter Upper Limit to f kHz. f can vary from 0.00 to 20.00 for range values
of 2, 5, 10 or 20 kHz. f varies from 0 to 100 otherwise.

STATe b
[M_DEV:UL:STATe b]
Enables Deviation Meter Upper Limit if b is 1, disables if b is 0. When enabled, exceeding
the Upper Limit sets bit 8 of the Instrument Status Register to one (activating bit 13 of the
Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper
Limit and Alarm (M_DEV:ALARM 1) are enabled, exceeding the Upper Limit activates an
audio alarm.

Example: SCREEN:DEV // Displays Deviation (Peak) Meter
// Operation Screen.
M_DEV:RANG:AUTO // Sets Range to Autorange.
M_DEV:UL:LEV 30 // Sets Upper Limit to 30 kHz.
M_DEV:UL:STAT 1 // Enables Upper Limit.
M_DEV:LL:LEV 5 // Sets Lower Limit to 5 kHz.
M_DEV:LL:STAT 1 // Enables Lower Limit.
M_DEV:ALARM 1 // Enables Alarm.
M_DEV:PH 1 // Enables Peak Hold feature.
M_DEV:AVE 1 // Enables Deviation (Peak) Meter averaging.
M_DEV:MODE:NEG // Selects the Negative Mode.
SCREEN:DEV // Updates the Deviation (Peak) Meter
// Operation Screen.
M_DEV:NEG? // Queries Negative Deviation (Peak) Meter
// reading.
6-11-5 MODULATION METER COMMANDS

The Modulation Meter measures AM.

**M_MOD:**

**ALARM b**

[M_MOD:ALARM b]

Enables Modulation Meter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

**LL:**

**LEVEL n**

[M_MOD:LL:LEVEL n]

Sets Modulation Meter Lower Limit to n%. Range of n is 0.0 to 100.0.

**STATE b**

[M_MOD:LL:STATE b]

Enables Modulation Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding the Lower Limit sets bit 11 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16).

When the Lower Limit and Alarm (M_MOD:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

**PEAK?**

[M_MOD:PEAK?]

Returns Modulation Meter Peak reading as a percentage (0.0 to 100.0).

**PH b**

[M_MOD:PH b]

Enables Modulation Meter Peak Hold Feature if b is 1, disables if b is 0.

**RANGE:**

**AUTO**

[M_MOD:RANGE:AUTO]

Sets Modulation Meter range to Autorange.

**UPPER n**

[M_MOD:RANGE:UPPER n]

Sets Modulation Meter range to n%. Select 40 or 100.

Example:

M_MOD:RANG:UPP 100 // Sets AM Modulation Meter Range to 100%.
M_MOD:PH 1 // Enables AM Modulation Meter Peak Hold.
M_MOD:LL:LEV 20.5 // Sets a Lower Limit of 20.5%.
M_MOD:LL:STAT 1 // Enables Lower Limit.
M_MOD:ALARM 1 // Enables Alarm.
M_MOD:

RCL n
[M_MOD:RCL n]
Recalls Modulation Meter environment (routings and settings) stored in memory location n.
Range of n is 1 to 9.

STORe n
[M_MOD:STORe n]
Stores current Modulation Meter environment (routings and settings) in memory location n.
Range of n is 1 to 9.

UL:

LEVel n
[M_MOD:UL:LEVel n]
Sets Modulation Meter Upper Limit to n%. Range of n is 0.0 to 100.0.

STATe b
[M_MOD:UL:STATe b]
Enables Modulation Meter Upper Limit if b is 1, disables if b is 0. When enabled,
exceeding the Upper Limit sets bit 10 of the Instrument Status Register to one (activating
bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16).
When the Upper Limit and Alarm (M_MOD:ALARM 1) are enabled, exceeding the Upper
Limit activates an audio alarm.

M_MOD?
[M_MOD?]
Returns a Modulation Meter reading as a percentage (0.0 to 100.0).

Example:
SCREEN:MOD          // Displays Modulation Meter Operation Screen.
M_MOD:UL:LEV 30     // Sets Modulation Meter Upper Limit to 30%.
M_MOD:UL:STAT 1     // Enables Upper Limit.
M_MOD:LL:LEV 5      // Sets AM Modulation Meter Lower Limit to 5%.
M_MOD:LL:STAT 1     // Enables Lower Limit.
M_MOD:ALARM 1       // Enables Alarm.
M_MOD:PH 1          // Enables Peak Hold feature.
M_MOD:PEAK?         // Queries Modulation Meter Peak reading.
M_MOD?              // Queries Modulation Meter reading.
6-11-6 DISTORTION METER COMMANDS

M_DIST:

ALARM b
[M_DIST:ALARM b]
Enables Distortion Meter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

AVERAGE b
[M_DIST:AVERAGE b]
Enables Average Feature if b is 1, disables if b is 0.

FILTER f
[M_DIST:FILTER f]
Sets Notch Filter frequency to f Hz. Range of f is 600 to 1400.

INPUT: type
[M_DIST:INPUT: type]
Sets Distortion Meter Input to type. Select DEMOD (Demod Audio), SINAD (SINAD/Ber), XAUDio (Ext Mod) or FGEM (Func Gen).

LL:

LEVEL n
[M_DIST:LL:LEVEL n]
Sets Lower Limit to n%. Range of n is 0.0 to 20.0.

STATE b
[M_DIST:LL:STATE b]
Enables Distortion Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding the Lower Limit sets bit 13 of the Instrument Status Register to one (activating bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-15). When the Lower Limit and Alarm (M_DIST:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

Example: M_DIST:INPUT:XAUD // Selects EXT MOD IN Connector for
// Distortion Meter Input.
M_DIST:FILTER 770 // Sets Notch Filter frequency to 770 Hz.
M_DIST:PH 1 // Enables Distortion Meter Peak Hold.
M_DIST:LL:LEVEL 5.3 // Sets a Lower Limit of 5.3 kHz.
M_DIST:LL:STATE 1 // Enables Lower Limit.
M_DIST:ALARM 1 // Enables Alarm.

PEAK?
[M_DIST:PEAK?]
Returns a Distortion Meter Peak reading as a percentage (0.0 to 20.0).

PH b
[M_DIST:PH b]
Enables Distortion Meter Peak Hold Feature if b is 1, disables if b is 0. Peak Hold takes effect only after SCREEN:DISTortion command.
M_DIST:

RCL n
[M_DIST:RCL n]
Recalls Distortion Meter environment (routings and settings) stored at memory location n.
Range of n is 1 to 9.

STOR e n
[M_DIST:STOR e n]
Stores current Distortion Meter environment (routings and settings) in memory location n.
Range of n is 1 to 9.

SELe ct:

CWEight
[M_DIST:SELe ct:CWEight]
Selects the C-Weight Filter.

CMES sage
[M_DIST:SELe ct:CMES sage]
Same as M_DIST:SELe ct:CWEight command.

LPASs f
[M_DIST:SELe ct:LPASs f]
Selects a Low-Pass Filter with cutoff frequency of f Hz. Range of f is 100 to 30000.
Example: M_DIST:SELe ct:LPAS 15000 // Selects a Low-Pass Filter with a
// 15 kHz cutoff frequency.

UL:

LEV el n
[M_DIST:UL:LEV el n]
Sets Upper Limit to n%. Range of n is 0.0 to 20.0.

STAT e b
[M_DIST:UL:STAT e b]
Enables Distortion Meter Upper Limit if b is 1, disables if b is 0. When enabled,
exceeding the Upper Limit sets bit 12 of the Instrument Status Register to one (activating
bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16).
When the Upper Limit and Alarm (M_DIST:ALARM 1) are enabled, exceeding the Upper
Limit activates an audio alarm.
M_DIST?

Returns a Distortion Meter reading as a percentage (0.0 to 20.0).

Example:

SCREEN: DIST // Displays Distortion Meter Operation Screen.
M_DIST: INP: DEMOD // Selects Demodulated Audio as the Distortion
                   // Meter Input.
M_DIST: FILT 1000 // Sets Notch Filter frequency to 1 kHz.
M_DIST: UL: LEV 15 // Sets Distortion Meter Upper Limit to 15%.
M_DIST: UL: STAT 1 // Enables Upper Limit.
M_DIST: LL: LEV 4 // Sets Distortion Meter Lower Limit to 4%.
M_DIST: LL: STAT 1 // Enables Lower Limit.
M_DIST: ALARM 1 // Enables Alarm.
M_DIST: SELECT: LPAS 20000 // Selects a Low-Pass Filter with a 20 kHz
                   // cutoff frequency.
M_DIST: AVE 1 // Enables Averaging.
M_DIST: PH 1 // Enables Peak Hold feature.
M_DIST: PEAK? // Queries Distortion Meter Peak reading.
M_DIST? // Queries Distortion Meter reading.
6-11-7  SINAD METER COMMANDS

M_SINAD:

ALARM b
[M_SINAD:ALARM b]
Enables SINAD Meter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

AVERAGE b
[M_SINAD:AVERAGE b]
Enables Average Feature if b is 1, disables if b is 0.

FILTER f
[M_SINAD:FILTER f]
Sets Notch frequency to f Hz. Range of f is 600 to 1400.

INPUT:type
[M_SINAD:INPUT:TYPE]
Selects type as the SINAD Meter Input. Select DEMOD (Demod Audio), SINAD (SINAD/BER), XAUDio (Ext Mod) or FGEN (Func Gen).

LL:

LEVEL n
[M_SINAD:LL:LEVEL n]
Sets SINAD Meter Lower Limit to n dB. Range of n is 3.0 to 40.0.

STATE b
[M_SINAD:LL:STATE b]
Enables SINAD Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding Lower Limit sets bit 1 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_SINAD:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

Example: M_SINAD:INPUT:XAUD // Selects EXT MOD IN Connector for Distortion Meter Input.
M_SINAD:FILTER 850 // Sets Notch Filter frequency to 850 Hz.
M_SINAD:PH 1 // Enables Distortion Meter Peak Hold.
M_SINAD:LL:LEVEL 4 // Sets a Lower Limit of 4 dB.
M_SINAD:LL:STATE 1 // Enables Lower Limit.

PEAK?
[M_SINAD:PEAK?]
Returns a SINAD Meter Peak reading in dB (3.0 to 40.0).

PH b
[M_SINAD:PH b]
Enables SINAD Meter Peak Hold Feature if b is 1, disables if b is 0.

RCL n
[M_SINAD:RCL n]
Recalls SINAD Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.
M_SIMAD:

RESolution n
[M_SIMAD:RESolution n]
Sets SINAD Meter readout resolution to n dB. n is 0.1 or 0.5.

RESolution?
[M_SIMAD:RESolution?]
Returns current resolution setting in dB (0.1 or 0.5).

SELetc:

CWEight
[M_SIMAD:SELetc:CWEight]
Selects C-Weight Filter.

CMESsage
[M_SIMAD:SELetc:CMESsage]
Same as M_SIMAD:SELetc:CWEight command.

LPASs f
[M_SIMAD:SELetc:LPASs f]
Selects Low-Pass Filter with cutoff frequency of f Hz. Range of f is 100 to 30000.

STORRe n
[M_SIMAD:STORRe n]
Stores current SINAD Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

UL:

LEVel n
[M_SIMAD:UL:LEVel n]
Sets SINAD Meter Upper Limit to n dB. Range of n is 3.0 to 40.0.

STATE b
[M_SIMAD:UL:STATE b]
Enables SINAD Meter Upper Limit if b is 1, disables if b is 0. When enabled, exceeding Upper Limit sets bit 0 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_SIMAD:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.
M_SINAD?
[M_SINAD?]

Returns a SINAD Meter reading in dB (3.0 to 40.0).

Example:

SCREEN:SINAD // Displays SINAD Meter Operation Screen.
M_SINAD:INP:DEMOD // Selects Demodulated Audio as the
                    // SINAD Meter.
M_SINAD:FLTL 1100 // Sets Notch Filter frequency to 1100 Hz.
M_SINAD:UL:LEV 15 // Sets SINAD Meter Upper Limit to 15 dB.
M_SINAD:UL:STAT 1 // Enables Upper Limit.
M_SINAD:LL:LEV 4 // Sets SINAD Meter Lower Limit to 4 dB.
M_SINAD:LL:STAT 1 // Enables Lower Limit.
M_SINAD:SELECT:LPAS 15000 // Selects a Low-Pass Filter with a 15 kHz
                           // cutoff frequency.
M_SINAD:AVE 1 // Enables Averaging.
M_SINAD:PH 1 // Enables Peak Hold feature.
M_SINAD:PEAK? // Queries SINAD Meter Peak reading.
M_SINAD? // Queries SINAD Meter reading.
6-11-8  SIGNAL STRENGTH METER COMMANDS

**M_SIG:**

PEAK?
[M_SIG:PEAK?]
Returns a Signal Strength Meter Peak reading (0 to 100).

PH b
[M_SIG:PH b]
Enables Peak Hold Feature if b is 1, disables if b is 0.

RCL n
[M_SIG:RCL n]
Recalls Signal Strength Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

STORE n
[M_SIG:STORE n]
Stores current Signal Strength Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

M_SIG?
[M_SIG?]
Returns a Signal Strength Meter reading (0 to 100).

Example:  M_SIG:PH 1  // Enables Peak Hold feature.
          M_SIG:PEAK?   // Queries Signal Strength Meter Peak reading.
          M_SIG?       // Queries Signal Strength Meter reading.
6-11-9  BIT ERROR RATE (BER) METER COMMANDS

M_BER:

PATtern:

FIXED
[M_BER:PATtern:FIXED]
Selects Fixed pattern for the BER Meter test data.

RANDom
[M_BER:PATtern:RANDom]
Selects Random pattern for BER Meter test data.

USER nn
[M_BER:PATtern:USER nn]
Selects a USER data pattern (could be a variable or expression) for the BER Meter test data using the specified 8 bit pattern nn. Patterns entered that are not base 10 are preceded with # character and letter signifying the number base: H (Hexadecimal), B (Binary) or Q (Octal). Pattern is displayed in Hexadecimal.

Example: M_BER:PAT:USER #Q216 // Selects a User Pattern of BE hexadecimal
         // (216 Octal) for the BER Meter test data.
         X=#HC0 // Assigns variable to a usable pattern.
         M_BER:PAT:USER X+3 // Selects a User Pattern of C3 hexadecimal.

USER?
[M_BER:PATtern:FIXEDUSER?]
Returns the specified USER data pattern.

POLarity:

NEGative
[M_BER:POLarity:NEGative]
Selects Negative Polarity for the BER Meter.

POSitive
[M_BER:POLarity:POSitive]
Selects Positive Polarity for the BER Meter.

POLarity?
[M_BER:POLarity?]
Returns the selected data Polarity. Returns POSITIVE or NEGATIVE.

RATE n
[M_BER:RATE n]
Sets BER Meter rate to n. Select from:

<table>
<thead>
<tr>
<th>Rate</th>
<th>75</th>
<th>150</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>600</td>
<td>1200</td>
<td>2400</td>
</tr>
<tr>
<td></td>
<td>4800</td>
<td>16000</td>
<td></td>
</tr>
</tbody>
</table>

RATE?
[M_BER:RATE?]
Returns the BER Meter Rate setting.
M_BER:

RCL n
[M_BER:RCL n]
Recalls the BER Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

SIZE n
[M_BER:SIZE n]
Sets BER Meter block size in bits to n. Range of n is 100 to 100000.
Example: M_BER:POL:NEG  // Selects Negative Polarity for the BER Meter.
M_BER:RATE 1200  // Sets the data rate to 1200 bps.
M_BER:SIZE 1000  // Sets the data pattern size to 1000 bits.

SIZE?
[M_BER:SIZE?]  
Returns the BER Meter block size setting.

STORE n
[M_BER:STORE n]
Stores current BER Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

TYPE:xxx
[M_BER:TYPE:xxx]
Selects xxx for the Bit Error Rate Type. Select GENerator, RECeiver, DUplex or BASEband.

M_BER?
[M_BER?]  
Returns the number of errors for the last pass.
Example: M_BER:TYPE:DUP  // Selects Duplex for BER Meter Type.
M_BER:RATE 2400  // Sets BER Meter data rate to 2400 bps.
M_BER:SIZE 10000  // Sets BER Meter Pattern Size to 10,000 bits.
M_BER:POL:POS  // Sets BER Meter Polarity to Positive.
M_BER:PAT:RAND  // Selects the Random Pattern Type for the BER
M_BER?  // Meter test.
M_BER?  // Queries a bit error rate reading.
DIGITAL MULTIMETER COMMANDS

M_DMM:

ALARM b
[M_DMM:ALARM b]
Enables Multimeter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when Upper or Lower Limit is surpassed.

FUNCTION:

CURRENT:

AC
[M_DMM:FUNCTION:CURRENT:AC]
Selects the AC Ammeter for the Multimeter Function.

DC
[M_DMM:FUNCTION:CURRENT:DC]
Selects the DC Ammeter for the Multimeter Function.

RESistance
[M_DMM:FUNCTION:RESistance]
Selects the Ohmmeter for the Multimeter Function.

VOLTage:

AC
[M_DMM:FUNCTION:VOLTage:AC]
Selects the AC Voltmeter for the Multimeter Function.

DC
[M_DMM:FUNCTION:VOLTage:DC]
Selects the DC Voltmeter for the Multimeter Function.

FUNCTION?
[M_DMM:FUNCTION?]
Returns active Multimeter Function (CURR:AC, CURR:DC, RES, VOLT:AC or VOLT:DC).

INPUT:IMPedance n
[M_DMM:INPUT:IMPedance n]
Sets Input Impedance to n ohms. Select 150, 600 or 1e6. Command valid only in AC Voltmeter Function.

Example:

M_DMM:FUNC? // Queries Multimeter Function. DCC is returned.
M_DMM:INPUT:IMP:1E6 // Sets Input Impedance at 1 MΩ.
M_DMM:

LL:

LEVEL n
[M_DMM:LL:LEVEL n]

Sets the Multimeters Lower Limit to n. Table 6-1 lists units and ranges for n.

<table>
<thead>
<tr>
<th>FUNCTION</th>
<th>ACC, DCC</th>
<th>ACV, DCV</th>
<th>OHMMETER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>20 and 200 mA</td>
<td>200 mV</td>
<td>200 Ω</td>
</tr>
<tr>
<td>n Units</td>
<td>A</td>
<td>V</td>
<td>kΩ</td>
</tr>
<tr>
<td>Range of n</td>
<td>0.00000 to 0.19999</td>
<td>0.000 to 19.999</td>
<td>0.000 to 19999</td>
</tr>
</tbody>
</table>

Table 6-1 DMM Upper and Lower Limit Ranges and Units

STATE b
[M_DMM:LL:STATE b]

Enables the Multimeters Lower Limit if b is 1, disables if b is 0. When enabled, exceeding Lower Limit sets bit 3 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_DMM:ALARm 1) are enabled, exceeding the Lower Limit activates an audio alarm.

PH b
[M_DMM:PH b]

Enables Multimeter Peak Hold feature if b is 1, disables if b is 0.

RANGE:

AUTO
[M_DMM:RANGE:AUTO]

Sets the Multimeter range to Autorange.

UPPer n
[M_DMM:RANGE:UPPer n]

For AC or DC Voltmeter, sets range to n volts. Settings for n are: 0.2, 2, 20, 200 or 2000.

For AC or DC Ammeter, sets range to n amps. Settings for n are: 0.02, 0.2, 2 or 20.

For Ohmmeter, sets range to n kΩ. Settings for n are: 0.2, 2, 20, 200, 2000 or 20000.

M_DMM:RANGE:AUTO and M_DMM:RANGE:UPPer commands must be followed by a SCREEN:DMM command.

Example: M_DMM:FUNC:CURR:AC // Selects AC Ammeter for the Multimeter
       // Function.
M_DMM:UPP 2    // Sets Ammeter range to 2 A.
M_DMM:FUNC:VOLT:AC // Selects AC Voltmeter for the Multimeter
       // Function.
M_DMM:RANG:UPP .2 // Sets Ohmmeter range to 200 Ω.
M_DMM:

RCL n
[M_DMM:RCL n]
Recalls Digital Multimeter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

STORE n
[M_DMM:STORE n]
Stores current Digital Multimeter environment (routings and settings) in memory location n. Range of n is 1 to 9.

UL:

LEVEL n
[M_DMM:UL:LEVEL n]
Sets Multimeters Upper Limit to n. Table 6-1 lists units and ranges for n.

STATE b
[M_DMM:UL:STATE b]
Enables Multimeters Upper Limit if b is 1, disables if b is 0. When enabled, exceeding Upper Limit sets bit 2 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_DMM:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

M_DMM?
[M_DMM?]
Returns a Multimeter reading in the current Function units according to Table 6-1. See also MEASURE: commands (6-16).

Example:  M_DMM:FUNC:CURR:DC  // Selects DC Ammeter for the Multimeter
            // Function.
            M_DMM:RANG:LEV 0.2  // Sets Ammeter range to 200 mA.
            M_DMM:PH 1  // Enables Peak Hold feature.
            M_DMM:ALARM 1  // Enables Alarm.
            M_DMM:UL:LEV 150  // Sets Upper Limit to 150 mA.
            M_DMM:UL:STAT 1  // Enables Upper Limit.
            M_DMM:LL:LEV 30  // Sets Lower Limit to 30 mA.
            M_DMM:LL:STAT 1  // Enables Lower Limit.
6-11-11  PHASE METER COMMANDS

M_PM:

ALARM b
[M_PM:ALARM b]
Enables Phase Meter Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when a limit is surpassed.

LL:

LEVEL n
[M_PM:LL:LEVEL n]
Sets the Phase Meter Lower Limit to n radians. Range of n is 0.00 to 10.00.

STATE b
[M_PM:LL:STATE b]
Enables the Phase Meter Lower Limit if b is 1, disables if b is 0. When enabled, exceeding Lower Limit sets bit 5 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Lower Limit and Alarm (M_PM:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

PH b
[M_PM:PH b]
Enables the Phase Meter Peak Hold Feature if b is 1, disables if b is 0.

RANGE:

AUTO
[M_PM:RANGE:AUTO]
Sets Phase Meter range to Autorange.

UPPER n
[M_PM:RANGE:UPPER n]
Sets Phase Meter range to n radians. Select 1, 5 or 10.

Example: M_PM:RANGE:UPP 5  // Sets Phase Meter Range to 5 radians.
M_PM:PH 1  // Enables Phase Meter Peak Hold.
M_PM:LL:LEV 1.65  // Sets a Lower Limit of 1.65 radians.
M_PM:LL:STAT 1  // Enables Lower Limit.
M_PM:ALARM 1  // Enables Alarm.

RCL n
[M_PM:RCL n]
Recalls Phase Meter environment (routings and settings) stored in memory location n. Range of n is 1 to 9.
M_PM:

STORE n
[M_PM:STORE n]
Stores current Phase Meter environment (routings and settings) in memory location n. Range of n is 1 to 9.

UL:

LEVEL n
[M_PM:UL:LEVEL n]
Sets Phase Meter Upper Limit to n radians. Range of n is 0.00 to 10.00.

STATe b
[M_PM:UL:STATE b]
Enables Phase Meter Upper Limit if b is 1, disables if b is 0. When enabled, exceeding Upper Limit sets bit 4 of the Instrument Summary Status Register to one (activating bit 1 of the Instrument Status Register, bit 13 of the Questionable Status Register and bit 3 of the Status Byte) (see 2-16). When the Upper Limit and Alarm (M_PM:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

M_PM?
[M_PM?]?
Returns a Phase Meter reading in radians (0.00 to 10.00).

Example:
SCREEN:PM // Displays Phase Meter Operation Screen.
M_PM:RANG:AUTO // Sets Phase Meter Range to Autorange.
M_PM:UL:LEV 7 // Sets Phase Meter Upper Limit to 7 radians.
M_PM:UL:STAT 1 // Enables Upper Limit.
M_PM:LL:LEV .5 // Sets Phase Meter Lower Limit to 0.5 radians.
M_PM:LL:STAT 1 // Enables Lower Limit.
M_PM:ALARM 1 // Enables Alarm.
M_PM:PH 1 // Enables Peak Hold feature.
M_PM:PEAK? // Queries Phase Meter Peak reading.
M_PM? // Queries Phase Meter reading.
6-11-12  DEVIATION METER (RMS) COMMANDS

M_DRMS:

ALARM b
[M_DRMS:ALARM b]
Enables Deviation Meter (RMS) Alarm if b is 1, disables if b is 0. Enabled Alarm sounds when a limit is surpassed.

AVERAGE b
[M_DRMS:AVERAGE b]
Enables Deviation Meter (RMS) Averaging if b is 1, disables if b is 0.

LL:

LEVEL f
[M_DRMS:LL:LEVEL f]
Sets Deviation (RMS) Lower Limit to f kHz. Range of f is 0.00 to 10.00.

STATE b
[M_DRMS:LL:STATE b]
Enables Deviation (RMS) Meter Lower Limit if b is 1, disables if b is 0. When the Lower Limit and Alarm (M_DRMS:ALARM 1) are enabled, exceeding the Lower Limit activates an audio alarm.

PH b
[M_DRMS:PH b]
Enables Deviation Meter (RMS) Peak Hold feature if b is 1, disables if b is 0.

RANGE:

AUTO
[M_DRMS:RANGE:AUTO]
Sets Deviation Meter (RMS) range to Autorange.

UPPER f
[M_DRMS:RANGE:UPPER f]
Sets Deviation Meter (RMS) range to f kHz. Select 2, 5 or 10.
Example: M_DRMS:RANG:UPP 10  // Sets Deviation Meter (RMS) Range to 10 kHz.
M_DRMS:PH 1  // Enables Deviation Meter (RMS) Peak Hold.
M_DRMS:LL:LEV 1.5  // Sets a Lower Limit of 1.5 kHz.
M_DRMS:LL:STAT 1  // Enables Lower Limit.
M_DRMS:ALARM 1  // Enables Alarm.

RCL n
[M_DRMS:RCL n]
Recalls Deviation Meter (RMS) environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

STORE n
[M_DRMS:STORE n]
Stores current Deviation Meter (RMS) environment (routings and settings) in memory location n. Range of n is 1 to 9.
M_DRMS:

UL:

LEVEL \( f \)

\([M\_DRMS:UL:LEVEL \( f \)]\)

Sets Deviation (RMS) Upper Limit to \( f \) kHz. Range of \( f \) is 0.00 to 10.00.

STATE \( b \)

\([M\_DRMS:UL:STATE \( b \)]\)

Enables Deviation Meter (RMS) Upper Limit if \( b \) is 1, disables if \( b \) is 0. When the Upper Limit and Alarm (M_DRMS:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

M_DRMS?

\([M\_DRMS?]\)

Returns a Deviation Meter (RMS) reading in kHz (0.00 to 10.00).

Example:

```
SCREEN:DRMS  // Displays Deviation Meter (RMS) Operation
              // Screen.
M_DRMS:RANG:AUTO  // Sets Deviation Meter (RMS) Range to Autorange.
M_DRMS:UL:LEV 8.4  // Sets Deviation Meter (RMS) Upper Limit
                    // to 8.4 kHz.
M_DRMS:UL:STAT 1  // Enables Upper Limit.
M_DRMS:LL:LEV 2  // Sets Deviation Meter (RMS) Lower Limit
                    // to 2 kHz.
M_DRMS:LL:STAT 1  // Enables Lower Limit.
M_DRMS:ALARM 1  // Enables Alarm.
M_DRMS:PH 1  // Enables Peak Hold feature.
M_DRMS:AVE 1  // Enables Averaging.
M_DRMS?  // Queries Deviation Meter (RMS) reading.
```
6-11-13  PHASE METER (RMS) COMMANDS

**M_PMRMS:**

**ALARM b**

*[M_PMRMS:ALARM b]*

Enables Phase Meter (RMS) Alarm if b is 1, disables Alarm if b is 0. Enabled Alarm sounds when a limit is surpassed.

**AVERAGE b**

*[M_PMRMS:AVERAGE b]*

Enables (b = 1) or disables (b = 0) the Averaging mode.

**LL:**

**LEVEL n**

*[M_PMRMS:LL:LEVEL n]*

Sets Phase Meter (RMS) Lower Limit to n radians. Range of n is 0.00 to 10.00.

**STATE b**

*[M_PMRMS:LL:STATE b]*

Enables Phase Meter (RMS) Lower Limit if b is 1, disables Lower Limit if b is 0. When the Lower Limit and Alarm (*M_PMRMS:ALARM 1*) are enabled, exceeding the Lower Limit activates an audio alarm.

**PH b**

*[M_PMRMS:PH b]*

Enables Phase Meter (RMS) Peak Hold feature if b is 1, disables Peak Hold feature if b is 0.

**RANGE:**

**AUTO**

*[M_PMRMS:RANGE:AUTO]*

Sets Phase Meter (RMS) range to Autorange.

**UPPER n**

*[M_PMRMS:RANGE:UPPER n]*

Sets Phase Meter (RMS) range to n radians. Select 1, 5 or 10.

**RCL n**

*[M_PMRMS:RCL n]*

Recalls Phase Meter (RMS) environment (routings and settings) stored in memory location n. Range of n is 1 to 9.

**STORE n**

*[M_PMRMS:STORE n]*

Stores current Phase Meter (RMS) environment (routings and settings) in memory location n. Range of n is 1 to 9.

Example:  

*M_PMRMS:RANG:AUTO* // Sets Phase Meter (RMS) range to Autorange.  
*M_PMRMS:PH 1* // Enables Phase Meter (RMS) Peak Hold.  
*M_PMRMS:STOR 3* // Stores state of Phase Meter (RMS) to Phase Meter (RMS) memory location 3.
M_PMRMS:

UL:

LEVEL n
\text{[M\_PMRMS\_UL\_LEVEL n]}
Sets Phase Meter (RMS) Upper Limit to \( n \) radians. Range of \( n \) is 0.00 to 10.00.

STATE b
\text{[M\_PMRMS\_UL\_STATE b]}
Enables Phase Meter (RMS) Upper Limit if \( b \) is 1, disables Upper Limit if \( b \) is 0. When the Upper Limit and Alarm (M_PMRMS:ALARM 1) are enabled, exceeding the Upper Limit activates an audio alarm.

M_PMRMS?
\text{[M\_PMRMS?]}
Returns a Phase Meter (RMS) reading in radians (0.00 to 10.00).

Example:
\begin{verbatim}
SCREEN:PMRMS // Displays Phase Meter (RMS) Operation Screen.
M_PMRMS:UPP 5 // Sets Phase Meter (RMS) Range to 5 radians.
M_PMRMS:UL:LEV 4.6 // Sets Phase Meter (RMS) Upper Limit to
// 4.6 radians.
M_PMRMS:UL:STAT 1 // Enables Phase Meter (RMS) Upper Limit.
M_PMRMS? // Queries Phase Meter (RMS) reading.
\end{verbatim}

6-11-14 AF LEVEL METER COMMAND

M_VRMS?
\text{[M\_VRMS?]}
Returns a Voltage RMS reading of the received AF Level (0.00 to 10.00).

6-12 ACCESSORY COMMANDS

The Accessory commands are used to remotely control the MIC/ACC Connector (see Figure 3-1 in the IFR-1900 Operation Manual).

ACCESSory:

INIT
\text{[ACCESSory\_INIT]}
Initializes MIC/ACC Connector prior to conducting communication. PTT (Press to Talk) is turned On (low). See PTT:STATE.

IO
\text{[ACCESSory\_IO]}
Sends/receives data through the MIC/ACC Connector.

RELEASE
\text{[ACCESSory\_RELEASE]}
"Releases" MIC/ACC Connector. PTT is turned Off (high).

STATE?
\text{[ACCESSory\_STATE?]}
Checks presence of MCC/ACC I/O device. 0 = none; 1 = device present.
6-13  CELLULAR AMPS/NAMPS

The AMPS/NAMPS\(^1\) Cell Site Monitor and Cellular Simulation is used to test Analog AMPS and NAMPS Cell Sites and Mobile Phones.

The following commands remotely direct the Test Set to display some of the main screens that make up the Cell Site Monitor and Cellular Simulation.

SCREEN:

The following commands remotely direct the Test Set to display some of the main screens that comprise the Cell Site Monitor and Cellular Simulation.

CELL

(SCREEND:CELL)
Displays Cell Site Monitor Forward Control Channel Screen.

GENCELLular

(SCREEND:GENCELLular)
Displays Cellular Simulation Main Menu.

GENFOCC

(SCREEND:GENFOCC)
Displays Cell Site Simulation Forward Control Channel Screen.

GENRECC

(SCREEND:GENRECC)
Displays Mobile Simulation Reverse Control Channel Screen.

GENFVC

(SCREEND:GENFVC)
Displays Cell Site Simulation Forward Voice Channel Screen.

GENRVC

(SCREEND:GENRVC)
Displays Mobile Simulation Reverse Voice Channel Screen.

---

1. AMPS and NAMPS protocols are used; however, frequencies are extended to include all of North American Digital Cellular (NADC) frequencies. NADC frequencies consist of the following: 800 MHz (AMPS), 450 MHz (NT400©) and 1900 MHz (Hyperband).
6-13-1 AMPS/NAMPS CELL SITE MONITOR COMMANDS

CELL:

WORDA
[CELL:WORDA]
Selects decoding of Stream A words.

WORDB
[CELL:WORDB]
Selects decoding of Stream B words.

BOTH
[CELL:BOTH]
Selects decoding of both Stream A and B words.

WORD?
[CELL:WORD?]
Returns current decoding selection: WORDA, WORDB or BOTH.

CHANnel n
[CELL:CHANnel n]
Specifies Channel. Range of n is 1 to 2047, depending on format.

CHANnel?
[CELL:CHANnel?]
Returns current Channel.

B_I?
[CELL:B_I?]
Returns current state of Busy/Idle bit or -1 if not available.

DCC?
[CELL:DCC?]
Returns current DCC value or -1 if not available.

SCC?
[CELL:SCC?]
Returns current SCC value or -1 if not available.

SID?
[CELL:SID?]
Returns current SID value or -1 if not available.

MIN?
[CELL:MIN?]
Returns current MIN value or -1 if not available. Returning format is: "xxx/xxx-xxxx."

ORDer?
[CELL:ORDer?]
Returns current ORDER value or -1 if not available.
VMAC?
[CELL:VMAC?]
Returns current VMAC value or -1 if not available.

VCHAN?
[CELL:VCHAN?]
Returns current Voice Channel number or -1 if not available.

CMAX_1?
[CELL:CMAX_1?]
Returns current CMAX-1 value or -1 if not available.

N_1?
[CELL:N_1?]
Returns current N-1 value or -1 if not available.

CMAC?
[CELL:CMAC?]
Returns current CMAC value or -1 if not available.

END?
[CELL:END?]
Returns current state of END bit or -1 if not available.

WFOM?
[CELL:WFOM?]
Returns current state of WFOM bit or -1 if not available.

ACTion?
[CELL:ACTion?]
Returns current Action field value or -1 if not available.

NAWC?
[CELL:NAWC?]
Returns current NAWC value or -1 if not available.

S?
[CELL:S?]
Returns current state of S bit or -1 if not available.

E?
[CELL:E?]
Returns current state of E bit or -1 if not available.

REGH?
[CELL:REGH?]
Returns current state of REGH bit or -1 if not available.

REGR?
[CELL:REGR?]
Returns current state of REGR bit or -1 if not available.

DTX?
[CELL:DTX?]
Returns current state of DTX bit or -1 if not available.
CELL:

RCF?
[CELL:RCF?]
Returns current state of RCF bit value or -1 if not available.

CPA?
[CELL:CPA?]
Returns current state of CPA bit or -1 if not available.

OLC?
[CELL:OLC?]
Returns current OLC value or -1 if not available.

BIS?
[CELL:BIS?]
Returns current state of BIS bit or -1 if not available.

REGINCR?
[CELL:REGINCR?]
Returns current REGINCR value or -1 if not available.

CHANPOS1?
[CELL:CHANPOS1?]
Returns current value of first channel position field from the directed retry message or -1 if not available.

CHANPOS2?
[CELL:CHANPOS2?]
Returns current value of the second channel position field from the directed retry message or -1 if not available.

CHANPOS3?
[CELL:CHANPOS3?]
Returns current value of the third channel position field from the directed retry message or -1 if not available.

CHANPOS4?
[CELL:CHANPOS4?]
Returns current value of the fourth channel position field from the directed retry message or -1 if not available.

CHANPOS5?
[CELL:CHANPOS5?]
Returns current value of the fifth channel position field from the directed retry message or -1 if not available.

CHANPOS6?
[CELL:CHANPOS6?]
Returns current value of the sixth channel position field from the directed retry message or -1 if not available.

NEWACC?
[CELL:NEWACC?]
Returns current NEWACC value or -1 if not available.
CELL:

MAXbusy:

PGR?
[CELL:MAXbusy:PGR?]
Returns current value of Maximum Busy for page response or -1 if not available.

OTHer?
[CELL:MAXbusy:OTHer?]
Returns current value of Maximum Busy for Other accesses or -1 if not available.

MAXSzttr:

PGR?
[CELL:MAXSzttr:PGR?]
Returns current value of Maximum Seizure Tries for page response or -1 if not available.

OTHer?
[CELL:MAXSzttr:OTHer?]
Returns current value of Maximum Seizure Tries for Other accesses or -1 if not available.

PSCC?
[CELL:PSCC?]
Returns current PSCC value or -1 if not available.

ESN?
[CELL:ESN?]
Returns current ESN value or -1 if not available.

DIGITs?
[CELL:DIGITs?]
Returns current Call address value or -1 if not available.

PDSCC?
[CELL:PDSCC?]
Returns current PDSCC value or -1 if not available.

EP?
[CELL:EP?]
Returns current EP value or -1 if not available.

EF?
[CELL:EF?]
Returns current EF value or -1 if not available.

DSCC?
[CELL:DSCC?]
Returns current DSCC value or -1 if not available.
CELL:

LOCALCTRL1?
[CELL:LOCALCTRL1?]
Returns current first position field for the Local Control value or -1 if not available.

LOCALCTRL2?
[CELL:LOCALCTRL2?]
Returns current second position field for the Local Control value or -1 if not available.

C12?
[CELL:C12?]
Returns current C12 value or -1 if not available.

C13?
[CELL:C13?]
Returns current C13 value or -1 if not available.

MST?
[CELL:MST?]
Returns current MST value or -1 if not available.

MSL?
[CELL:MSL?]
Returns current MSL value or -1 if not available.

FORMat:

AMPS
[CELL:FORMat:AMPS]
Sets channel format to AMPS (800 MHz).

NT400
[CELL:FORMat:NT400]
Sets channel format to NT400 (450 MHz).

PCS
[CELL:FORMat:PCS]
Sets channel format to PCS (1900 MHz).
CELL:

FORMAT:

NADC:

North American Digital Cellular. NADC frequencies consist of the following: 800 MHz (AMPS), 450 MHz (NT400©) and 1900 MHz (Hyperband).

BAND:

U8

[CELL:FORMAT:NADC:BAND:U8]
Sets NADC band to U8 (AMPS - 800 MHz).

U4

Sets NADC band to U4 (NT400© - 450 MHz).

HY

[CELL:FORMAT:NADC:BAND:HY]
Sets NADC band to HY (Hyperband - 1900 MHz).

NAMPS:

BAND:

Lower

[CELL:FORMAT:NAMPS:BAND:Lower]
Sets NAMPS band to Lower.

Middle

[CELL:FORMAT:NAMPS:BAND:Middle]
Sets NAMPS band to Middle.

Upper

[CELL:FORMAT:NAMPS:BAND:Upper]
Sets NAMPS band to Upper.

BAND?

[CELL:BAND?]
Returns NADC or NAMPS band. Returns one of the following: U8, U4, HY, L, M or U.

FORMAT?

[CELL:FORMAT?]
Returns cellular format. Returns one of the following: AMPS, NT400 or PCS.
CELL:

CAPTURE:

MODE: x
[CELL:CAPTURE:MODE:x]
Specifies Mode on which to Capture: MIN, ORDER, BOTH (MIN and ORDER) or OFF (none).

MODE?
[CELL:CAPTURE:MODE?] Returns current capture mode: MIN, ORDER, BOTH or OFF

MIN "xxx/xxx-xxxx"
[CELL:CAPTURE:MIN "xxx/xxx-xxxx"] Specifies MIN capture value. Range of x is 0-9, # and *.

MIN?
[CELL:CAPTURE:MIN?] Returns current MIN Capture value in the format: "xxx/xxx-xxxx."

ORDER:x
[CELL:CAPTURE:ORDER:x]
Specifies Order on which to capture. The following orders are valid for x:

<table>
<thead>
<tr>
<th>PAGE</th>
<th>ALERT</th>
<th>RELEASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>REORDER</td>
<td>SALERT</td>
<td>AUDIT</td>
</tr>
<tr>
<td>SNDAddr</td>
<td>INTERCEPT</td>
<td>MAINTENANCE</td>
</tr>
<tr>
<td>POWer</td>
<td>DRETTRY</td>
<td>AUTREG</td>
</tr>
<tr>
<td>AINTERCEPT</td>
<td>AREORDER</td>
<td>AALERT</td>
</tr>
<tr>
<td>VCDES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ORDER?
[CELL:CAPTURE:ORDER?] Returns current Order on which to capture.
6-13-2 AMPS/NAMPS CELLULAR SIMULATION COMMANDS

A. FORWARD CONTROL CHANNEL

CELL:

GEN:

FOCC:

SETUP
[CELL:GEN:FOCC:SETUP]
Configures the Test Set for the Forward Control Channel Screen without displaying screen.

SEND
[CELL:GEN:FOCC:SEND]
Begin transmitting the System Parameter Overhead Message

STOP
[CELL:GEN:FOCC:STOP]
Stops transmitting the System Parameter Overhead Message

CHANnel n
[CELL:GEN:FOCC:CHANnel n]
Specifies Channel. Range of n is 1 to 2047, depending on format.

CHANnel?
[CELL:GEN:FOCC:CHANnel?] Returns current Channel.

CMAC n
[CELL:GEN:FOCC:CMAC n]
Specifies value of Control Mobile Attenuation Code in the Control-Filler Message. Range of n is 0 to 7.

CMAC?

CMAx n
[CELL:GEN:FOCC:CMAx n]
Specifies value of Maximum number of channels in System Parameter Overhead Message. Range of n is 1 to 128.

CMAx?
CELL:

GEN:

FOCC:

CPA \ b
\{CELL:GEN:FOCC:CPA b\}
Enables \( b = 1 \) or disables \( b = 0 \) Combined Paging/Access bit in System Parameter Overhead Message.

CPA?
\{CELL:GEN:FOCC:CPA?\}

DCC \ n
\{CELL:GEN:FOCC:DCC n\}
Specifies value of Digital Color Code in System Parameter Overhead Message. Range of \n is 0 to 3.

DCC?
\{CELL:GEN:FOCC:DCC?\}

DTX \ n
\{CELL:GEN:FOCC:DTX n\}
Specifies value of Discontinuous Transmission in System Parameter Overhead Message. Range of \n is 0 to 3.

DTX?
\{CELL:GEN:FOCC:DTX?\}

E \ b
\{CELL:GEN:FOCC:E b\}
Enables \( b = 1 \) or disables \( b = 0 \) Extended Address bit in System Parameter Overhead Message.

E?
\{CELL:GEN:FOCC:E?\}
Returns current state of Extended Address bit in System Parameter Overhead Message.

EP \ b
\{CELL:GEN:FOCC:EP b\}
Enables \( b = 1 \) or disables \( b = 0 \) Extended Protocol bit in System Parameter Overhead Message.

EP?
\{CELL:GEN:FOCC:EP?\}
CELL:

GEN:

FOCC:

N n
[CELL:GEN:FOCC:N n]
Specifies value of Number of paging channels in System Parameter Overhead Message. Range of n is 1 to 32.

N?
[CELL:GEN:FOCC:N?]
Returns current value of Number of paging channels in System Parameter Overhead Message.

RCF b
[CELL:GEN:FOCC:RCF b]
Enables (b = 1) or disabled (b = 0) Read Control Filler bit in System Parameter Overhead Message.

RCF?
[CELL:GEN:FOCC:RCF?]
Returns current state of Read Control Filler bit in System Parameter Overhead Message.

REGH b
[CELL:GEN:FOCC:REGH b]
Enables (b = 1) or disabled (b = 0) Registration for Home Stations bit in System Parameter Overhead Message.

REGH?
[CELL:GEN:FOCC:REGH?]
Returns current state of Registration for Home Stations bit in System Parameter Overhead Message.

REGR b
[CELL:GEN:FOCC:REGR b]
Enables (b = 1) or disabled (b = 0) Registration for Roaming Mobile Phones bit in System Parameter Overhead Message.

REGR?
[CELL:GEN:FOCC:REGR?]
Returns current state of Registration for Roaming Mobile Phones bit in System Parameter Overhead Message.

S b
[CELL:GEN:FOCC:S b]
Enables (b = 1) or disabled (b = 0) Serial Number bit in System Parameter Overhead Message.

S?
[CELL:GEN:FOCC:S?]
Returns current state of Serial Number bit in System Parameter Overhead Message.
CELL:

GEN:

FOCC:

SID $n$
[CELL:GEN:FOCC:SID $n$]
Specifies value of System Identification number in System Parameter Overhead Message. Range of $n$ is 0 to 32767.

SID?

WFOM $b$
[CELL:GEN:FOCC:WFOM $b$] Enables ($b = 1$) or disables ($b = 0$) Wait for Overhead Message bit in the Control-Filler Message.

WFOM?

GLACT:

SETUP

SEND

REPEAT:$x$
[CELL:GEN:GLACT:REPEAT:$x$] Turns Repeat on ($x = ON$) or off ($x = OFF$) for the Global Action Overhead Message.

REPEAT?

STOP
CELL:

GEN:

GLACT:

CHANNEL \(n\)
[CELL:GEN:GLACT:CHANNEL \(n\)]
Specifies Channel. Range of \(n\) is 1 to 2047, depending on format.

CHANNEL?
[CELL:GEN:GLACT:CHANNEL?] 
Returns current Channel.

ACTION:

RESCAN \(b\)
[CELL:GEN:GLACT:ACTION:RESCAN \(b\)]
Turns on \((b = 1)\) or off \((b = 0)\) Rescan in the Global Action Overhead Message menu.

RESCAN?
[CELL:GEN:GLACT:ACTION:RESCAN?] 
Returns current state of Rescan in the Global Action Overhead Message menu.

REGINCR \(b\)
[CELL:GEN:GLACT:ACTION:REGINCR \(b\)]
Turns on \((b = 1)\) or off \((b = 0)\) Registration Increment in the Global Action Overhead Message menu.

REGINCR?
[CELL:GEN:GLACT:ACTION:REGINCR?] 
Returns current state of Registration Increment in the Global Action Overhead Message menu.

NEWACC \(b\)
[CELL:GEN:GLACT:ACTION:NEWACC \(b\)]
Turns on \((b = 1)\) or off \((b = 0)\) New Access channel set in the Global Action Overhead Message menu.

NEWACC?
[CELL:GEN:GLACT:ACTION:NEWACC?] 

OLC \(b\)
[CELL:GEN:GLACT:ACTION:OLC \(b\)]
Turns on \((b = 1)\) or off \((b = 0)\) Overload Control in the Global Action Overhead Message menu.

OLC?
[CELL:GEN:GLACT:ACTION:OLC?] 
Returns current state of Overload Control in the Global Action Overhead Message menu.
CELL:

GEN:

GLACT:

ACTion:

BIS b
[CELL:GEN:GLACT:ACTion:BIS b]
Turns on (b = 1) or off (b = 0) Access Type in the Global Action Overhead Message menu.

BIS?
[CELL:GEN:GLACT:ACTion:BIS?]
Returns current state of Access Type in the Global Action Overhead Message menu.

ACCESS b
[CELL:GEN:GLACT:ACTion:ACCESS b]
Turns on (b = 1) or off (b = 0) Access Attempt in the Global Action Overhead Message menu.

ACCESS?
[CELL:GEN:GLACT:ACTion:ACCESS?]

LOCAL1 b
[CELL:GEN:GLACT:ACTion:LOCAL1 b]
Turns on (b = 1) or off (b = 0) Local Control 1 in the Global Action Overhead Message menu.

LOCAL1?
[CELL:GEN:GLACT:ACTion:LOCAL1?]
Returns current state of Local Control 1 in the Global Action Overhead Message menu.

LOCAL2 b
[CELL:GEN:GLACT:ACTion:LOCAL2 b]
Turns on (b = 1) or off (b = 0) Local Control 2 in the Global Action Overhead Message menu.

LOCAL2?
[CELL:GEN:GLACT:ACTion:LOCAL2?]
Returns current state of Local Control 2 in the Global Action Overhead Message menu.
CELL:

GEN:

GLACT:

BIS \( b \)
[CELL:GEN:GLACT:BIS b]
Enables \( (b = 1) \) or disables \( (b = 0) \) of Busy-Idle Status bit in the Global Action Overhead Message.

BIS?
[CELL:GEN:GLACT:BIS?]

LOCALcntl \( n \)
[CELL:GEN:GLACT:LOCALcntl n]
Specifies value of Local Control in the Global Action Overhead Message. Range of \( n \) is 0 to 65535.

LOCALcntl?
[CELL:GEN:GLACT:LOCALcntl?]
Returns current value of Local Control in the Global Action Overhead Message.

MAXBusy:

OTHERn \( n \)
[CELL:GEN:GLACT:MAXBusy:OTHERn n]
Specifies value of Maximum number of Busy occurrences allowed for Other accesses in the Global Action Overhead Message. Range of \( n \) is 0 to 15.

OTHERn?
[CELL:GEN:GLACT:MAXBusy:OTHERn?]
Returns current value of Maximum number of Busy occurrences allowed for Other accesses in the Global Action Overhead Message.

PGR \( n \)
[CELL:GEN:GLACT:MAXBusy:PGR n]
Specifies value of Maximum number of Busy occurrences allowed for Page Responses in the Global Action Overhead Message. Range of \( n \) is 0 to 15.

PGR?
[CELL:GEN:GLACT:MAXBusy:PGR?]
Returns current value of Maximum number of Busy occurrences allowed for Page Responses in the Global Action Overhead Message.
CELL:

GEN:

GLACT:

MAXSztr:

OTHER n
[CELL:GEN:GLACT:MAXSztr:OTHER n]
Specifies value of Maximum number of Seizure attempts allowed for Other accesses in the Global Action Overhead Message. Range of n is 0 to 15.

OTHER?
[CELL:GEN:GLACT:MAXSztr:OTHER?]
Returns current value of Maximum number of Seizure attempts allowed for Other accesses in the Global Action Overhead Message.

PGR n
[CELL:GEN:GLACT:MAXSztr:PGR n]
Specifies value of Maximum number of Seizure attempts allowed for Page Responses in the Global Action Overhead Message. Range of n is 0 to 15.

PGR?
[CELL:GEN:GLACT:MAXSztr:PGR?]
Returns current value of Maximum number of Seizure attempts allowed for Page Responses in the Global Action Overhead Message.

NEWACC n
[CELL:GEN:GLACT:NEWACC n]
Specifies value of New Access Channel starting point in the Global Action Overhead Message. Range of n is 0 to 2047.

NEWACC?
[CELL:GEN:GLACT:NEWACC?]
Returns current value of New Access Channel starting point in the Global Action Overhead Message.

OLC n
[CELL:GEN:GLACT:OLC n]
Specifies value of Overload Control Class in the Global Action Overhead Message. Range of n is 0 to 32767.

OLC?
[CELL:GEN:GLACT:OLC?]
Returns current value of Overload Control Class in the Global Action Overhead Message.
CELL:

GEN:

GLACT:

**REGINCR** \( n \)

\[ \text{CELL:GEN:GLACT:REGINCR}\ n\]

Specifies value of Registration Increment in the Global Action Overhead Message. Range of \( n \) is 0 to 4095.

**REGINCR?**

\[ \text{CELL:GEN:GLACT:REGINCR?}\]

Returns current value of Registration Increment in the Global Action Overhead Message.

MSCM:

**SETUP**

\[ \text{CELL:GEN:MSCM:SETUP}\]

Configures the Test Set for the Mobile Station Control Message Screen without displaying screen.

**SEND**

\[ \text{CELL:GEN:MSCM:SEND}\]

Sends the Mobile Station Control Message.

**REPEAT:** \( x \)

\[ \text{CELL:GEN:MSCM:REPEAT}:x\]

Sets Repeat on (\( x = \text{ON}\)) or off (\( x = \text{OFF}\)) in the Mobile Station Control Message.

**REPEAT?**

\[ \text{CELL:GEN:MSCM:REPEAT}\]

Returns current state of Repeat in the Mobile Station Control Message.

**STOP**

\[ \text{CELL:GEN:MSCM:STOP}\]

Stops transmission of the Mobile Station Control Message.

**CHANNEL** \( n \)

\[ \text{CELL:GEN:MSCM:CHANNEL}\ n\]

Specifies Channel. Range of \( n \) is 1 to 2047, depending on format.

**CHANNEL?**

\[ \text{CELL:GEN:MSCM:CHANNEL?}\]

Returns current Channel.
CELL:

GEN:

MSCM:

CHAN $n$
[CELL:GEN:MSCM:CHAN $n$]
Specifies voice channel to which call is assigned in the Mobile Station Control Message. Range of $n$ is 0 to 1024.

CHAN?
[CELL:GEN:MSCM:CHAN?] 
Returns current voice channel assignment for call in the Mobile Station Control Message.

C12 $b$
[CELL:GEN:MSCM:C12 $b$]
Enables ($b = 1$) or disables ($b = 0$) C12 bit in the Mobile Station Control Message.

C12?
[CELL:GEN:MSCM:C12?] 
Returns current state of C12 bit in the Mobile Station Control Message.

C13 $b$
[CELL:GEN:MSCM:C13 $b$]
Enables ($b = 1$) or disables ($b = 0$) C13 bit in the Mobile Station Control Message.

C13?
[CELL:GEN:MSCM:C13?] 
Returns current state of C13 bit in the Mobile Station Control Message.

CHANPos $x,y$
[CELL:GEN:MSCM:CHANPos $x,y$]
Sets the position of a control channel relative to the first access channel in the Mobile Station Control Message. Range of $x$ is 1 to 6; range of $y$ is 0 to 127.

CHANPos? $n$
[CELL:GEN:MSCM:CHANPos? $n$]
Returns the current position of a control channel relative to the first access channel in the Mobile Station Control Message. Range of $n$ is 1 to 6.

CLI "string"
[CELL:GEN:MSCM:CLI "string"]
Specifies Call Line Identifier string in the Mobile Station Control Message. String may consist of 0-9, #, * and N (Null) with a maximum of 32 characters.

CLI?
[CELL:GEN:MSCM:CLI?] 
Returns current Call Line Identifier string in the Mobile Station Control Message.
CELL:  

GEN:  

MSCM:  

**DSCC**  \( n \)  
[**CELL**:GEN:MSCM:DSCC \( n \)]  
Specifies value of DSAT Color Code in the Mobile Station Control Message. Range of \( n \) is 0 to 7.

**DSCC?**  
[**CELL**:GEN:MSCM:DSCC?]  
Returns current value of DSAT Color Code in the Mobile Station Control Message.

**EF**  \( b \)  
[**CELL**:GEN:MSCM:EF \( b \)]  
Enables \( (b = 1) \) or disables \( (b = 0) \) Extended Protocol Forward Channel Indicator bit in the Mobile Station Control Message.

**EF?**  
[**CELL**:GEN:MSCM:EF?]  
Returns current state of Extended Protocol Forward Channel Indicator bit in the Mobile Station Control Message.

**LOCAL**  \( n \)  
[**CELL**:GEN:MSCM:LOCAL \( n \)]  
Specifies value of LOCAL in the Mobile Station Control Message. Range of \( n \) is 0 to 31.

**LOCAL?**  
[**CELL**:GEN:MSCM:LOCAL?]  
Returns current value of LOCAL in the Mobile Station Control Message.

**MIN**  "xxx/xxx-xxxx"  
[**CELL**:GEN:MSCM:MIN "xxx/xxx-xxxx"]  
Specifies Mobile Identification Number in the Mobile Station Control Message. Range of \( x \) is 0-9, # and *.

**MIN?**  
[**CELL**:GEN:MSCM:MIN?]  
Returns current Mobile Identification Number in the Mobile Station Control Message.

**MSL**  \( n \)  
[**CELL**:GEN:MSCM:MSL \( n \)]  
Specifies value of Message Length in the Mobile Station Control Message. Range of \( n \) is 0 to 31.

**MSL?**  
[**CELL**:GEN:MSCM:MSL?]  
Returns current value of Message Length in the Mobile Station Control Message.
CELL:

GEN:

MSCM:

**MST** \( n \)  
[CELL:GEN:MSCM:MST \( n \)]  
Specifies value of Message Type in the Mobile Station Control Message. Range of \( n \) is 0 to 255.

**MST?**  
[CELL:GEN:MSCM:MST?]  
Returns current value of Message Type in the Mobile Station Control Message.

**ORDQ** \( n \)  
[CELL:GEN:MSCM:ORDQ \( n \)]  
Specifies value of Order Qualifier in the Mobile Station Control Message. Range of \( n \) is 0 to 7.

**ORDQ?**  
[CELL:GEN:MSCM:ORDQ?]  
Returns current value of Order Qualifier in the Mobile Station Control Message.

**ORDER:** \( x \)  
[CELL:GEN:MSCM:ORDER:**x**]  
Specifies Order in the Mobile Station Control Message. The following orders are valid for \( x \):

- AUDIT
- INTRCPT
- VC_DES
- LC
- RELease
- EXTENDED
- DIR_RTRY
- REORDER

**SCC** \( n \)  
[CELL:GEN:MSCM:SCC \( n \)]  
Specifies value of Supervisory Audio Tone Color Code in the Mobile Station Control Message. Range of \( n \) is 0 to 3.

**SCC?**  
[CELL:GEN:MSCM:SCC?]  
Returns current value of Supervisory Audio Tone Color Code in the Mobile Station Control Message.

**SHORT_MESSAGE** "string"  
[CELL:GEN:MSCM:SHORT_MESSAGE "string"]  
Specifies Short Message string in Mobile Station Control Message. See IS-88, Appendix A for string details.

**SHORT_MESSAGE?**  
[CELL:GEN:MSCM:SHORT_MESSAGE?]  
Returns current Short Message string in Mobile Station Control Message.
CELL:

GEN:

MSCM:

VMAC n
[CELL:GEN:MSCM:VMAC n]
Specifies value of Voice Mobile Attenuation Code in Mobile Station Control Message. Range of n is 0 to 7.

VMAC?
[CELL:GEN:MSCM:VMAC?]
Returns current value of Voice Mobile Attenuation Code in Mobile Station Control Message.

VOICE_MESSAGE "string"
[CELL:GEN:MSCM:VOICE_MESSAGE "string"]
Specifies Voice Mail message string in Mobile Station Control Message. See IS-88, Appendix A for string details.

VOICE_MESSAGE?
[CELL:GEN:MSCM:VOICE_MESSAGE?] 
Returns current Voice Mail message string in Mobile Station Control Message.

VOICE_UNANSWERED "nn"
[CELL:GEN:MSCM:VOICE_UNANSWERED "nn"]
Specifies number of unanswered messages in Mobile Station Control Message. Range of nn is 00 to 99.

VOICE_UNANSWERED?
[CELL:GEN:MSCM:VOICE_UNANSWERED?] 
Returns current number of unanswered messages in Mobile Station Control Message.

VOICE_URGENT:x
[CELL:GEN:MSCM:VOICE_URGENT:x]
Turns on (x = ON) or off (x = OFF) Urgent message identifier in Mobile Station Control Message.

VOICE_URGENT?
[CELL:GEN:MSCM:VOICE_URGENT?] 
Returns current state of Urgent message identifier in Mobile Station Control Message.
B. REVERSE CONTROL CHANNEL

CELL:

GEN:

RECC:

SETUP
[CELL:GEN:RECC:SETUP]
Configures the Test Set for the Reverse Control Channel Screen without displaying screen.

SEND
[CELL:GEN:RECC:SEND]
Sends Reverse Control Channel message.

REPEAT:
[x]
[CELL:GEN:RECC:REPEAT:x]
Turns Repeat on (x = ON) or off (x = OFF).

REPEAT?
[CELL:GEN:RECC:REPEAT?]
Returns current state of Repeat.

STOP
[CELL:GEN:RECC:STOP]
Stops transmission of Reverse Control Channel message.

CHANNEL n
[CELL:GEN:RECC:CHANNEL n]
Specifies Channel. Range of n is 0 to 1023.

CHANNEL?

CALLED_ADDRESS "string"
[CELL:GEN:RECC:CALLED_ADDRESS "string"]
Specifies Called Address. String may consist of 0-9, #, and * with a maximum of 32 characters.

CALLED_ADDRESS?

E b
[CELL:GEN:RECC:E b]
Enables (b = 1) or disables (b = 0) Extended Address.

E?
CELL:

GEN:

RECC:

EP b
[CELL:GEN:RECC:EP b]
Sets state of Extended Protocol bit. (0 or 1).

EP?
[CELL:GEN:RECC:EP?]
Returns current state of Extended Protocol bit.

ER b
[CELL:GEN:RECC:ER b]
Enables (b = 1) or disables (b = 0) Extended Protocol Reverse Channel.

ER?
[CELL:GEN:RECC:ER?]
Returns current state of Extended Protocol Reverse Channel.

DCC n
[CELL:GEN:RECC:DCC n]
Specifies value of Digital Color Code. Range of n is 0 to 3.

DCC?
[CELL:GEN:RECC:DCC?]
Returns current value of Digital Color Code.

ESN "string"
[CELL:GEN:RECC:ESN "string"]
Specifies Electronic Serial Number. String may consist of 0-9 with a maximum of 11 digits.

ESN?
[CELL:GEN:RECC:ESN?]
Returns current Electronic Serial Number.

LOCAL n
[CELL:GEN:RECC:LOCAL n]
Specifies value of LOCAL. Range of n is 0 to 31.

LOCAL?
[CELL:GEN:RECC:LOCAL?]
Returns current value of LOCAL.

LT b
[CELL:GEN:RECC:LT b]
Enables (b = 1) or disables (b = 0) Last Try.

LT?
[CELL:GEN:RECC:LT?]
Returns current state of Last Try.
CELL:

GEN:

RECC:

MIN "xxx/xxx-xxxx"
[CELL:GEN:RECC:MIN "xxx/xxx-xxxx"]
Specifies Mobile Identification Number. String may consist of 0-9, # and *.

MIN?
[CELL:GEN:RECC:MIN?]
Returns current Mobile Identification Number.

MSL n
[CELL:GEN:RECC:MSL n]
Specifies value of Message Length. Range of n is 0 to 31.

MSL?
[CELL:GEN:RECC:MSL?]
Returns current value of Message Length.

MST n
[CELL:GEN:RECC:MST n]
Specifies value of Extended Protocol Message Type. Range of n is 0 to 255.

MST?
[CELL:GEN:RECC:MST?]
Returns current value of Extended Protocol Message Type.

ORDER n
[CELL:GEN:RECC:ORDER n]
Specifies value of Order. Range of n is 0 to 31.

ORDER?
[CELL:GEN:RECC:ORDER?]
Returns current value of Order.

ORDQ n
[CELL:GEN:RECC:ORDQ n]
Specifies value of Order Qualifier. Range of n is 0 to 7.

ORDQ?
[CELL:GEN:RECC:ORDQ?]
Returns current value of Order Qualifier.

S b
[CELL:GEN:RECC S b]
Enables (b = 1) or disables (b = 0) Serial Number.

S?
CELL:

GEN:

RECC:

SCM \( n \)
[CELL:GEN:RECC:SCM \( n \)]
Specifies value of Station Class Mark. Range of \( n \) is 0 to 15.

SCM?
[CELL:GEN:RECC:SCM?]
Returns current value of Station Class Mark.

T \( b \)
[CELL:GEN:RECC:T \( b \)]
Enables \( b = 1 \) or disables \( b = 0 \) Type of message.

T?
[CELL:GEN:RECC:T?]
Returns current state of Type of message.

C. FORWARD VOICE CHANNEL (AMPS)

CELL:

GEN:

FVC:

SETUP
[CELL:GEN:FVC:SETUP]
Configures the Test Set for the AMPS Forward Voice Channel Screen without displaying screen.

SEND
[CELL:GEN:FVC:SEND]
Sends the Mobile Station Control Message.

REPEAT:\( x \)
[CELL:GEN:FVC:REPEAT:\( x \)]
Turns Repeat on \( x = \text{ON} \) or off \( x = \text{OFF} \).

REPEAT?
[CELL:GEN:FVC:REPEAT?]
Returns current state of Repeat.

STOP
[CELL:GEN:FVC:STOP]
Stops the transmission of the Mobile Station Control Message.
CELL:

GEN:

FVC:

CHANNEL n
[CELL:GEN:FVC:CHANNEL n]
Specifies Channel. Range of n is 1 to 2047, depending on format.

CHANNEL?
[CELL:GEN:FVC:CHANNEL?]
Returns current Channel.

CHAN n
[CELL:GEN:FVC:CHAN n]
Specifies voice channel to which call is assigned in the Mobile Station Control Message. Range of n is 1 to 1024.

CHAN?
[CELL:GEN:FVC:CHAN?]
Returns current voice channel to which call is assigned in the Mobile Station Control Message.

C12 b
[CELL:GEN:FVC:C12 b]
Enables (b = 1) or disables (b = 0) C12 bit in Mobile Station Control Message.

C12?
[CELL:GEN:FVC:C12?]
Returns current state of C12 bit in Mobile Station Control Message.

C13 b
[CELL:GEN:FVC:C13 b]
Enables (b = 1) or disables (b = 0) C13 bit in Mobile Station Control Message.

C13?
[CELL:GEN:FVC:C13?]
Returns current state of C13 bit in Mobile Station Control Message.

CLI "string"
[CELL:GEN:FVC:CLI "string"]
Specifies Call Line Identifier string in the Mobile Station Control Message. String may consist of 0-9, #, * and N (Null) with a maximum of 32 characters.

CLI?
[CELL:GEN:FVC:CLI?]
Returns current Call Line Identifier string in the Mobile Station Control Message.
CELL:

GEN:

FVC:

DSCC \( n \)
[CELL:GEN:FVC:DSCC \( n \)]
Specifies value of DSAT Color Code in the Mobile Station Control Message. Range of \( n \) is 0 to 7.

DSCC?
[CELL:GEN:FVC:DSCC?]
Returns current value of DSAT Color Code in the Mobile Station Control Message.

LOCAL \( n \)
[CELL:GEN:FVC:LOCAL \( n \)]
Specifies value of LOCAL in the Mobile Station Control Message. Range of \( n \) is 0 to 31.

LOCAL?
[CELL:GEN:FVC:LOCAL?]
Returns current value of LOCAL in the Mobile Station Control Message.

MSL \( n \)
[CELL:GEN:FVC:MSL \( n \)]
Specifies value of Message Length in the Mobile Station Control Message. Range of \( n \) is 0 to 31.

MSL?
[CELL:GEN:FVC:MSL?]
Returns current value of Message Length in the Mobile Station Control Message.

MST \( n \)
[CELL:GEN:FVC:MST \( n \)]
Specifies value of Message Type in the Mobile Station Control Message. Range of \( n \) is 0 to 225.

MST?
[CELL:GEN:FVC:MST?]
Returns current value of Message Type in the Mobile Station Control Message.

ORDER: \( x \)
[CELL:GEN:FVC:ORDER: \( x \)]
Specifies ORDER in the Mobile Station Control Message. The following orders are valid for \( x \):

<table>
<thead>
<tr>
<th>PAGE</th>
<th>SNDADDR</th>
<th>S_ALERT</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUDIT</td>
<td>MAINTenance</td>
<td>ALERT</td>
</tr>
<tr>
<td>RElease</td>
<td>PWRLvl</td>
<td>HANDoFF</td>
</tr>
<tr>
<td>NAMPS_CH_ASGN</td>
<td>EXTENDED</td>
<td></td>
</tr>
</tbody>
</table>
CELL:

GEN:

FVC:

PSCC \( n \)
\[ \text{CELL:GEN:FVC:PSCC } n \]
Specifies value of Present SAT Color Code in the Mobile Station Control Message. Range of \( n \) is 0 to 3.

PSCC?
\[ \text{CELL:GEN:FVC:PSCC?} \]
Returns current value of Present SAT Color Code in the Mobile Station Control Message.

PWR_LVL \( n \)
\[ \text{CELL:GEN:FVC:PWR_LVL } n \]
Specifies Power Level. Range of \( n \) is 0 to 7.

PWR_LVL?
\[ \text{CELL:GEN:FVC:PWR_LVL?} \]
Returns Power Level.

SCC \( n \)
\[ \text{CELL:GEN:FVC:SCC } n \]
Specifies value of SAT Color Code in the Mobile Station Control Message. Range of \( n \) is 0 to 3.

SCC?
\[ \text{CELL:GEN:FVC:SCC?} \]
Returns current value of SAT Color Code in the Mobile Station Control Message.

SHORT_MESSAGE "string"
\[ \text{CELL:GEN:FVC:SHORT_MESSAGE } \text{"string"} \]
Specifies Short Message string in the Mobile Station Control Message. See IS-88, Appendix A for string details.

SHORT_MESSAGE?
\[ \text{CELL:GEN:FVC:SHORT_MESSAGE?} \]
Returns current Short Message string in the Mobile Station Control Message.

VMAC \( n \)
\[ \text{CELL:GEN:FVC:VMAC } n \]
Specifies value of Voice Mobile Attenuation Code in the Mobile Station Control Message. Range of \( n \) is 0 to 7.

VMAC?
\[ \text{CELL:GEN:FVC:VMAC?} \]
Returns current value of Voice Mobile Attenuation Code in the Mobile Station Control Message.
CELL:

GEN:

FVC:

VOICE_MESSAGE "string"
[CELL:GEN:FVC:VOICE_MESSAGE "string"]
Specifies Voice Mail message string in the Mobile Station Control Message. See IS-88, Appendix A for string details.

VOICE_MESSAGE?
[CELL:GEN:FVC:VOICE_MESSAGE?] Returns current Voice Mail message string in the Mobile Station Control Message.

VOICE_UNANSWERED "nn"
[CELL:GEN:FVC:VOICE_UNANSWERED "nn"]
Specifies number of unanswered messages in Mobile Station Control Message. Range of nn is 00 to 99.

VOICE_UNANSWERED?
[CELL:GEN:FVC:VOICE_UNANSWERED?] Returns current number of unanswered messages in Mobile Station Control Message.

VOICE_URGENT: x
[CELL:GEN:FVC:VOICE_URGENT: x]
Turns on (x = ON) or off (x = OFF) Urgent message identifier in Mobile Station Control Message.

VOICE_URGENT?
[CELL:GEN:FVC:VOICE_URGENT?] Returns current state of Urgent message identifier in Mobile Station Control Message.
D. FORWARD VOICE CHANNEL (NAMPS)

CELL:

GEN:

FVC:

NAMPS:

SETUP
CELL:GEN:FVC:NAMPS:SETUP
Configures the Test Set for the NAMPS Forward Voice Channel Screen without displaying screen.

SEND
CELL:GEN:FVC:NAMPS:SEND
Sends the Mobile Station Control Message.

NEXT
CELL:GEN:FVC:NAMPS:NEXT
Sends the next word of the Mobile Station Control Message, if required.

CHANNEL n
CELL:GEN:FVC:NAMPS:CHANNEL n
Specifies Channel. Range of n is 0 to 1023.

CHANNEL:x
CELL:GEN:FVC:NAMPS:CHANNEL:x
Specifies Band. The following bands are valid for x: LOWer, MIDdle, UPper.

CHANNEL?
CELL:GEN:FVC:NAMPS:CHANNEL?
Returns current Channel.

CHAN n
CELL:GEN:FVC:NAMPS:CHAN n
Specifies voice channel to which call is assigned in the Mobile Station Control Message (11 least significant bits). Range of n is 1 to 1024.

CHAN?
CELL:GEN:FVC:NAMPS:CHAN?
Returns current voice channel to which call is assigned in the Mobile Station Control Message (11 least significant bits).

BER n
CELL:GEN:FVC:NAMPS:BER n
Specifies number of allowable bit errors in Mobile Station Control Message. Range of n is 0 to 127.

BER?
CELL:GEN:FVC:NAMPS:BER?
Returns current number of allowable bit errors in Mobile Station Control Message.
CELL:
GEN:
FVC:
NAMPS:
C13 b
[CELL:GEN:FVC:NAMPS:C13 b]
Enables \((b = 1)\) or disables \((b = 0)\) C13 bit in Mobile Station Control Message.

C13?
[CELL:GEN:FVC:NAMPS:C13?]
Returns current state of C13 bit in Mobile Station Control Message.

C12 b
[CELL:GEN:FVC:NAMPS:C12 b]
Enables \((b = 1)\) or disables \((b = 0)\) C12 bit in Mobile Station Control Message.

C12?
[CELL:GEN:FVC:NAMPS:C12?]
Returns current state of C12 bit in Mobile Station Control Message.

DSCC n
[CELL:GEN:FVC:NAMPS:DSCC n]
Specifies value of DSAT Color Code in Mobile Station Control Message. Range of \(n\) is 0 to 7.

DSCC?
[CELL:GEN:FVC:NAMPS:DSCC?]
Returns current value of DSAT Color Code in Mobile Station Control Message.

CLI "string"
[CELL:GEN:FVC:NAMPS:CLI "string"]
Specifies Call Line Identifier string in the Mobile Station Control Message. String may consist of 0-9, #, * and N (Null) with a maximum of 32 characters.

CLI?
[CELL:GEN:FVC:NAMPS:CLI?] Returns current Call Line Identifier string in the Mobile Station Control Message.

CTYP b
[CELL:GEN:FVC:NAMPS:CTYP b]
Enables \((b = 1)\) or disables \((b = 0)\) Channel Type indicator in the Mobile Station Control Message.

CTYP?
[CELL:GEN:FVC:NAMPS:CTYP?]
Returns current state of Channel Type indicator in the Mobile Station Control Message.
CELL:

GEN:

FVC:

NAMPS:

**DSAT n**

*[CELL:GEN:FVC:NAMPS:DSAT n]*

Specifies value of Digital Supervisory Audio Tone in the Mobile Station Control Message. Range of $n$ is 0 to 6.

**DSAT?**

*[CELL:GEN:FVC:NAMPS:DSAT?]*

Returns current value of Digital Supervisory Audio Tone in the Mobile Station Control Message.

**LOCAL n**

*[CELL:GEN:FVC:NAMPS:LOCAL n]*

Specifies value of LOCAL in the Mobile Station Control Message. Range of $n$ is 0 to 31.

**LOCAL?**

*[CELL:GEN:FVC:NAMPS:LOCAL?]*

Returns current value of LOCAL in the Mobile Station Control Message.

**MSL n**

*[CELL:GEN:FVC:NAMPS:MSL n]*

Specifies value of Message Length in the Mobile Station Control Message. Range of $n$ is 0 to 31.

**MSL?**

*[CELL:GEN:FVC:NAMPS:MSL?]*

Returns current value of Message Length in the Mobile Station Control Message.

**MST n**

*[CELL:GEN:FVC:NAMPS:MST n]*

Specifies value of Message Type in the Mobile Station Control Message. Range of $n$ is 0 to 255.

**MST?**

*[CELL:GEN:FVC:NAMPS:MST?]*

Returns current value of Message Type in the Mobile Station Control Message.

**O_E b**

*[CELL:GEN:FVC:NAMPS:O_E b]*

Enables ($b = 1$) or disables ($b = 0$) Odd/Even in the Mobile Station Control Message.

**O_E?**

*[CELL:GEN:FVC:NAMPS:O_E?]*

Returns current state of Odd/Even in the Mobile Station Control Message.
**CELL:**

**GEN:**

**FVC:**

**NAMPS:**

**ORDER: x**

(CELL:GEN:FVC:NAMPS.ORDER:x)

Specifies Order in the Mobile Station Control Message. The following orders are valid for x:

- PAGE
- S_ALERT
- ALERT
- SUSP_CALLED_ADDR
- HANDoff
- SNDADDR_EVEN
- AUDIT
- RELease
- HAIVDOFF_ CONFIRM
- MAINTenauce
- FADE
- PWRLvl
- MRI
- EXTENDED

**PDSCC n**

(CELL:GEN:FVC:NAMPS:PDSCC n)

Specifies value of Present DSAT Color Code in the Mobile Station Control Message. Range of n is 0 to 7.

**PDSCC?**

(CELL:GEN:FVC:NAMPS:PDSCC?)

Returns current value of Present DSAT Color Code in the Mobile Station Control Message.

**PWRLVL n**

(CELL:GEN:FVC:NAMPS:PWRLVL n)

Specifies Power Level. Range of n is 0 to 7.

**PWRLVL?**

(CELL:GEN:FVC:NAMPS:PWRLVL?)

Returns Power Level.

**RSSI n**

(CELL:GEN:FVC:NAMPS:RSSI n)

Specifies value of Received Signal Strength in the Mobile Station Control Message. Range of n is 0 to 7.

**RSSI?**

(CELL:GEN:FVC:NAMPS:RSSI?)

Returns current value of Received Signal Strength in the Mobile Station Control Message.
CELL:

GEN:

FVC:

NAMPS:

SHORT_MESSAGE "string"
[CELL:GEN:FVC:NAMPS:SHORT_MESSAGE "string"]
Specifies Short Message string in the Mobile Station Control Message.
See IS-88, Appendix A for string details.

SHORT_MESSAGE?
[CELL:GEN:FVC:NAMPS:SHORT_MESSAGE?]  
Returns current Short Message string in the Mobile Station Control Message.

VMAC n
[CELL:GEN:FVC:NAMPS:VMAC n]
Specifies value of Voice Mobile Attenuation Code in the Mobile Station Control Message. Range of n is 0 to 7.

VMAC?
[CELL:GEN:FVC:NAMPS:VMAC?]  
Returns current value of Voice Mobile Attenuation Code in the Mobile Station Control Message.

VOICE_MESSAGE "string"
[CELL:GEN:FVC:NAMPS:VOICE_MESSAGE "string"]
Specifies Voice Mail message string in the Mobile Station Control Message.
See IS-88, Appendix A for string details.

VOICE_MESSAGE?
[CELL:GEN:FVC:NAMPS:VOICE_MESSAGE?]  
Returns current Voice Mail message string in the Mobile Station Control Message.

VOICE_UNANSWERED "nn"
[CELL:GEN:FVC:NAMPS:VOICE_UNANSWERED "nn"]
Specifies number of unanswered messages in Mobile Station Control Message. Range of nn is 00 to 99.

VOICE_UNANSWERED?
[CELL:GEN:FVC:NAMPS:VOICE_UNANSWERED?]  
Returns current number of unanswered messages in Mobile Station Control Message.

VOICE_Urgent:x
[CELL:GEN:FVC:NAMPS:VOICE_Urgent:x]
Turns on (x = ON) or off (x = OFF) Urgent message identifier in Mobile Station Control Message.

VOICE_Urgent?
[CELL:GEN:FVC:NAMPS:VOICE_Urgent?]  
Returns current state of Urgent message identifier in Mobile Station Control Message.
E. REVERSE VOICE CHANNEL (AMPS)

CELL:

GEN:

RVC:

SETUP
[CELL:GEN:RVC:SETUP]
Configures the Test Set for the AMPS Reverse Voice Channel Screen without displaying screen.

SEND
[CELL:GEN:RVC:SEND]
Sends an RVC message.

REPEAT:x
[CELL:GEN:RVC:REPEAT:x]
Turns Repeat on (x = ON) or off (x = OFF).

REPEAT?
[CELL:GEN:RVC:REPEAT?]
Returns current state of Repeat.

STOP
[CELL:GEN:RVC:STOP]
Stops transmission of any RVC message.

CHANNEL n
[CELL:GEN:RVC:CHANNEL n]
Specifies Channel. Range of n is 0 to 1023.

CHANNEL?
[CELL:GEN:RVC:CHANNEL?]
Returns current Channel.

CALLED_ADDRess "string"
[CELL:GEN:RVC:CALLED_ADDRess "string"]
Specifies Called Address string for the Called-Address Message. String consists of 0-9, # and * with a maximum of 32 characters.

CALLED_ADDRess?
[CELL:GEN:RVC:CALLED_ADDRess?]
Returns current Called Address string for the Called-Address Message.

LOCAL n
[CELL:GEN:RVC:LOCAL n]
Specifies value of LOCAL in Order Confirmation Message. Range of n is 0 to 31.

LOCAL?
[CELL:GEN:RVC:LOCAL?]
Returns current value of LOCAL in Order Confirmation Message.
CELL:

GEN:

RVC:

MESSAGE:x
[CELL:GEN:RVC:MESSAGE:x]
Specifies Message to be transmitted. The following messages are valid for x:
ORDERCONFIRMation, CALLEDADDRESS, EXTENDED.

MSL n
[CELL:GEN:RVC:MSL n]
Specifies value of Message Length in Extended Protocol Message. Range of
n is 0 to 31.

MSL?
[CELL:GEN:RVC:MSL?]
Returns current value of Message Length in Extended Protocol Message.

MST n
[CELL:GEN:RVC:MST n]
Specifies Message Type in Extended Protocol Message. Range of n is 0 to 255.

MST?
[CELL:GEN:RVC:MST?]
Returns current value of Message Type in Extended Protocol Message.

ORDER n
[CELL:GEN:RVC:ORDER n]
Specifies value of ORDER in Order Confirmation Message. Range of
n is 0 to 31.

ORDER?
[CELL:GEN:RVC:ORDER?]
Returns current value of ORDER in Order Confirmation Message.

ORDQ n
[CELL:GEN:RVC:ORDQ n]
Specifies value of Order Qualifier in Order Confirmation Message. Range of
n is 0 to 7.

ORDQ?
[CELL:GEN:RVC:ORDQ?]
Returns current value of Order Qualifier in Order Confirmation Message.
F. REVERSE VOICE CHANNEL (NAMPS)

CELL:

GEN:

RVC:

NAMPS:

SETUP
[CELL:GEN:RVC:NAMPS:SETUP]
Configures the Test Set for the NAMPS Reverse Voice Channel Screen without displaying screen.

SEND
[CELL:GEN:RVC:NAMPS:SEND]
Sends message.

NEXT
[CELL:GEN:RVC:NAMPS:NEXT]
Sends the next word of the Mobile Station Control Message, if required.

CHANNEL $n$
[CELL:GEN:RVC:NAMPS:CHANNEL $n$]
Specifies Channel. Range of $n$ is 1 to 2047, depending on format.

CHANNEL: $x$
[CELL:GEN:RVC:NAMPS:CHANNEL:$x$]
Specifies Band. The following bands are valid for $x$: LOWer, MIDdle, UPper.

CHANNEL?

BER $n$
[CELL:GEN:RVC:NAMPS:BER $n$]
Specifies number of Bit Errors in the MRI (Mobile Reported Interference) Order Message. Range of $n$ is 0 to 127.

BER?
[CELL:GEN:RVC:NAMPS:BER?] Returns current number of Bit Errors in the MRI (Mobile Reported Interference) Order Message.

CALLED_ADDRESS "string"
[CELL:GEN:RVC:NAMPS:CALLED_ADDRESS "string"]
Specifies Called Address for Flash/Called-Address Message string. String may consist of 0-9, # and * with a maximum of 32 characters.

CALLED_ADDRESS?
CELL:

GEN:

RVC:

NAMPS:

CONFIRMation \( b \)
\([CELL:GEN:RVC:NAMPS:CONFIRMation \( b \)]\)
Sets state of T bit in Reverse Voice Channel Message to specify that message is a Order Confirmation Message (\( b = 1 \)) or an Order (\( b = 0 \)).

CONFIRMation?  
\([CELL:GEN:RVC:NAMPS:CONFIRMation?]\)
Returns current state of T bit in Reverse Voice Channel Message to specify that message is a Order Confirmation Message (1) or an Order (0).

DSAT \( n \)
\([CELL:GEN:RVC:NAMPS:DSAT \( n \)]\)
Specifies value of Digital Supervisory Audio Tone. Range of \( n \) is 0 to 6.

DSAT?
\([CELL:GEN:RVC:NAMPS:DSAT?]\)
Returns current value of Digital Supervisory Audio Tone.

LOCAL \( n \)
\([CELL:GEN:RVC:NAMPS:LOCAL \( n \)]\)
Specifies value of LOCAL for Order Message or Order Confirmation Message. Range of \( n \) is 0 to 31.

LOCAL?
\([CELL:GEN:RVC:NAMPS:LOCAL?]\)
Returns current value of LOCAL for Order Message or Order Confirmation Message.

MESSAGE:\( x \)
\([CELL:GEN:RVC:NAMPS:MESSAGE:\( x \)]\)
Specifies message to transmitted. The following messages are valid for \( x \): MRI, ORDER, FLASH.

O_E \( b \)
\([CELL:GEN:RVC:NAMPS:O_E \( b \)]\)
Sets state of Odd/Even bit. (0 or 1).

O_E?
\([CELL:GEN:RVC:NAMPS:O_E?]\)
Returns current state of Odd/Even bit.

ORDQ \( n \)
\([CELL:GEN:RVC:NAMPS:ORDQ \( n \)]\)
Specifies value of Order Qualifier for order messages. Range of \( n \) is 0 to 7.

ORDQ?
\([CELL:GEN:RVC:NAMPS:ORDQ?]\)
Returns current value of Order Qualifier for order messages,
CELL:

GEN:

RVC:

NAMPS:

ORDER n  
[CELL:GEN:RVC:NAMPS:ORDER n]  
Specifies value of ORDER for order messages. Range of n is 0 to 31.

ORDER?  
[CELL:GEN:RVC:NAMPS:ORDER?]  
Returns current value of ORDER for order messages.

RSSI n  
[CELL:GEN:RVC:NAMPS:RSSI n]  
Specifies value of Received Signal Strength in MRI (Mobile Reported Interference) Order Message. Range of n is 0 to 7.

RSSI?  
[CELL:GEN:RVC:NAMPS:RSSI?]  
Returns current value of Received Signal Strength in MRI (Mobile Reported Interference) Order Message.

VMAC n  
[CELL:GEN:RVC:NAMPS:VMAC n]  
Specifies value of Voice Mobile Attenuation Code in Order Messages. Range of n is 0 to 7.

VMAC?  
[CELL:GEN:RVC:NAMPS:VMAC?]  
Returns current value of Voice Mobile Attenuation Code in Order Messages.
6-13-3  REMOTE AMPS/NAMPS CELL SITE MONITOR EXAMPLE

The following macro performs the Receiver Setup for AMPS/NAMPS Cell Site Monitor Operation:

// Define a macro named CELL_SETUP.
*DNC "CELL_SETUP",BEGIN

// Displays the Receiver Operation Screen.
SCREEN:REC

// Waits for command to finish executing.
*WAI

// Selects User Defined FM Data for Receiver Modulation.
REC:MOD:USER:MOD DATA

// Selects a Low-Pass IF Filter with a 30 kHz cutoff frequency.
REC:MOD:USER:FILT 30

// Selects a Low-Pass Post Detection Filter with a 15 kHz cutoff frequency.
REC:MOD:USER:POST:LPAS 15

// Selects User Defined High Speed for AGC.
REC:AGC:USER:H

// Sets Deviation Meter Range to 10 kHz.
M_DEV:RANG:UPP 10

// Sets AP Meter Gate Time to 0.1 sec (1 Hz).
M_AF:RES 1

END // End macro CELL_SETUP.

The following macro queries cellular readings once a min number is received:

// Define macro named Cell_Read.
*DNC "CELL_TEST",BEGIN

// Declare strings needed.
STRING MIN,ORDER

// Declare variables needed.
VAR SCC,DCC,SID,VMAC

// Declare variables needed.
VAR CHN,CMX,N1,CMAC

// Display AMPS/NAMPS Cell Site Monitor Screen.
SCREEN:CELL

// Wait for command execution to finish.
*WAI

// Select Control Channel 327
CELL:CHAN 327

// Loop while no Min is received.
DO

// Pause macro to allow Test Set processing.
$=STR(CELL:MIN?)

// Query for a MIN.
TPAUSE

// End of loop.
UNTIL $ !="-1"

// Set MIN string equal to received MIN.
MIN=$

// Query Order.
ORDER=STR(CELL:ORDER?)

// Query SCC value.
SCC=CELL:SCC?

// Query DCC value.
DCC=CELL:DCC?

// Query SID value.
SID=CELL:SID?

// Query VMAC value.
VMAC=CELL:VMAC?

// Query Voice Channel.
CHN=CELL:VCHAN?

// Query CMAX value.
CMX=CELL:CMAX_1?

// Query N-1 value.
N1=CELL:N_1?

// Query CMAX value.
CMAC=CELL:CMAC?

// Send MIN and Order to RS-232 Connector.
PPRINT STR(MIN)+","+STR(ORDER)

// Send readings to RS-232 Connector.
PPRINT SCC,DCC,SID,VMAC

// Send readings to RS-232 Connector.
PPRINT CHN,CMX,N1,CMAC

END // End of macro CELL_TEST.
6-14 PROGRAM COMMANDS

PROGram:STARTup:

NAME "name"
[PROGram:STARTup:NAME "name"]
Selects the macro name to execute at power up. The startup macro executes after the POWER Switch is pressed and the automatic self test is performed.

The startup macro can be avoided by continually pressing STOP TEST CONTROL Key during the 2 beeps of the 1-2-4 beep sequence at power up.

NAME?
[PROGram:STARTup:NAME?] Returns the name of the current power up macro designated by a PROGram:STARTup:NAME command.

DELETE
[PROGram:STARTup:DELETE]
Deletes the power up designation of the power up macro. After this command is executed, there is no power up macro until a PROGram:STARTup:NAME command is executed. This command does not delete the macro itself.

Example:
PROG:START:NAME "Main_Menu" // Selects macro Main_Menu as the power up macro. This macro now executes when the Test Set is turned on.

PROG:START:NAME? // Queries the name of the current power up macro. Main_Menu is returned.

PROG:START:DELETE // Deletes the power up designation.
6-15 FLASH MEMORY FILE DIRECTORY OPERATION

The Flash Memory File Directory allows storage of various files including Calibration Data Sets and allows user to select a macro to be executed from the front panel of the HOST. Operate Flash Memory File Directory using following procedure:

<table>
<thead>
<tr>
<th>STEP</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Press MTRS MODE Key. Press &quot;AUX&quot; Soft Function Key F6 to display Auxiliary Functions Menu. Press 7 DATA ENTRY Key to display File Directory Screen.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>File Directory</th>
<th>Free: 2551040</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Type</td>
</tr>
<tr>
<td>SET11</td>
<td>M</td>
</tr>
<tr>
<td>SET10</td>
<td>M</td>
</tr>
<tr>
<td>STATE1</td>
<td>S</td>
</tr>
<tr>
<td>CAL1</td>
<td>B</td>
</tr>
<tr>
<td>TRACE1</td>
<td>T</td>
</tr>
<tr>
<td>VALUE01</td>
<td>B</td>
</tr>
<tr>
<td>STRING_12</td>
<td>A</td>
</tr>
</tbody>
</table>

2. To load a macro from Flash memory into Test Set memory and then execute the macro, perform the following two steps: Move cursor to macro. Press "Exec" Soft Function Key F5.

3. To load a Calibration Data Set or a stored Test Set State (Test Set settings at time of store), move cursor to file. Press "Load" Soft Function Key F5.


Flash Memory space is released only when a Pack operation is performed.


Do not power off Test Set during Pack operation as files may be lost. Do not pack with a WINDOW open.

6. To Initialize Flash Memory, press "Init" Soft Function Key F4. Initializing clears Flash Memory and all files are lost.


Files are stored in Flash Memory using remote commands only. Spectrum Analyzer and Oscilloscope Traces and variables are loaded into Test Set using remote commands.
The following remote commands are used to operate Flash Memory File Directory.

**MMEMory:**

**ATTRIBUTE:**

**DELETE** "f"

[MME:MEMORY:ATTRIBUTE:DELETE "f"]

Sets the File Attribute of the specified Flash Memory file to DELETE. *f* is Flash Memory file name.

**HIDE** "f"

[MME:MEMORY:ATTRIBUTE:HIDE "f"]

Sets the File Attribute of the specified Flash Memory file to HIDE. *f* is Flash Memory file name.

**INITIALIZE** "f"

[MME:MEMORY:ATTRIBUTE:INITIALIZE "f"]

Sets the File Attribute of the specified Flash Memory file to INIT. *f* is Flash Memory file name.

**PACK** "f"

[MME:MEMORY:ATTRIBUTE:PACK "f"]

Sets the File Attribute of the specified Flash Memory file to PACK. *f* is Flash Memory file name.

**ATTRIBUTE?** "f"

[MME:MEMORY:ATTRIBUTE? "f"]

Returns the Attribute (DELETE, HIDE, INIT or PACK) of the specified Flash Memory. *f* is Flash Memory file name.

**CATalog?**

[MME:MEMORY:CATalog?]

Returns Flash Memory status. First number returned is memory space used in bytes. Second number returned is memory space available in bytes. Remainder data is returned in sets of 3 consisting of file name, file type and file size for each file stored in Flash Memory.

**CATalog:**

**ENTRY?** *n*


Returns file entry (file name, file type, file size) for given index. Returns $$$$ if past end of directory or --- for deleted file. *n* is line number (index) in Flash Memory File Directory. Range of *n* is 0 to 512.

**USED?**

[MME:MEMORY:CATalog:USED?]

Returns file space used, in bytes.

**FREE?**

[MME:MEMORY:CATalog:FREE?]

Returns available file space, in bytes.
**MMEMory:**

**DELETE** "f"

[MMEMory:DELETE "f"]

Deletes file but does not release memory space until Pack operation is done. *f* is Flash Memory file name.

**INITialize**

[MMEMory:INITialize]

Erases all files stored in Flash Memory.

**INITialize?**

[MMEMory:INITialize?]

Returns 1 if file system has been initialized; otherwise 0.

**LOAD:**

**MACRo** "m", "f"

[MMEMory:LOAD:MACRo "m", "f"]

Loads macros and variables stored as the file name from Flash Memory into Test Set memory. If *m* is ", designated macro is executed. If *m* is macro name, that macro is executed. If *m* is omitted (""), no macro is executed. *m* is name of designated macro; *f* is Flash Memory file name.

**STATe** *n*, "f"

[MMEMory:LOAD:STATe n,"f"]

Loads Test Set State stored as *f* from Flash Memory into Auxiliary Functions "Store Parameters Menu" as entry *n*. (*n* = 0 loads current state.) *n* is number of stored state of Test Set; *f* is Flash Memory file name. Range of *n* is 0 to 9.

**TRACe:**

**SCOPe** *n*, "f"

[MMEMory:LOAD:TRACe:SCOPe n,"f"]

Loads Oscilloscope trace stored as *f* into Oscilloscope "Store Parameters Menu" as entry *n*. (*n* = 0 loads live trace.) *n* is number of stored trace; *f* is Flash Memory file name. Range of *n* is 0 to 9.

**ANLZ** *n*, "f"

[MMEMory:LOAD:TRACe:ANLZ n,"f"]

Loads Spectrum Analyzer trace stored as *f* into Spectrum Analyzer "Store Parameters Menu" as entry *n*. (*n* = 0 loads live trace.) *n* is number of stored trace; *f* is Flash Memory file name. Range of *n* is 0 to 9.

**DATA** *v*, "f"

[MMEMory:LOAD:DATA v,"f"]

Loads variable stored as *f* into Test Set memory with name *v*. *v* is name of variable; *f* is Flash Memory file name.

**CALibration** "f"

[MMEMory:LOAD:CALibration "f"]

Loads Calibration Data from Flash Memory into Test Set memory. *f* is Flash Memory file name.
MMEemory:

**PACK**

Packs Flash Memory and frees memory space from deleted files.

**Do not power off Test Set during Pack function as files may be lost. Do not Pack with a WINDOW open.**

**STORe:**

**MACRo** "m", "f"

Stores all Test Set macros and variables (except free variables) in Flash Memory as f with macro specified as designated macro. m is name of designated macro; f is Flash Memory file name.

**STATe** n, "f"

Stores entry n of Auxiliary Functions "Store Parameters Menu" as f in Flash Memory. (n = 0 stores current state.) n is number of stored state of Test Set. Range of n is 0 to 9.

**TRACe:**

**SCOPe** n, "f"

Stores entry n (stored trace) of Oscilloscope "Store Parameters Menu" as f in Flash Memory. (n = 0 stores live trace.) n is number of stored trace; f is Flash Memory file name. Range of n is 0 to 9.

**ANLZ** n, "f"

Stores entry n (stored trace) of Spectrum Analyzer "Store Parameters Menu" as f in Flash Memory. (n = 0 stores live trace,) n is number of stored trace; f is Flash Memory file name. Range of n is 0 to 9.

**DATA** v, "f"

Stores variable v into Flash Memory as f. v is name of variable; f is Flash Memory file name.

**CALibration** "f"

Stores Test Set Calibration Data into Flash Memory. f is Flash Memory file name.

**TYPE?** "f"

Returns file type. Returns null string if file does not exist. f is Flash Memory file name.
Error messages are returned to HOST when an error occurs. Refer to Table 6-2 for description of error messages.

<table>
<thead>
<tr>
<th>ERROR NUMBER</th>
<th>ERROR DEFINITION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>220</td>
<td>Parameter Error</td>
<td>Incorrect number of parameters were entered with command.</td>
</tr>
<tr>
<td>224</td>
<td>Illegal Parameter Value</td>
<td>A parameter entered was not appropriate for command.</td>
</tr>
<tr>
<td>225</td>
<td>Out of Memory</td>
<td>Insufficient memory space to perform command.</td>
</tr>
<tr>
<td>250</td>
<td>Flash Storage Error</td>
<td>Indicates Flash Memory could not be erased or data could not be stored in Flash Memory.</td>
</tr>
<tr>
<td>253</td>
<td>Corrupt Media</td>
<td>Indicates Flash Memory not properly initialized. Initialize Flash Memory.</td>
</tr>
<tr>
<td>254</td>
<td>Media Full</td>
<td>Indicates insufficient Flash Memory space to perform command.</td>
</tr>
<tr>
<td>255</td>
<td>Directory Full</td>
<td>Indicates command not performed because 512 file names have been used.</td>
</tr>
<tr>
<td>256</td>
<td>File Name Not Found</td>
<td>Specified file not stored in Flash Memory.</td>
</tr>
<tr>
<td>257</td>
<td>File Name Error</td>
<td>Indicates command attempted to create file name already stored or file name syntax incorrect.</td>
</tr>
</tbody>
</table>

Table 6-2 Flash Memory Error Messages
6-16 GENERIC MEASURE COMMANDS

Generic commands use optional values, e to signify expected value and r to signify resolution. Expected values help determine Meter range. If r is omitted, the last resolution value is used.

MEASure:

**AUDio? [e,r]**

*MEASure:AUDio? [e,r]*

Returns an AF Meter frequency counter reading in Hz (0.0 to 200000.0).

**CURRent:**

**AC? [e]**

*MEASure:CURRent:AC? [e]*

Returns a DMM ac current reading in amps (0.00000 to 19.990).

**DC? [e]**

*MEASure:CURRent:DC? [e]*

Returns a DMM dc current reading in amps (0.00000 to 19.990).

**FREQuency? [e,r]**

*MEASure:FREQuency? [e,r]*

Returns a signal frequency reading in kHz (250.0 to 2010000.0 kHz).

**MIC?**

*MEASure:MIC?*

Returns a 0 if receiving MIC/ACC Input or 1 otherwise.

**PHASE? [e]**

*MEASure:PHASE? [e]*

Returns a Phase Meter reading in radians (0.00 to 10.00).

**POWer? [e]**

*MEASure:POWer? [e]*

Returns a Power Meter reading in mW (0.0 to 100000.0).

**RESistance? [e]**

*MEASure:RESistance? [e]*

Returns a DMM resistance reading in kΩ (0.0000 to 19990).

**SINAD? [r]**

*MEASure:SINAD? [r]*

Returns a SINAD Meter reading in dB (3.0 to 40.0). Select .1 or .5 as optional r in dB.

**SQUElch**

*MEASure:SQUElch?*

Returns a 1 if squelch broken, 0 if squelch is unbroken.
**MEASURE:**

**TEMPERATURE:**

**AMBIENT?**

[MEASURE:TEMPERATURE:AMBIENT?]

Returns the ambient temperature in °C (0.00000 to 100.00000).

**POWER?**

[MEASURE:TEMPERATURE:POWER?]

Returns the Power Termination temperature in °C (0.00000 to 100.00000).

**VOLTAGE:**

**AC? [e]**

[MEASURE:VOLTAGE:AC? [e]]

Returns a DMM ac voltage reading in volts (0.0000 to 1000.00).

**DC? [e]**

[MEASURE:VOLTAGE:DC? [e]]

Returns a DMM dc voltage reading in volts (0.0000 to 1000.00).

**SUPPLY? n**

[MEASURE:VOLTAGE:SUPPLY? n]

Returns a voltage measurement of n power supply in volts (0.00 to 20.00). Select -15, 5 or 15 for n.
6-17 INITIATE AND FETCH COMMANDS

INITiate and FETCH commands break the measure process into two commands. INITiate commands prepare the meter and FETCH commands return the reading. The event register of the Operation Instrument Status Register indicates when an INITiate command is completed (see example). Figure 2-6 displays the Operation Instrument Status bits.

INITiate:

AF
[INITiate:AF]
Prepares the Audio Frequency Meter for a FETCH: command to take a reading.

DMM
[INITiate:DMM]
Prepares the Audio Frequency Meter for a FETCH: command to take a reading.

RF
[INITiate:RF]
Prepares the Frequency Error Meter for a FETCH: command to take a reading.

FETCH:

AF?
[FETCH:AF?]
Returns Audio Frequency Meter reading in kHz. Must be used with INITiate command.

DMM?
[FETCH:DMM?]
Returns Digital Multimeter reading. For ACV and DCV Multimeter Function, reading is returned in V. For ACC and DCO Multimeter Function, reading is returned in A. For Ohm Multimeter Function, reading is returned in kΩ. Must be used with INITiate command.

RF?
[FETCH:RF?]
Returns Frequency Error Meter reading in kHz. Must be used with INITiate command.

The following example reads the Operation Instrument Register until the RF Meter is ready to be read.

Example:  *DMC "RF_Read",BEGIN // Define macro named RF_Read.
N=0 // Set variable equal to zero.
SCREEN:FREQ // Display Frequency Error Meter Operation
// Screen.
SCREEN:USER // Display User Screen.
*CLS // Clear all condition and event registers.
INIT:RF // Initialize for a RF Meter reading.
WHILE (N&8)=0 // Loop until Frequency Error Meter bit of
N=STAT:OPER:instr:EVENT? // Operation Instrument event register is
// set to 1.
WEND
PRINT FETCH:RF? // Print Frequency Error Meter reading.
END // End of macro RF_Read.
6-18 AUXILIARY (SPECIAL TEST) COMMANDS

Auxiliary (Sp Tst) commands are used to communicate with the Sp Tst through the internal SCSI bus. Because of the vast command possibilities, the HOST does not check the Auxiliary commands for correct string syntax.

AHIT?
[AHIT?]
Returns a 1 if there is input waiting from the Sp Tst.

AUX "string"
[AUX "string"]
Issues commands, as strings, to the Sp Tst.
Example: AUX "DELAY 200"  // The DELAY 200 command is passed to and // executed by the Sp Tst.

AUX? "string?"
[AUX? "string??]
Issues queries, as strings, to the Sp Tst.
Example: AUX? "IDN?"  // The IDN? query is passed to the Sp Tst. // The Sp Tst executes the query and // returns the result.
INDEX

B
BCOLOR Command 2-18, 3-22
BEGIN Command 2-10, 3-22
BER
Bit Definition (Sp Tst) 9-49
Commands (Sp Tst) 9-389
Meter Commands (HOST) 6-113
Program Example (Sp Tst) 10-21
Bit Error Rate (BER) Meter Commands 6-113
Bitwise
AND Operation Results 3-5
OR Operation Results 3-7
XOR Operation Results 3-6
BOX
Command 2-21, 3-23
Example 3-23
Buffered Raw Data 9-62
FDCCH 9-136
RDCCH 9-136
Builder, RACH Layer 3 Message 9-349
Byte. Status 2-30
C
CALCulate? Query 3-24
Calculation, Order of 2-8
Calling Setup 9-166
CASE OF OTHERWISE ENDCASE Command 3-24
Cell Site Monitor, AMPS/NAMPS 6-125
Remote Example 6-163
Cell Site Simulation
Commands 9-156
Program Examples
ACC 10-10
DCCH 10-25
Cellular AMPS/NAMPS 6-124
Simulation 6-132
CENTER Command 2-21, 3-26
Characters, Screen
*CLS Command 2-34, 3-12
D-1
CLS Command 2-19, 3-27
Character Rotation for n Values 3-73
CHR Command 2-10, 3-27
COLOR Command and query 2-18, 3-28
Color Display 2-19
Colors 2-18
and Color Selection Numbers 2-18
Comments 2-3
Communication, Operating Using GPIB
HOST 2-25
Sp Tst 2-24
Compliance Commands, IEEE 488.2 2-34
Configuration, Status Register 2-29
Configure Commands, System RS-232 2-24
CONST Command 2-5, 3-29
Constants
Predefined Macros and 2-9
and Data Arrays 2-5
Continuous and Sub Channel
Transmission 9-337, 9-345
Continuous Remote Raw Data
FDCCH 9-61
RDCCH 9-135
Continuous Remote Raw Timeslot Data
FDCCH 9-60
RDCCH 9-134
A
Abbreviated Length Message in the RDCCH 9-336
Abbreviations E-1
ABS Command 2-8, 3-19
Accessory Commands 6-123
Accuracy Commands (FDTC), Modulation 9-391
ACTIVATE Command 2-15, 3-20
AF Generator 6-56
Remote Examples 6-65
AF Level Meter Commands 6-123
AF Meter Commands 6-92
AMPS/NAMPS, Cellular
Monitor 6-125
Remote Example 6-163
Simulation 6-132
Analyzer, Spectrum 6-79
Remote Example 6-91
AND
Command 2-13, 3-20
Operation Results 3-21
AND Operation Results, Bitwise 3-5
Arrays
Data 2-6
in Macros 2-14
Numeric 2-3
Saving (HOST Only) 2-4
ASC Command 2-10, 3-21
Audio Tones 2-22
Auxiliary (Special Test) Commands 6-173
Index-1
C (cont)

Control
Display and Sound 2-18
Frequency 2-27
GPIO Print 2-27
Video Page 2-22

COS Command
2-8, 3-29

Creating a TMAC Program 4-3
Creating and Loading TMAC Programs 4-1
Setup 4-2

Cycles, Examples of Primary and Secondary 9-159

D

Data Arrays, Constants and 2-5
DATA Command 2-5, 2-22, 3-30

Data Message
Diagram, User 9-342
Superframe 9-220

Data Monitor
FDCCH 9-59
FDCCH Real Time 9-71
Layer 2 9-63
RCCH 9-137
RDCH 9-133
RDCH Real Time 9-140

Date Commands, System Time and 2-23
DCCH Mobile Station Simulation Commands 9-334
Setup Commands 9-338

Decode Data Monitor Commands
FTOT 9-27
FOCC 9-10
FVC 9-22
RDCT 9-49
RECC 9-42
RVC 9-46

*DDT Command 2-34, 3-12
Decision Points, Macro 2-12
Defaults Commands (HOST Only), System 2-23

DELAY Command 3-30
Deviation Meter Commands
Peak 6-101
RMS 6-120

Digital Multimeter Commands 6-115

Display
Screen See Color Display
Sound Control and 2-18
Displays, Soft Function Key 2-20
Distortion Meter Commands 6-106

*DMC Command 2-10, 2-34, 3-13

DMM
Commands 6-115
Ranges and Units 3-116

DO UNTIL Command 3-31

DRAW Command 2-21, 3-33

Duplex Commands 6-40
Receiver 6-50
Transmitter 6-41

Duplex Examples, Remote 6-55

E

e (Scientific Notation) 2-6

E-BCCH and F-BCCH, User Defined Optional 9-283
Message Types and Information Elements
for the

EDIT:COLOR:LETTER Command 2-18, 3-33
and query

EDIT:COLOR:MENU Command and query 2-18, 3-34

EDIT:COLOR:SOFT:BOX Command 2-18, 3-34
and query

EDIT:COLOR:SOFT:LETTER Command 2-18, 3-35
and query

EDIT:COLOR:SOFT:SELECT Command 2-18, 3-35
and query

ELIF ELSE ENDIF, IF Command

ELLIPSE Command 2-10, 2-21, 3-36
Width to Height Equations 3-37
ELSE ENDIF, IF Command 2-33, 3-52
ELSE ENDIF, IF ELIF Command 2-31, 3-51

EMG Command and query 2-34, 3-14
Emulation Program Setup, RS-232 Terminal 4-2
Enable Register Preset Conditions 3-77
END Command 2-11, 3-38
ENDING CASE, CASE OF OTHERWISE

Command 3-24
ENDIF, IF Command 2-12, 3-50
ENDIF, IF ELIF ELSE Command 2-31, 3-51
ENDIF, IF ELSE Command 2-33, 3-52
Equations, Ellipse Width to Height 3-37

ENTER:TEXT Command 2-21, 3-36
Error
System Command 2-23
Messages, Flash Memory 9-169, 9-393

*ESR? Query 2-25, 2-31, 2-34, 3-14

EVENT Command 2-31, 2-34, 3-15
Event Status Register, Standard 2-31
Example TMAC Program 4-5
with Multiple Macros 4-6

Examples
BOX 3-23
Graphic 2-21
ICON 2-22
KEYPAD:ERASE 3-57
KEYPAD:LABEL 3-58
KEYPAD:SOFT 3-59

OMT 10-12

Page and VCDES Message in OMTs 10-19
Primary and Secondary Cycles 9-158
Program See Program Examples

Remote
AF Generator 6-65
AMPS/NAMPS Cell Site Monitor 6-163
Duplex 6-55
Cilloscope 6-76
Receiver 6-34
RF Generator 6-15
Spectrum Analyzer 6-81
Soft Function Key 2-26
TMAC Program 4-5
Window 2-19

EXEC Command 2-11, 3-39
Executing TMAC Program 4-4, 4-7
From HOST Front Panel 4-4, 4-7
From RS-232 Terminal 2-8, 3-39

Index-2
F

F-BCCH and E-BCCH: User Defined Optional Message Types and Information Elements 9-283
for the
F-BCCH Simulation Commands 9-224
FALSE Command 2-8, 3-40
FDCH Data Monitor 9-59
Setup Commands 9-59
FDCH Real Time Data Monitor 9-71
Feedback Response, Shared Channel 9-223
Fetch Commands, Initiate and 6-172
Flash Fill and Macro Name Relationship 4-6
Flash Memory
Commands 6-165, 9-392
Error Messages 6-169, 9-393
Storing Uploaded TMAC Program into 4-6
FLOOR Command 2-8, 3-40
FLUSH Command 3-41
FOR NEXT Command 2-15, 3-42
FORGET Command 2-10, 2-11, 3-44
FORMAT Commands 2-6, 3-45
Keys, Numeric 2-6
Forward Control Channel (FOCC)
Monitor Commands 9-4
Decode Data 9-10
Raw Data 9-16
Program Examples 10-1
Monitoring
Decoded Data 10-1
Raw Data 10-2
Simulation
Mobile Station Control Messages 9-212
Overhead Enable Commands 9-218
Overhead Message Parameters 9-157
Process and Handoff (FOCC) 9-168
Forward Digital Traffic Channel (FDTC)
Modulation Accuracy Commands 9-391
Monitor Commands 9-26
Decode Data 9-27
Raw Data 9-39
Program Examples 10-5
Monitoring
Decoded Data 10-5
IS-54 Raw Data 10-7
Raw Data 10-6
Simulation Commands 9-178
Forward Voice Channel (FVC)
Monitor Commands 9-20
Decode Data 9-22
Raw Data 9-25
Program Examples 10-3
Monitoring
Decoded Data 10-3
Raw Data 10-4
Simulation Commands 9-178
FREQ:BAND Command 9-3
Frequency Control (HOST Only) 2-27
Frequency Error Meter Commands 6-96
Front Panel
Executing TMAC Program from HOST 4-7
Keys and Keycodes 9-1
Function
Generator See AF Generator
Soft Key
Displays 2-20
Example 2-20
Functions 2-9
Mathematical 2-8

G

General TMAC Commands 3-1
Generation, HOST RQS Bit and SRQ 2-30
Generator
RACH Message 9-343
RDCCH Raw 9-340
AF 6-55
Remote Examples 6-65
Function See AF Generator
RF 6-6
Remote Examples 6-15
Generic Measure Commands 6-170, 9-392
Global Action Overhead Messages 9-206
GPIB Operation
HOST Only 2-24
Sp Tst Only 2-25
Program Example 10-23
GPIB Print Control (HOST Only) 2-27
GPIB:ADDRESS Command 2-24, 3-46
GPIB:MASK Command 2-24, 3-46
GPIB:SRQ Command 2-24, 3-46
Graphic Examples 2-21
Graphics and Text 2-21

H

Handoff and Process (FOCC) 9-168
HEIGHT Command 2-21, 3-47
HFLUSH Command 3-47
Host Commands (Sp Tst) 9-2
HOST 9-2
RQS Bit and SRQ Generation 2-30
Setup Commands (RS-232) 6-1
HOST Specific
TMAC Commands 6-1
TMAC Quick Reference List 5-1
HPRINT Command 2-21, 3-48

Index-3
IS-136 Information Elements (cont)

Encodes (cont)

Called Party Number

Address 9-366
Address Encoding 9-365
Numbering Plan ID 9-366
Type of Number 9-366

Called Party Subaddress

RACH
Address (Subaddressing) 9-367
Enable 9-383
Odd/Even Indicator 9-367
Type of Subaddress 9-367
SPACH
Address (Subaddressing) 9-309
Enable 9-329
Length 9-309
Odd/Even Indicator 9-309
Reserved 9-309
Type of Subaddress 9-309

Calling Party Number

RACH
Address 9-368
Address Encoding 9-368
Enable 9-382
Numbering Plan ID 9-367
Type of Number 9-367
SPACH
Address 9-310
Address Encoding 9-310
Enable 9-329
Numbering Plan ID 9-310
Type of Number 9-310

Calling Party Subaddress

RACH
Address (Subaddressing) 9-369
Enable 9-383
Length 9-369
Odd/Even Indicator 9-369
Type of Subaddress 9-369
SPACH
Address (Subaddressing) 9-311
Enable 9-329
Length 9-311
Odd/Even Indicator 9-311
Reserved 9-311
Type of Subaddress 9-311

Capability Request 9-237

Cause
Registration Reject 9-324
Release 9-325
Reorder/Intercept 9-325

CBN_High
CBN_High 9-229
Enable 9-245

Call barred 9-233
CHAN
E-BCCH 9-279
SPACH 9-299
## IS-136 Information Elements (cont)

<table>
<thead>
<tr>
<th>Encode (cont)</th>
<th>IS-136 Information Elements (cont)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Confirmed Message Type</strong></td>
<td>9-379</td>
</tr>
<tr>
<td><strong>Count</strong></td>
<td>9-357</td>
</tr>
<tr>
<td><strong>Custom Control</strong></td>
<td>9-364</td>
</tr>
<tr>
<td><strong>E-BCCH</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>F-BCCH</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Custom Control</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>RACH</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Custom Control</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>SPACH</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Custom Control</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-365</td>
</tr>
<tr>
<td><strong>Data Mode</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Acked Data</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>CRC</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Data Part</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>PM_D</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>RLP</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>SAP</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Data Privacy Mode Map</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>E-BCCH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>F-BCCH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Debug Display Allowed</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>DELAY</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Delay Interval Compensation Mode</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>E-BCCH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>SPACH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Dereg</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Directory Address</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Address</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Address Encoding</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Numbering Plan ID</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Type of Number</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Directory Subaddress</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Address (Subaddressing)</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Odd/Even Indicator</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Reserved</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Type of Subaddress</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Display</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>RACH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Display Character</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>SPACH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Display Character</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Length</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>DMAC</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Double Rate DTC Support</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>DTX Support</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>DTX</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>DVC</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>F-BCCH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>SPACH</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Emergency Call</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>ESN</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Extended Hyperframe Counter</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>EHFC</strong></td>
<td>9-366</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>9-366</td>
</tr>
</tbody>
</table>

**Encode (cont)**

<p>| <strong>FACH/SACCH ARQ Map</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>Firmare Vintage</strong> | 9-277 |
| <strong>Forced Re-registration</strong> | 9-277 |
| <strong>FOREG</strong> | 9-277 |
| <strong>G3-Fax Support</strong> | 9-277 |
| <strong>Half-Rate DTC Support</strong> | 9-277 |
| <strong>Hyperband Info</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Hyperband</strong> | 9-277 |
| <strong>SPACH</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Hyperband</strong> | 9-277 |
| <strong>Hyperframe Counter</strong> | 9-277 |
| <strong>Initial Selection Control</strong> | 9-277 |
| <strong>IRA Support</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>RACH</strong> | 9-277 |
| <strong>LAREG</strong> | 9-277 |
| <strong>Last Try</strong> | 9-277 |
| <strong>RACH</strong> | 9-277 |
| <strong>SPACH</strong> | 9-277 |
| <strong>LTM Measurement</strong> | 9-277 |
| <strong>BER</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Full Measurement</strong> | 9-277 |
| <strong>LT_RSS</strong> | 9-277 |
| <strong>WER</strong> | 9-277 |
| <strong>MACA_B_CONTROL</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>Control</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>Control</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>MACA_LIST</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>CHAN</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Number of MACA Channels</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>CHAN</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Number of MACA Channels</strong> | 9-277 |
| <strong>MACA_LIST (Other Hyperband)</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>CHAN</strong> | 9-277 |
| <strong>Enable</strong> | 9-277 |
| <strong>Hyperband</strong> | 9-277 |
| <strong>Number of MACA Channels</strong> | 9-277 |
| <strong>MACA_STATUS</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>MACA_TYPE</strong> | 9-277 |
| <strong>E-BCCH</strong> | 9-277 |
| <strong>F-BCCH</strong> | 9-277 |
| <strong>Manufacturer Code</strong> | 9-277 |</p>
<table>
<thead>
<tr>
<th>IS-136 Information Elements (cont)</th>
<th>IS-136 Information Elements (cont)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Encode (cont)</strong></td>
<td><strong>Encode (cont)</strong></td>
</tr>
<tr>
<td>Max Busy/Reserved</td>
<td>Access Parameters</td>
</tr>
<tr>
<td>Max Retries</td>
<td>BSME Message Delivery</td>
</tr>
<tr>
<td>Max Stop Counter</td>
<td>Control Channel Selection Parameters</td>
</tr>
<tr>
<td>MAX_SUPPORTED_PFC</td>
<td>DCCH Structure</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>Mobile Assisted Channel Allocation</td>
</tr>
<tr>
<td>RACH</td>
<td>Overload Class</td>
</tr>
<tr>
<td>MEM</td>
<td>Registration Parameters</td>
</tr>
<tr>
<td>Menu Map</td>
<td>Service Menu</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>SOC Message Delivery</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>SOC/BSMC Identification</td>
</tr>
<tr>
<td>Message Center Address</td>
<td>System Identity</td>
</tr>
<tr>
<td>RACH</td>
<td>R-DATA</td>
</tr>
<tr>
<td>Address</td>
<td>R-DATA ACCEPT</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>R-DATA REJECT</td>
</tr>
<tr>
<td>Enable</td>
<td>Registration</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>Serial Number</td>
</tr>
<tr>
<td>Type of Number</td>
<td>SOC Message Delivery</td>
</tr>
<tr>
<td>SPACH</td>
<td>SPACH Confirmation</td>
</tr>
<tr>
<td>Address</td>
<td>SSD Update Confirmation</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>Test Registration</td>
</tr>
<tr>
<td>Enable</td>
<td>Unique Challenge Order Confirmation</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>SPACH</td>
</tr>
<tr>
<td>Type of Number</td>
<td>Analog Voice Channel Designation</td>
</tr>
<tr>
<td>Message Encryption Algorithm Map</td>
<td>Audit Order</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>Base Station Challenge Order</td>
</tr>
<tr>
<td>Domain Map</td>
<td>BSME Message Delivery</td>
</tr>
<tr>
<td>Encryption Algorithms</td>
<td>Capability Request</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>Digital Traffic Channel Designation</td>
</tr>
<tr>
<td>Domain Map</td>
<td>Directed Retry</td>
</tr>
<tr>
<td>Encryption Algorithms</td>
<td>Message Waiting</td>
</tr>
<tr>
<td>Message Encryption Key Map</td>
<td>Page</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>Parameter Update</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>R-DATA</td>
</tr>
<tr>
<td>Message Encryption Mode</td>
<td>R-DATA ACCEPT</td>
</tr>
<tr>
<td>RACH</td>
<td>R-DATA REJECT</td>
</tr>
<tr>
<td>Enable</td>
<td>Registration Accept</td>
</tr>
<tr>
<td>Enable DCCH MEM</td>
<td>Registration Reject</td>
</tr>
<tr>
<td>MEA</td>
<td>Release</td>
</tr>
<tr>
<td>MED</td>
<td>Reorder/Intercept</td>
</tr>
<tr>
<td>MEK</td>
<td>SOC Message Delivery</td>
</tr>
<tr>
<td>SPACH</td>
<td>SPACH Notification</td>
</tr>
<tr>
<td>Enable</td>
<td>SSD Update Order</td>
</tr>
<tr>
<td>MEA</td>
<td>Test Registration Response</td>
</tr>
<tr>
<td>MED</td>
<td>Unique Challenge Order</td>
</tr>
<tr>
<td>MEK</td>
<td>User Alert</td>
</tr>
<tr>
<td>Message Waiting Info</td>
<td>Message Waiting</td>
</tr>
<tr>
<td>Alternate RCI Info</td>
<td>Number of Messages Waiting</td>
</tr>
<tr>
<td>BSME Message Delivery</td>
<td>Number of Values</td>
</tr>
<tr>
<td>Emergency Information Broadcast</td>
<td>Type of Message Waiting</td>
</tr>
<tr>
<td>Mobile Assisted Channel Allocation</td>
<td></td>
</tr>
<tr>
<td>Neighbor Cell</td>
<td></td>
</tr>
<tr>
<td>Neighbor Service Info</td>
<td></td>
</tr>
<tr>
<td>Regulatory Configuration</td>
<td></td>
</tr>
<tr>
<td>Service Menu</td>
<td></td>
</tr>
<tr>
<td>SOC Message Delivery</td>
<td></td>
</tr>
<tr>
<td>SOC/BSMC Identification</td>
<td></td>
</tr>
<tr>
<td>Time and Date</td>
<td></td>
</tr>
</tbody>
</table>

Index-6
IS-136 Information Elements (cont)

Encode (cont)

Mobile Country Code
E-BCCH Enable 9-282
Mobile Country Code 9-279
F-BCCH Enable 9-245
Mobile Country Code 9-239
Model Number 9-359
MS_ACC_PWR 9-231
MSID Assignment Enable 9-332
IDT 9-320
MSID 9-321

Neighbor Cell List (Analog)

CELLTYPE 9-261
CHAN 9-259
DCC 9-259
DELAY 9-260
Directed Retry Channel 9-261
Enable 9-280
HL_FREQ 9-260
MS_ACC_PWR 9-262
Network Type 9-261
Number of Analog Neighbor Cells 9-259
Protocol Version 9-259
RESEL_OFFSET 9-260
RSS_ACC_MIN 9-262
SS_SUFF 9-260

Neighbor Cell List (Other Hyperband)

CELL_SYNC 9-254
CELLTYPE 9-254
CHAN 9-252
DELAY 9-253
Directed Retry Channel 9-255
DVCC 9-252
Enable 9-280
HL_FREQ 9-253
MS_ACC_PWR 9-255
Network Type 9-254
Number of Neighbor Cells 9-262
Protocol Version 9-252
PSID/RSSID Indicator 9-256
PSID/RSSID Support 9-257
PSID/RSSID Support LENGTH 9-256
RESEL_OFFSET 9-253
RSS_ACC_MIN 9-255
SS_SUFF 9-253

Neighbor Cell List (TDMA)

CELL_SYNC 9-254
CELLTYPE 9-254
CHAN 9-252
DELAY 9-253
Directed Retry Channel 9-255
DVCC 9-252
Enable 9-280
HL_FREQ 9-253
MS_ACC_PWR 9-255
Network Type 9-254
Number of TDMA Neighbor Cells 9-252
Protocol Version 9-252
PSID/RSSID Indicator 9-256
PSID/RSSID Support 9-257
PSID/RSSID Support LENGTH 9-256
RESEL_OFFSET 9-253
RSS_ACC_MIN 9-255
SS_SUFF 9-253
Network Type 9-238

IS-136 Information Elements (cont)

Encode (cont)

Non-Public Probability Blocks
E-BCCH Enable 9-279
Non-Public Block Map 9-251
Non-Public Map Length 9-251
F-BCCH Enable 9-246
Non-Public Block Map 9-229
Non-Public Map Length 9-229
Non-Public Registration Control Enable 9-246
Non-Public Registration Control 9-226
Number of E-BCCH 9-226
Number of F-BCCH 9-226
Number of Non-PCH Subchannel Slots 9-227
Number of Reserved Slots 9-227
Number of S-BCCH 9-227
OLC 9-242
PCH_DISPLACEMENT 9-228
PDRCH 9-236
PFC Assignment Enable 9-331
PFC Minus One 9-320
PFC Minus One 9-356
PFC Request Enable 9-385
PFC Minus One 9-378
PFM_DIRECTION 9-228
Present RNUM Enable 9-247
RNUM 9-237
Primary Superframe Indicator 9-227
Protocol Version F-BCCH 9-236
RACH 9-258
SPACH 9-299
PSID/RSSID Available Enable 9-332
Number of PSID/RSSID 9-321
PSID/RSSID Type Indicator 9-321
PSID/RSSID Value 9-321
PSID/RSSID Map RACH 9-356
SPACH 9-322
PSID/RSSID Set Enable 9-247
Number of PSID/RSSID 9-238
PSID/RSSID Type Indicator 9-239
PSID/RSSID Value 9-239
SOC 9-238
PUREG 9-236
R-Cause RACH 9-372
SPACH 9-324
R-DATA Message Length 9-233
R-Data Unit RACH Higher Layer Protocol Data Unit 9-371
Higher Protocol Identifier 9-371
Length Indicator 9-371
SPACH Higher Layer Protocol Data Unit 9-313
Higher Protocol Identifier 9-313
Length Indicator 9-313
R-Transaction Identifier RACH 9-370
SPACH 9-312

Index-7
<table>
<thead>
<tr>
<th>IS-136 Information Elements (cont)</th>
<th>Encode (cont)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I (cont)</td>
<td></td>
</tr>
<tr>
<td><strong>Encode (cont)</strong></td>
<td></td>
</tr>
<tr>
<td>RAND</td>
<td>9-230</td>
</tr>
<tr>
<td>RANDBS</td>
<td>9-357</td>
</tr>
<tr>
<td>RANDC</td>
<td>9-357</td>
</tr>
<tr>
<td>RANDSD</td>
<td>9-326</td>
</tr>
<tr>
<td>RANDU</td>
<td>9-327</td>
</tr>
<tr>
<td>RCF and AUTH</td>
<td>9-329</td>
</tr>
<tr>
<td>RCI</td>
<td>9-270</td>
</tr>
<tr>
<td>REG Period</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-247</td>
</tr>
<tr>
<td>REGRPER</td>
<td>9-237</td>
</tr>
<tr>
<td>REGH</td>
<td>9-235</td>
</tr>
<tr>
<td>REGID Parameters</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-247</td>
</tr>
<tr>
<td>REGID</td>
<td>9-237</td>
</tr>
<tr>
<td>REGID_PER</td>
<td>9-237</td>
</tr>
<tr>
<td>Registration Type</td>
<td>9-377</td>
</tr>
<tr>
<td>REGR</td>
<td>9-235</td>
</tr>
<tr>
<td>Reject Time</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-332</td>
</tr>
<tr>
<td>Lower Time Boundary in 100 SF</td>
<td>9-324</td>
</tr>
<tr>
<td>Upper Time Boundary in 100 SF</td>
<td>9-324</td>
</tr>
<tr>
<td>Request Number</td>
<td>9-312</td>
</tr>
<tr>
<td>RF Channel Allocation</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-281</td>
</tr>
<tr>
<td>First Channel</td>
<td>9-271</td>
</tr>
<tr>
<td>Last Channel</td>
<td>9-271</td>
</tr>
<tr>
<td>Number of Channel Groups</td>
<td>9-270</td>
</tr>
<tr>
<td>RNUM List</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-331</td>
</tr>
<tr>
<td>RNUM</td>
<td>9-320</td>
</tr>
<tr>
<td>Number of RNUMs</td>
<td>9-320</td>
</tr>
<tr>
<td>RSS_ACC_MIN</td>
<td>9-231</td>
</tr>
<tr>
<td>S</td>
<td>9-230</td>
</tr>
<tr>
<td>SCS</td>
<td>9-303</td>
</tr>
<tr>
<td>SCANINTERVAL</td>
<td>9-234</td>
</tr>
<tr>
<td>Scanning Option Indicator</td>
<td>9-234</td>
</tr>
<tr>
<td>SCC</td>
<td>9-299</td>
</tr>
<tr>
<td>SCM</td>
<td>9-358</td>
</tr>
<tr>
<td>Selected PSID/PSID</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-381</td>
</tr>
<tr>
<td>Selected PSID/PSID</td>
<td>9-356</td>
</tr>
<tr>
<td>SERV_SS</td>
<td>9-251</td>
</tr>
<tr>
<td>Service Code</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td>9-363</td>
</tr>
<tr>
<td>SPACH</td>
<td>9-307</td>
</tr>
<tr>
<td>SID</td>
<td>9-279</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-238</td>
</tr>
<tr>
<td>Signal</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-281</td>
</tr>
<tr>
<td>Cadence</td>
<td>9-273</td>
</tr>
<tr>
<td>Duration</td>
<td>9-273</td>
</tr>
<tr>
<td>Pitch</td>
<td>9-272</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-329</td>
</tr>
<tr>
<td>Cadence</td>
<td>9-308</td>
</tr>
<tr>
<td>Duration</td>
<td>9-308</td>
</tr>
<tr>
<td>Pitch</td>
<td>9-308</td>
</tr>
<tr>
<td>Slot Configuration</td>
<td>9-228</td>
</tr>
<tr>
<td>SMS Broadcast Support</td>
<td>9-360</td>
</tr>
<tr>
<td>SMS Map</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-277</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-244</td>
</tr>
<tr>
<td>IS-136 Information Elements (cont)</td>
<td>Encode (cont)</td>
</tr>
<tr>
<td>SOC</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-277</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-244</td>
</tr>
<tr>
<td>RACH</td>
<td>9-379</td>
</tr>
<tr>
<td>SPACH</td>
<td>9-325</td>
</tr>
<tr>
<td>SOC Support</td>
<td>9-350</td>
</tr>
<tr>
<td>Software Vintage</td>
<td>9-359</td>
</tr>
<tr>
<td>SPACH Notification Type</td>
<td>9-326</td>
</tr>
<tr>
<td>SS_SUFF</td>
<td>9-233</td>
</tr>
<tr>
<td>SSD Update Status</td>
<td>9-360</td>
</tr>
<tr>
<td>STM Measurement</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-381</td>
</tr>
<tr>
<td>ST_RSS</td>
<td>9-363</td>
</tr>
<tr>
<td>Number of Values</td>
<td>9-363</td>
</tr>
<tr>
<td>Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-370</td>
</tr>
<tr>
<td>Enable</td>
<td>9-363</td>
</tr>
<tr>
<td>Length</td>
<td>9-370</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-370</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-370</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-300</td>
</tr>
<tr>
<td>Enable</td>
<td>9-328</td>
</tr>
<tr>
<td>Length</td>
<td>9-299</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-300</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-300</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-300</td>
</tr>
<tr>
<td>Subaddressing Support</td>
<td></td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-233</td>
</tr>
<tr>
<td>RACH</td>
<td>9-360</td>
</tr>
<tr>
<td>Supported Frequency Bands</td>
<td>9-360</td>
</tr>
<tr>
<td>SYREG</td>
<td>9-236</td>
</tr>
<tr>
<td>TDM Service Info</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-280</td>
</tr>
<tr>
<td>Service Map</td>
<td>9-258</td>
</tr>
<tr>
<td>Service Map Indicator</td>
<td>9-258</td>
</tr>
<tr>
<td>TDM Neighbor Count</td>
<td>9-258</td>
</tr>
<tr>
<td>TDM Service Info (Other Hyperbands)</td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>9-280</td>
</tr>
<tr>
<td>Hyperband</td>
<td>9-269</td>
</tr>
<tr>
<td>Service Map</td>
<td>9-258</td>
</tr>
<tr>
<td>Service Map Indicator</td>
<td>9-258</td>
</tr>
<tr>
<td>TDM Neighbor Count</td>
<td>9-258</td>
</tr>
<tr>
<td>Text Message Data Unit</td>
<td></td>
</tr>
<tr>
<td>Encoding Identifier</td>
<td>9-272</td>
</tr>
<tr>
<td>Length Indicator</td>
<td>9-272</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-272</td>
</tr>
<tr>
<td>Short Message Character</td>
<td>9-272</td>
</tr>
<tr>
<td>Time Alignment</td>
<td>9-303</td>
</tr>
<tr>
<td>Time from Jan 1, 1980</td>
<td>9-278</td>
</tr>
<tr>
<td>Time Zone Offset</td>
<td>9-278</td>
</tr>
<tr>
<td>Daylight Savings Indicator</td>
<td>9-278</td>
</tr>
<tr>
<td>Direction</td>
<td>9-278</td>
</tr>
<tr>
<td>Minutes</td>
<td>9-278</td>
</tr>
<tr>
<td>Tone Indicator</td>
<td>9-325</td>
</tr>
<tr>
<td>Triple Rate DTC Support</td>
<td>9-362</td>
</tr>
</tbody>
</table>
### IS-136 Information Elements (cont)

#### Encode (cont)

<table>
<thead>
<tr>
<th>User Destination Address</th>
<th>9-374</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACH Address</td>
<td>9-374</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-384</td>
</tr>
<tr>
<td>Enable</td>
<td>9-374</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-384</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-374</td>
</tr>
<tr>
<td>SPACH Address</td>
<td>9-315</td>
</tr>
<tr>
<td>Address</td>
<td>9-315</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-330</td>
</tr>
<tr>
<td>Enable</td>
<td>9-315</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-315</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-315</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User Destination Subaddress</th>
<th>9-375</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACH (Subaddressing)</td>
<td>9-384</td>
</tr>
<tr>
<td>Address</td>
<td>9-375</td>
</tr>
<tr>
<td>Enable</td>
<td>9-375</td>
</tr>
<tr>
<td>Length</td>
<td>9-375</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-375</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-375</td>
</tr>
<tr>
<td>SPACH (Subaddressing)</td>
<td>9-316</td>
</tr>
<tr>
<td>Address</td>
<td>9-316</td>
</tr>
<tr>
<td>Enable</td>
<td>9-316</td>
</tr>
<tr>
<td>Length</td>
<td>9-316</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-316</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-316</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-316</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User Group RACH</th>
<th>9-384</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable</td>
<td>9-384</td>
</tr>
<tr>
<td>User Group ID</td>
<td>9-373</td>
</tr>
<tr>
<td>User Group Status</td>
<td>9-373</td>
</tr>
<tr>
<td>User Group Type</td>
<td>9-372</td>
</tr>
<tr>
<td>SPACH (Subaddressing)</td>
<td>9-331</td>
</tr>
<tr>
<td>Enable</td>
<td>9-331</td>
</tr>
<tr>
<td>User Group ID</td>
<td>9-317</td>
</tr>
<tr>
<td>User Group Status</td>
<td>9-317</td>
</tr>
<tr>
<td>User Group Type</td>
<td>9-317</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User Group E-BCCH</th>
<th>9-277</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable</td>
<td>9-244</td>
</tr>
<tr>
<td>User Group Support</td>
<td>9-351</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User Originating Address RACH</th>
<th>9-376</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>9-376</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-384</td>
</tr>
<tr>
<td>Enable</td>
<td>9-376</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-376</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-376</td>
</tr>
<tr>
<td>SPACH (Subaddressing)</td>
<td>9-318</td>
</tr>
<tr>
<td>Address</td>
<td>9-318</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-318</td>
</tr>
<tr>
<td>Enable</td>
<td>9-318</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-318</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-318</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User Originating Subaddress RACH</th>
<th>9-377</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address (Subaddressing)</td>
<td>9-377</td>
</tr>
<tr>
<td>Enable</td>
<td>9-377</td>
</tr>
<tr>
<td>Length</td>
<td>9-377</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-377</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-377</td>
</tr>
<tr>
<td>SPACH (Subaddressing)</td>
<td>9-319</td>
</tr>
<tr>
<td>Enable</td>
<td>9-319</td>
</tr>
<tr>
<td>Length</td>
<td>9-319</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-319</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-319</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-319</td>
</tr>
</tbody>
</table>

### IS-136 Information Elements (cont)

#### Decode

<table>
<thead>
<tr>
<th>800 MHz Analog Speech Support</th>
<th>9-152</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Burst Size</td>
<td>9-177</td>
</tr>
<tr>
<td>Additional DCCH Information</td>
<td>9-176</td>
</tr>
<tr>
<td>Alphanumeric PSID/RSID List</td>
<td>9-131</td>
</tr>
<tr>
<td>Display Character</td>
<td>9-131</td>
</tr>
<tr>
<td>Length of Alphanumeric PSID/RSID List</td>
<td>9-131</td>
</tr>
<tr>
<td>Name</td>
<td>9-131</td>
</tr>
<tr>
<td>Alphanumeric System ID</td>
<td>9-131</td>
</tr>
<tr>
<td>Display Character</td>
<td>9-131</td>
</tr>
<tr>
<td>Length</td>
<td>9-131</td>
</tr>
<tr>
<td>Async Data Support</td>
<td>9-151</td>
</tr>
<tr>
<td>ATS</td>
<td>9-110</td>
</tr>
<tr>
<td>AUTH</td>
<td>9-76</td>
</tr>
<tr>
<td>AUTHBS</td>
<td>9-109</td>
</tr>
<tr>
<td>AUTHR</td>
<td>9-142</td>
</tr>
<tr>
<td>AUTHU</td>
<td>9-142</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>9-142</td>
</tr>
<tr>
<td>BSMC</td>
<td>9-82</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-98</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-143</td>
</tr>
<tr>
<td>RACH</td>
<td>9-143</td>
</tr>
<tr>
<td>SPACH</td>
<td>9-110</td>
</tr>
<tr>
<td>BSMC Support</td>
<td>9-151</td>
</tr>
<tr>
<td>C-Number</td>
<td>9-145</td>
</tr>
<tr>
<td>Address</td>
<td>9-145</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-145</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-145</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-145</td>
</tr>
<tr>
<td>Called Party</td>
<td>9-114</td>
</tr>
<tr>
<td>Address</td>
<td>9-114</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-114</td>
</tr>
<tr>
<td>Length</td>
<td>9-114</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-114</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-114</td>
</tr>
<tr>
<td>Called Party Number Address</td>
<td>9-143</td>
</tr>
<tr>
<td>Address</td>
<td>9-143</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-143</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-143</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-143</td>
</tr>
</tbody>
</table>

Index-9
<table>
<thead>
<tr>
<th>I (cont) IS-136 Information Elements (cont)</th>
<th>I (cont) IS-136 Information Elements (cont)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decode (cont)</td>
<td>Decode (cont)</td>
</tr>
<tr>
<td>Called Party Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td>Data Mode</td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>Acked Data</td>
</tr>
<tr>
<td>Length</td>
<td>CRC</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>Data Part</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>PM_D</td>
</tr>
<tr>
<td>SPACH</td>
<td>RLP</td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>SAP</td>
</tr>
<tr>
<td>Length</td>
<td>Data Privacy Mode Map</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>E-BCCH</td>
</tr>
<tr>
<td>Reserved</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>Debug Display Allowed</td>
</tr>
<tr>
<td>Calling Party Number</td>
<td>DELAY</td>
</tr>
<tr>
<td>RACH</td>
<td>Delay Interval Compensation Mode</td>
</tr>
<tr>
<td>Address</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>Length</td>
<td>DERC</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>Directory Address</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>Address</td>
</tr>
<tr>
<td>Type of Number</td>
<td>Address Encoding</td>
</tr>
<tr>
<td>SPACH</td>
<td>Length</td>
</tr>
<tr>
<td>Address</td>
<td>Numbering Plan ID</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>Type of Number</td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td></td>
</tr>
<tr>
<td>Type of Number</td>
<td></td>
</tr>
<tr>
<td>Calling Party Number Presentation</td>
<td></td>
</tr>
<tr>
<td>Indicator</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Presentation Indicator</td>
<td></td>
</tr>
<tr>
<td>Screening Indicator</td>
<td></td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Presentation Indicator</td>
<td></td>
</tr>
<tr>
<td>Screening Indicator</td>
<td></td>
</tr>
<tr>
<td>Calling Party Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>DMAC</td>
</tr>
<tr>
<td>Length</td>
<td>Double Rate DTC Support</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>DTX Support</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>DVCO</td>
</tr>
<tr>
<td>SPACH</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>SPACH</td>
</tr>
<tr>
<td>Length</td>
<td>Emergency Call</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>ESN</td>
</tr>
<tr>
<td>Reserved</td>
<td>Extended Hyperframe Counter</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>FACH/SACh ARQ Map</td>
</tr>
<tr>
<td>Capability Request</td>
<td></td>
</tr>
<tr>
<td>Cause</td>
<td></td>
</tr>
<tr>
<td>Registration Reject</td>
<td></td>
</tr>
<tr>
<td>Release</td>
<td></td>
</tr>
<tr>
<td>Recorder/Intercept</td>
<td></td>
</tr>
<tr>
<td>CBN_High</td>
<td></td>
</tr>
<tr>
<td>Call Barred</td>
<td></td>
</tr>
<tr>
<td>CHAN</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td></td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Confirmed Message Type</td>
<td></td>
</tr>
<tr>
<td>COUNT</td>
<td></td>
</tr>
<tr>
<td>Custom Control</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
<tr>
<td>F-BCCH</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
</tbody>
</table>

Index-10
<table>
<thead>
<tr>
<th>IS-136 Information Elements (cont)</th>
<th>IS-136 Information Elements (cont)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Decode (cont)</strong></td>
<td><strong>Decode (cont)</strong></td>
</tr>
<tr>
<td>LTM Measurement</td>
<td>Message Encryption Mode</td>
</tr>
<tr>
<td>BER</td>
<td>RACH</td>
</tr>
<tr>
<td>Full Measurement Indicator</td>
<td>MEA</td>
</tr>
<tr>
<td>LT_RSS</td>
<td>MED</td>
</tr>
<tr>
<td>WER</td>
<td>MEK</td>
</tr>
<tr>
<td>MACA_8_CONTROL</td>
<td>SPACH</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>MEA</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>MED</td>
</tr>
<tr>
<td>MACA_LIST</td>
<td>MEK</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>Message Type</td>
</tr>
<tr>
<td>CHAN</td>
<td>E-BCCH</td>
</tr>
<tr>
<td>Number of MACA Channels</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>RACH</td>
</tr>
<tr>
<td>CHAN</td>
<td>MACA_LIST (Other Hyperband)</td>
</tr>
<tr>
<td>Number of MACA Channels</td>
<td>E-BCCH</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>CHAN</td>
</tr>
<tr>
<td>Hyperband</td>
<td>Number of MACA Channels</td>
</tr>
<tr>
<td>Number of MACA Channels</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>MACA_STATUS</td>
<td>CHAN</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>Hyperband</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>Number of MACA Channels</td>
</tr>
<tr>
<td>MACA_TYPE</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>Max Busy/Reserved</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-124</td>
</tr>
<tr>
<td>Manufacturer Code</td>
<td>9-146</td>
</tr>
<tr>
<td>Max Busy/Reserved</td>
<td>9-77</td>
</tr>
<tr>
<td>Max Repetitions</td>
<td>Max Retries</td>
</tr>
<tr>
<td>Max Retries</td>
<td>9-77</td>
</tr>
<tr>
<td>Max Stop Counter</td>
<td>MAX_SUPPORTED_PFC</td>
</tr>
<tr>
<td>MAX_SUPPORTED_PFC</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>MEM</td>
</tr>
<tr>
<td>MEM</td>
<td>Menu Map</td>
</tr>
<tr>
<td>Menu Map</td>
<td>E-BCCH</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>Message Center Address</td>
</tr>
<tr>
<td>Message Center Address</td>
<td>RACH</td>
</tr>
<tr>
<td>Address</td>
<td>Address</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>Address Encoding</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-148</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-148</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-151</td>
</tr>
<tr>
<td>Message Encryption Algorithm Map</td>
<td>Message Encryption Key Map</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>E-BCCH</td>
</tr>
<tr>
<td>Domain Map</td>
<td>F-BCCH</td>
</tr>
<tr>
<td>Encryption Algorithms</td>
<td>Domain Map</td>
</tr>
<tr>
<td>Encryption Algorithms</td>
<td>Encryption Algorithms</td>
</tr>
<tr>
<td>Message Encryption Key Map</td>
<td>9-102</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-85</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-85</td>
</tr>
</tbody>
</table>
I (cont)

IS-136 Information Elements (cont)
Decode (cont)

Neighbor Cell List (TDMA)
   CELL_SYNC  9-88
   CELL_TYPE  9-39
   CHAN       9-87
   DELAY      9-88
   Directed Retry Channel 9-89
   DVCC       9-88
   HL_FREQ    9-88
   MS_ACC_PWR 9-89
   Network Type 9-89
   Number of TDMA Neighbor Cells 9-87
   Protocol Version 9-87
   PSID/RSID Indicator 9-90
   PSID/RSID Support 9-90
   PSID/RSID Support LENGTH 9-90
   RESEL_OFFSET 9-88
   RSS_ACC_MIN 9-89
   SS_SUFF    9-88
   Network Type 9-81
   Non-Public Probability Blocks
   E-BCCH
      Non-Public Block Map 9-87
      Non-Public Block Map Length 9-87
   F-BCCH
      Non-Public Block Map 9-76
      Non-Public Block Map Length 9-76
   Non-Public Registration Control 9-75
   Number of E-BCCH 9-74
   Number of F-BCCH 9-74
   Number of Non-PCH Subchannel Slots 9-74
   Number of Reserved Slots 9-74
   Number of S-BCCH 9-74
   OLC 9-84
   PCH_DISPLACEMENT 9-75
   POREG 9-79
   PFC Assignment 9-125
   PFC Minus One 9-142
   PFC Request 9-149
   PMF_DIRECTION 9-75
   Present RNUM 9-80
   Primary Superframe Indicator 9-74
   Protocol Discriminator
   E-BCCH 9-86
   F-BCCH 9-73
   RACH 9-142
   SPACH 9-107
   Protocol Version
   F-BCCH 9-81
   RACH 9-149
   SPACH 9-108
   PSID/RSID Available
   Number of PSID/RSID 9-126
   PSID/RSID Type Indicator 9-126
   PSID/RSID Value 9-126
   PSID/RSID Map
   RACH 9-149
   SPACH 9-126
   PSID/RSID Set
   Number of PSID/RSID 9-81
   PSID/RSID Type Indicator 9-81
   PSID/RSID Value 9-81
   QC 9-81
   PRC 9-79
   R-Cause
   RACH 9-150
   SPACH 9-129

I (cont)

IS-136 Information Elements (cont)
Decode (cont)

R-DATA Message Length 9-77
R-DATA Unit
   RACH
      Higher Layer Protocol Data Unit 9-150
      Higher Protocol Identifier 9-150
      Length Indicator 9-150
   SPACH
      Higher Layer Protocol Data Unit 9-118
      Higher Protocol Identifier 9-118
      Length Indicator 9-118
R-Transaction Identifier
   RACH 9-150
   SPACH 9-118
   RAND 9-76
   RANDBS 9-150
   RANDC 9-150
   RANDSSID 9-130
   RANDU 9-132
   RCI 9-97
   REG_Period 9-80
   REGH 9-79
   REGID_Parameters
      REGID 9-80
      REGID_PER 9-80
   Registration_Type 9-150
   REGR 9-79
   Reject_Time
      Lower Time Boundary in 100 SF 9-129
      Upper Time Boundary in 100 SF 9-129
   Request_Number 9-118
   RF Channel Allocation
      First Channel 9-98
      Last Channel 9-98
      Number of Channel Groups 9-98
   RNUM List
      RNUM 9-125
      Number of RNUMs 9-125
      RSS_ACC_MIN 9-77
      S 9-76
      SB 9-110
      SCANINTERVAL 9-78
      Scanning Option Indicator 9-78
      SCG 9-107
      SCM 9-150
      Selected PSID/RSID 9-149
      SERV_SS 9-86
   Service Code
      RACH 9-150
      SPACH 9-113
   SID
      E-BCCH 9-103
      F-BCCH 9-81
   Signal
      Cadence 9-99
      Duration 9-99
      Pitch 9-99
      SPACH
      Cadence 9-113
      Duration 9-113
      Pitch 9-113
      Slot_Configuration 9-75
      SMS_Broadcast_Support 9-151
      SMS_Map
         E-BCCH 9-102
         F-BCCH 9-85

Index-12
Decode (cont)

**IS-136 Information Elements (cont)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOC</td>
<td>9-102</td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-85</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-97</td>
</tr>
<tr>
<td>RACH</td>
<td>9-150</td>
</tr>
<tr>
<td>SPACH</td>
<td>9-130</td>
</tr>
<tr>
<td>SOC Support</td>
<td>9-151</td>
</tr>
<tr>
<td>Software Vintage</td>
<td>9-155</td>
</tr>
<tr>
<td>SPACH Notification Type</td>
<td>9-130</td>
</tr>
<tr>
<td>SS, SUFF</td>
<td>9-78</td>
</tr>
<tr>
<td>SSD Update Status</td>
<td>9-151</td>
</tr>
<tr>
<td>STM Measurement</td>
<td>9-147</td>
</tr>
<tr>
<td>Number of Values</td>
<td>9-147</td>
</tr>
<tr>
<td>ST, RSS</td>
<td>9-147</td>
</tr>
<tr>
<td>Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-151</td>
</tr>
<tr>
<td>Length</td>
<td>9-151</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-151</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-151</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-108</td>
</tr>
<tr>
<td>Length</td>
<td>9-108</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-108</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-108</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-108</td>
</tr>
<tr>
<td>Subaddressing Support</td>
<td></td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-78</td>
</tr>
<tr>
<td>RACH</td>
<td>9-152</td>
</tr>
<tr>
<td>Supported Frequency Bands</td>
<td>9-152</td>
</tr>
<tr>
<td>SYREG</td>
<td>9-79</td>
</tr>
<tr>
<td>TDMA Service Info</td>
<td></td>
</tr>
<tr>
<td>Service Map</td>
<td>9-91</td>
</tr>
<tr>
<td>Service Map Indicator</td>
<td>9-91</td>
</tr>
<tr>
<td>TDMA Neighbor Count</td>
<td>9-90</td>
</tr>
<tr>
<td>TDMA Service Info (Other Hyperband)</td>
<td></td>
</tr>
<tr>
<td>Hyperband</td>
<td>9-97</td>
</tr>
<tr>
<td>Service Map</td>
<td>9-91</td>
</tr>
<tr>
<td>Service Map Indicator</td>
<td>9-91</td>
</tr>
<tr>
<td>TDMA Neighbor Count</td>
<td>9-90</td>
</tr>
<tr>
<td>Text Message Data Unit</td>
<td></td>
</tr>
<tr>
<td>Encoding Identifier</td>
<td>9-99</td>
</tr>
<tr>
<td>Length Indicator</td>
<td>9-99</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-99</td>
</tr>
<tr>
<td>Short Message Character</td>
<td>9-99</td>
</tr>
<tr>
<td>Time Alignment</td>
<td>9-110</td>
</tr>
<tr>
<td>Time from Jan 1, 1980</td>
<td>9-102</td>
</tr>
<tr>
<td>Time Zone Offset</td>
<td></td>
</tr>
<tr>
<td>Direction</td>
<td>9-103</td>
</tr>
<tr>
<td>Daylight Savings Indicator</td>
<td>9-103</td>
</tr>
<tr>
<td>Minutes</td>
<td>9-103</td>
</tr>
<tr>
<td>Tone Indicator</td>
<td>9-130</td>
</tr>
<tr>
<td>Triple Rate DTC Support</td>
<td>9-152</td>
</tr>
<tr>
<td>User Destination Address</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>9-153</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-153</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-153</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-153</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>9-120</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-120</td>
</tr>
<tr>
<td>Length</td>
<td>9-120</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-120</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-120</td>
</tr>
<tr>
<td>User Destination Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-153</td>
</tr>
<tr>
<td>Length</td>
<td>9-153</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-153</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-153</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-121</td>
</tr>
<tr>
<td>Length</td>
<td>9-121</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-121</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-121</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-121</td>
</tr>
<tr>
<td>User Group</td>
<td></td>
</tr>
<tr>
<td>User Group ID</td>
<td>9-154</td>
</tr>
<tr>
<td>User Group Status</td>
<td>9-154</td>
</tr>
<tr>
<td>User Group Type</td>
<td>9-154</td>
</tr>
<tr>
<td>User Group ID</td>
<td>9-122</td>
</tr>
<tr>
<td>User Group Status</td>
<td>9-122</td>
</tr>
<tr>
<td>User Group Type</td>
<td>9-122</td>
</tr>
<tr>
<td>User Group Map</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-102</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-85</td>
</tr>
<tr>
<td>User Group Support</td>
<td>9-152</td>
</tr>
<tr>
<td>User Originating Address</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>9-154</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-154</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-154</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-154</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>9-123</td>
</tr>
<tr>
<td>Address Encoding</td>
<td>9-123</td>
</tr>
<tr>
<td>Length</td>
<td>9-123</td>
</tr>
<tr>
<td>Numbering Plan ID</td>
<td>9-123</td>
</tr>
<tr>
<td>Type of Number</td>
<td>9-123</td>
</tr>
<tr>
<td>User Originating Subaddress</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-155</td>
</tr>
<tr>
<td>Length</td>
<td>9-155</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-155</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-155</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Address (Subaddressing)</td>
<td>9-124</td>
</tr>
<tr>
<td>Length</td>
<td>9-124</td>
</tr>
<tr>
<td>Odd/Even Indicator</td>
<td>9-124</td>
</tr>
<tr>
<td>Reserved</td>
<td>9-124</td>
</tr>
<tr>
<td>Type of Subaddress</td>
<td>9-124</td>
</tr>
<tr>
<td>VMAC</td>
<td>9-108</td>
</tr>
<tr>
<td>Voice Coder Map</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-101</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-84</td>
</tr>
<tr>
<td>Voice Mode</td>
<td></td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>PM_V</td>
<td>9-148</td>
</tr>
<tr>
<td>VC</td>
<td>9-148</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>PM_V</td>
<td>9-111</td>
</tr>
<tr>
<td>VC</td>
<td>9-111</td>
</tr>
<tr>
<td>Voice Privacy Mode Map</td>
<td></td>
</tr>
<tr>
<td>E-BCCH</td>
<td>9-101</td>
</tr>
<tr>
<td>F-BCCH</td>
<td>9-84</td>
</tr>
</tbody>
</table>
### IS-136 Layer 3 Messages (cont)

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>F-BCCH</td>
<td></td>
</tr>
<tr>
<td>Access Parameters</td>
<td>11-5</td>
</tr>
<tr>
<td>BSMC Message Delivery</td>
<td>11-9</td>
</tr>
<tr>
<td>Control Channel Selection Parameters</td>
<td>11-6</td>
</tr>
<tr>
<td>DCCH Structure</td>
<td>11-4</td>
</tr>
<tr>
<td>Mobile Assisted Channel Allocation (Multi Hyperband)</td>
<td>11-9</td>
</tr>
<tr>
<td>Mobile Assisted Channel Allocation</td>
<td>11-12</td>
</tr>
<tr>
<td>Overload Class</td>
<td>11-10</td>
</tr>
<tr>
<td>Registration Parameters</td>
<td>11-7</td>
</tr>
<tr>
<td>Service Menu</td>
<td>11-10</td>
</tr>
<tr>
<td>SOC Message Delivery</td>
<td>11-11</td>
</tr>
<tr>
<td>SOC/BSMC Identification</td>
<td>11-11</td>
</tr>
<tr>
<td>System Identity</td>
<td>11-8</td>
</tr>
<tr>
<td>E-BCCH</td>
<td></td>
</tr>
<tr>
<td>Alternate RCII Info</td>
<td>11-20</td>
</tr>
<tr>
<td>BSMC Message Delivery</td>
<td>11-16</td>
</tr>
<tr>
<td>Emergency Information Broadcast</td>
<td>11-17</td>
</tr>
<tr>
<td>Mobile Assisted Channel Allocation (Multi Hyperband)</td>
<td>11-26</td>
</tr>
<tr>
<td>Neighbor Cell (Multi Hyperband)</td>
<td>11-21</td>
</tr>
<tr>
<td>Neighbor Service Info</td>
<td>11-20</td>
</tr>
<tr>
<td>Neighbor Service Info (Multi Hyperband)</td>
<td>11-25</td>
</tr>
<tr>
<td>Regulatory Configuration</td>
<td>11-16</td>
</tr>
<tr>
<td>Service Menu</td>
<td>11-18</td>
</tr>
<tr>
<td>SOC Message Delivery</td>
<td>11-19</td>
</tr>
<tr>
<td>SOC/BSMC Identification</td>
<td>11-18</td>
</tr>
<tr>
<td>Time and Date</td>
<td>11-19</td>
</tr>
<tr>
<td>RACH</td>
<td></td>
</tr>
<tr>
<td>Audit Confirmation</td>
<td>11-59</td>
</tr>
<tr>
<td>Authentication</td>
<td>11-60</td>
</tr>
<tr>
<td>Base Station Challenge Order</td>
<td>11-60</td>
</tr>
<tr>
<td>BSMC Message Delivery</td>
<td>11-61</td>
</tr>
<tr>
<td>Capability Report</td>
<td>11-62</td>
</tr>
<tr>
<td>MACA Report</td>
<td>11-64</td>
</tr>
<tr>
<td>Origination</td>
<td>11-65</td>
</tr>
<tr>
<td>Page Response</td>
<td>11-68</td>
</tr>
<tr>
<td>Queue Disconnect</td>
<td>11-77</td>
</tr>
<tr>
<td>R-DATA</td>
<td>11-70</td>
</tr>
<tr>
<td>R-DATA ACCEPT</td>
<td>11-72</td>
</tr>
<tr>
<td>R-DATA REJECT</td>
<td>11-74</td>
</tr>
<tr>
<td>Registration</td>
<td>11-73</td>
</tr>
<tr>
<td>Serial Number</td>
<td>11-74</td>
</tr>
<tr>
<td>SOC Message Delivery</td>
<td>11-75</td>
</tr>
<tr>
<td>SPACH Confirmation</td>
<td>11-75</td>
</tr>
<tr>
<td>SSD Update Order Confirmation</td>
<td>11-76</td>
</tr>
<tr>
<td>Test Registration</td>
<td>11-76</td>
</tr>
<tr>
<td>Unique Challenge Order Confirmation</td>
<td>11-77</td>
</tr>
<tr>
<td>SPACH</td>
<td></td>
</tr>
<tr>
<td>Analog Voice Channel Designation</td>
<td>11-27</td>
</tr>
<tr>
<td>Audit Order</td>
<td>11-28</td>
</tr>
<tr>
<td>Base Station Challenge Order Confirmation</td>
<td>11-29</td>
</tr>
<tr>
<td>BSMC Message Delivery</td>
<td>11-30</td>
</tr>
<tr>
<td>Capability Request</td>
<td>11-31</td>
</tr>
<tr>
<td>Digital Traffic Channel Designation</td>
<td>11-32</td>
</tr>
<tr>
<td>Directed Retry</td>
<td>11-34</td>
</tr>
<tr>
<td>Message Waiting</td>
<td>11-35</td>
</tr>
<tr>
<td>Page</td>
<td>11-36</td>
</tr>
<tr>
<td>Parameter Update</td>
<td>11-39</td>
</tr>
<tr>
<td>Queue Disconnect Ack</td>
<td>11-57</td>
</tr>
<tr>
<td>Queue Update</td>
<td>11-58</td>
</tr>
<tr>
<td>R-DATA</td>
<td>11-40</td>
</tr>
<tr>
<td>R-DATA ACCEPT</td>
<td>11-43</td>
</tr>
<tr>
<td>R-DATA REJECT</td>
<td>11-44</td>
</tr>
</tbody>
</table>

### IS-136 Layer 3 Messages (cont)

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPACH (cont)</td>
<td></td>
</tr>
<tr>
<td>Registration Accept</td>
<td>11-45</td>
</tr>
<tr>
<td>Registration Reject</td>
<td>11-48</td>
</tr>
<tr>
<td>Release</td>
<td>11-49</td>
</tr>
<tr>
<td>Reorder/Intercept</td>
<td>11-50</td>
</tr>
<tr>
<td>SOC Message Delivery</td>
<td>11-51</td>
</tr>
<tr>
<td>SPACH Notification</td>
<td>11-52</td>
</tr>
<tr>
<td>SSD Update Order</td>
<td>11-53</td>
</tr>
<tr>
<td>Test Registration Response</td>
<td>11-54</td>
</tr>
<tr>
<td>Unique Challenge Order</td>
<td>11-55</td>
</tr>
<tr>
<td>User Alert</td>
<td>11-56</td>
</tr>
<tr>
<td>IS-54 Raw Data</td>
<td>9-40</td>
</tr>
</tbody>
</table>

### K

<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEY Command</td>
<td>2-23, 3-55</td>
</tr>
<tr>
<td>KEY? Query</td>
<td>2-23, 3-56</td>
</tr>
<tr>
<td>Keyboard Lock Command</td>
<td>2-24</td>
</tr>
<tr>
<td>KEYPAD:CLAIM Command</td>
<td>2-23, 3-57</td>
</tr>
<tr>
<td>KEYPAD:ERASE Command</td>
<td>2-20, 3-57</td>
</tr>
<tr>
<td>Example</td>
<td>3-57</td>
</tr>
<tr>
<td>KEYPAD:LABEL Command</td>
<td>2-20, 3-58</td>
</tr>
<tr>
<td>Example</td>
<td>3-58</td>
</tr>
<tr>
<td>KEYPAD:SOFT Command</td>
<td>2-20, 3-59</td>
</tr>
<tr>
<td>Example</td>
<td>3-59</td>
</tr>
<tr>
<td>KEYPAD:UNCLAIM Command</td>
<td>2-23, 3-60</td>
</tr>
<tr>
<td>Keys and Keycodes, Front Panel</td>
<td>B-1</td>
</tr>
<tr>
<td>KILL Command</td>
<td>2-15, 3-60</td>
</tr>
</tbody>
</table>

### L

<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layer 2 Data Monitor</td>
<td></td>
</tr>
<tr>
<td>FDCH</td>
<td>9-63</td>
</tr>
<tr>
<td>RDCH</td>
<td>9-137</td>
</tr>
<tr>
<td>LEN Command</td>
<td>2-10, 3-60</td>
</tr>
<tr>
<td>Level Meter Command, AF</td>
<td>6-123</td>
</tr>
<tr>
<td>List, HOST Specific TMAC Quick Reference</td>
<td>5-1</td>
</tr>
<tr>
<td>Listener Only, Test Set Operating as</td>
<td>2-26</td>
</tr>
<tr>
<td>LJPRINT Command</td>
<td>2-21, 3-61</td>
</tr>
<tr>
<td>*LMC? Query</td>
<td>2-11, 2-34, 3-16</td>
</tr>
<tr>
<td>LN Command</td>
<td>2-8, 3-61</td>
</tr>
<tr>
<td>Loading TMAC Programs, Creating and</td>
<td>4-1</td>
</tr>
<tr>
<td>Lock Command (HOST Only), Keyboard</td>
<td>2-24</td>
</tr>
<tr>
<td>LOG Command</td>
<td>2-8, 3-62</td>
</tr>
</tbody>
</table>

### M

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro</td>
<td></td>
</tr>
<tr>
<td>Decision Points</td>
<td>2-12</td>
</tr>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>Identification</td>
<td>4-6</td>
</tr>
<tr>
<td>Relationship, Flash File and</td>
<td>4-6</td>
</tr>
<tr>
<td>Macros</td>
<td>2-10</td>
</tr>
<tr>
<td>Example TMAC Program with Multiple</td>
<td>4-8</td>
</tr>
<tr>
<td>Multitasking</td>
<td>2-15</td>
</tr>
<tr>
<td>Predefined</td>
<td>A-1</td>
</tr>
<tr>
<td>Variables and Arrays in</td>
<td>2-14</td>
</tr>
<tr>
<td>Mass Memory</td>
<td>See Flash Memory</td>
</tr>
<tr>
<td>Mathematical Functions</td>
<td>2-8</td>
</tr>
<tr>
<td>Mathematical Operators</td>
<td>2-7</td>
</tr>
<tr>
<td>Measure Commands, Generic</td>
<td></td>
</tr>
<tr>
<td>Memory Commands, Flash</td>
<td>6-165, 9-392</td>
</tr>
<tr>
<td>Memory, Flash Messages, Flash</td>
<td>6-169, 9-393</td>
</tr>
</tbody>
</table>
### M (cont)

<table>
<thead>
<tr>
<th>Message Builder, RACH Layer 3</th>
<th>9-349</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagram</td>
<td>9-344</td>
</tr>
<tr>
<td>RDCCH</td>
<td>9-342</td>
</tr>
<tr>
<td>User Data</td>
<td>9-343</td>
</tr>
<tr>
<td>Generator, RACH</td>
<td>9-220</td>
</tr>
<tr>
<td>Superframe Data</td>
<td>6-92</td>
</tr>
<tr>
<td>Meter Commands (HOST Only)</td>
<td></td>
</tr>
<tr>
<td>AF</td>
<td>6-123</td>
</tr>
<tr>
<td>AF Level</td>
<td>6-113</td>
</tr>
<tr>
<td>BER</td>
<td>6-101</td>
</tr>
<tr>
<td>Deviation</td>
<td>6-120</td>
</tr>
<tr>
<td>Peak</td>
<td>6-115</td>
</tr>
<tr>
<td>RMS</td>
<td>6-106</td>
</tr>
<tr>
<td>Digital Multimeter</td>
<td>6-96</td>
</tr>
<tr>
<td>Distortion</td>
<td>6-104</td>
</tr>
<tr>
<td>Frequency Error</td>
<td>6-98</td>
</tr>
<tr>
<td>Modulation</td>
<td>6-98</td>
</tr>
<tr>
<td>Power</td>
<td></td>
</tr>
<tr>
<td>Phase</td>
<td>6-118</td>
</tr>
<tr>
<td>Phase (RMS)</td>
<td>6-122</td>
</tr>
<tr>
<td>Signal Strength</td>
<td>6-112</td>
</tr>
<tr>
<td>SINAD</td>
<td>6-109</td>
</tr>
<tr>
<td>Miscellaneous Commands</td>
<td>9-394</td>
</tr>
<tr>
<td>Mobile Station</td>
<td></td>
</tr>
<tr>
<td>Control Messages</td>
<td>9-212</td>
</tr>
<tr>
<td>MSID</td>
<td>9-139</td>
</tr>
<tr>
<td>Simulation Commands (DCCH)</td>
<td>9-334</td>
</tr>
<tr>
<td>Setup</td>
<td>9-338</td>
</tr>
<tr>
<td>Modulation Accuracy (MODACC)</td>
<td></td>
</tr>
<tr>
<td>Commands (FDTG)</td>
<td>9-391</td>
</tr>
<tr>
<td>Program Example</td>
<td>10-22</td>
</tr>
<tr>
<td>Modulation Meter Commands</td>
<td>6-104</td>
</tr>
<tr>
<td>Monitor</td>
<td></td>
</tr>
<tr>
<td>AMPS/NAMPS</td>
<td>6-125</td>
</tr>
<tr>
<td>FDCH Data</td>
<td>9-59</td>
</tr>
<tr>
<td>FDCH Real Time Data</td>
<td>9-71</td>
</tr>
<tr>
<td>Layer 2 Data</td>
<td></td>
</tr>
<tr>
<td>FDCH</td>
<td>9-63</td>
</tr>
<tr>
<td>RDCCH</td>
<td>9-137</td>
</tr>
<tr>
<td>RDCCH Data</td>
<td>9-133</td>
</tr>
<tr>
<td>RDCCH Real Time Data</td>
<td>9-140</td>
</tr>
<tr>
<td>RDCD</td>
<td>9-47</td>
</tr>
<tr>
<td>RECC</td>
<td>9-41</td>
</tr>
<tr>
<td>RVC</td>
<td>9-45</td>
</tr>
<tr>
<td>MSID, Mobile Station</td>
<td>9-139</td>
</tr>
<tr>
<td>Multimeter Commands, Digital</td>
<td>6-115</td>
</tr>
<tr>
<td>Multitasking Macros</td>
<td>2-15</td>
</tr>
</tbody>
</table>

### O

<table>
<thead>
<tr>
<th>OFF Command</th>
<th>2-8, 3-63</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMT Examples</td>
<td>10-12</td>
</tr>
<tr>
<td>ON Command</td>
<td>2-8, 3-63</td>
</tr>
<tr>
<td>*OPC Command</td>
<td>2-25, 2-31, 2-34, 3-16</td>
</tr>
<tr>
<td>*OPC? Query</td>
<td>2-34, 3-16</td>
</tr>
<tr>
<td>Operating</td>
<td>2-26</td>
</tr>
<tr>
<td>as Listener Only (HOST Only)</td>
<td>2-26</td>
</tr>
<tr>
<td>as Talker Only (HOST Only)</td>
<td></td>
</tr>
<tr>
<td>Via GPIB</td>
<td>2-25</td>
</tr>
<tr>
<td>HOST Only</td>
<td>2-24</td>
</tr>
<tr>
<td>Sp Tst Only</td>
<td>4-1</td>
</tr>
<tr>
<td>Via RS-232</td>
<td></td>
</tr>
<tr>
<td>Operation Status Register</td>
<td>2-31</td>
</tr>
<tr>
<td>Operation Instrument Register</td>
<td>2-32</td>
</tr>
<tr>
<td>Operation, GPIB</td>
<td>2-24</td>
</tr>
<tr>
<td>Operators</td>
<td>2-7</td>
</tr>
<tr>
<td>Mathematical</td>
<td>2-7</td>
</tr>
<tr>
<td>*OPT? Query</td>
<td>2-34, 3-16</td>
</tr>
<tr>
<td>OR</td>
<td>2-13, 3-63</td>
</tr>
<tr>
<td>Operation Results</td>
<td>3-64</td>
</tr>
<tr>
<td>OR Operation Results, Bitwise</td>
<td>3-7</td>
</tr>
<tr>
<td>Order of Calculation</td>
<td>2-8</td>
</tr>
<tr>
<td>Oscilloscope</td>
<td>6-66</td>
</tr>
<tr>
<td>Remote Examples</td>
<td>6-76</td>
</tr>
<tr>
<td>OTHERWISE ENCASE, CASE OF Command</td>
<td>3-24</td>
</tr>
<tr>
<td>Overhead</td>
<td></td>
</tr>
<tr>
<td>Enable Commands (FOCC)</td>
<td>9-218</td>
</tr>
<tr>
<td>Message Parameters, FOCC</td>
<td>9-157</td>
</tr>
<tr>
<td>Messages, Global Action</td>
<td>9-206</td>
</tr>
<tr>
<td>Overview</td>
<td></td>
</tr>
<tr>
<td>of TMAC</td>
<td>1-1, 2-1</td>
</tr>
<tr>
<td>TDMA Transmission</td>
<td>9-334</td>
</tr>
</tbody>
</table>

### P

| Page, Video (HOST Only)        | 2-22 |
| Control                        | 3-99 |
| Settings                       |       |
| PAINT Command                  | 3-64 |
| Parameter, Speech              | 9-48 |
| Parameters for Sp Tst Editing Commands | 9-395 |
| Phase Meter Commands           | 6-118 |
| RMS                            | 6-122 |
| PIXEL Command                  | 2-21, 3-65 |
| PIXEL? Command                 | 2-10, 3-65 |
| Plot Commands (HOST Only), System | 2-23 |
| *PMC Command                   | 2-11, 2-34, 3-17 |
| Points, Macro Decision         | 2-12 |
| Power Meter Commands           | 6-98 |
| PPCTRL Command                 | 2-33, 3-67 |
| Predefined Macros and Constants| A-1   |
| Preset Conditions, Enable Register | 3-77  |
| Primary and Secondary Cycles, Examples of | 9-159 |
| PRINT Command                  | 2-21, 3-68 |
| Print Control, GPIB (HOST Only) | 2-27  |
| Process and Handoff (FOCC)     | 9-168 |
| Program                        |       |
| Creating a TMAC                | 4-3   |
| Example TMAC                   | 4-5   |
| with Multiple Macros           | 4-8   |
| Executing TMAC                 | 4-4   |
| Uploading TMAC                 | 4-3   |
| Setup, RS-232 Terminal Emulation| 4-2   |

---

Index-15
P (cont)

Program Commands
Program Examples
Bit Error Rate (BER)
Cell Site Simulation (CSS)
Base Station Initiated Call
Global Action Overhead Message
Handoff
Mobile Station Control
Mobile Station Initiated Call
Overhead Message for FOCC
Overhead Message Using Primary and Secondary Cycles
Page
Digital Control Channel (DCCH) Cell Site Simulation
Forward Control Channel (FOCC)
Monitoring Decoded Data
Monitoring Raw Data
Forward Digital Traffic Channel (FDTC)
Monitoring Decoded Data
Monitoring Raw Data
Forward Voice Channel (FVC)
Monitoring Decoded Data
Monitoring Raw Data
GPB
Modulation Accuracy (MODACC)
Reverse Control Channel (RECC)
Reverse Digital Traffic Channel (RDTC)
Reverse Voice Channel (RVC)
PScreen Command

Q
Questionable Status Register
Quick Reference List
HOST Specific
Special Test (Sp Tst)

R
RACH Layer 3 Message Builder
RACH Message Transmission
Contiguous
Sub Channel
RAND Command
Random Access Control Channel (RACH) Message Generator
Ranges and Units, DMM
Raw Data
Buffered
FDCCH
RDCCCH
Continuous Remote
FDCCH
RDCCCH
IS-54
Monitor Commands
FDT
FOCC
FVC
FDCCH
RDCCCH
Raw Generator, RDCCCH
Raw Timeslot Data, Continuous Remote
FDCCH
RDCCCH
• RCL Command

R (cont)

RCDCCH
Data Monitor
Message Diagram
Normal Length Message in the
Raw Generator
Real Time Data Monitor
Real Time Data Monitor
FDCCH
RDCCCH
Receiver
Remote Examples
Register (HOST Only)
Enable, Preset Conditions
Instrument Status
Instrument Summary Status
Operation Instrument
Operation Status
Questionable Status
Standard Event Status
Status
Configuration
Reference List
HOST Specific Quick
Sp Tst Quick
Relationships, Flash File and Macro Name
Remote
AF Generator Examples
AMPS/NAMPS Cell Site Monitor Example
Duplex Examples
Oscilloscope Examples
Receiver Examples
RF Generator Examples
Spectrum Analyzer Example
RETURN Command
Reverse Control Channel (RECC)
Monitor
Decode Data
Program Example
Reverse Digital Traffic Channel (RDTC)
Monitor
Decode Data
Program Example
Simulation
Reverse Voice Channel (RVC)
Monitor
Decode Data
Program Example
Simulation
RF Generator
Examples, Remote
RJPRINT Command
RND Command
ROOM Command
ROTATE Command
Rotation for n Values, Character
RQS Bit and SRQ Generation, HOST
RS-232
Configure Commands
Operating IPR-1900 CSA Via
Terminal
Emulation Program Setup
Executing TMAC Program from
• RST Command

Index-16
S

*SAV Command 2-34, 3-17
Saving Variables and Arrays (HOST Only) 2-4
Scope 6-66
Remote Examples 6-78
Screen See Color Display
Characters D-1
Commands 6-2
SCREEN:USER Command 2-19, 3-74
Secondary Cycles, Examples of Primary and 9-159
Selection Numbers, Colors and Color 2-18
Self Test Commands (HOST Only), Individual C-1
Settings, Video Page 3-99
Setup
Commands 9-338
Screen Mobile Station Simulation 9-338
FDCCH Data Monitor 9-59
HOST (RS-232) 6-1
RDCC Data Monitor 9-133
Creating and Loading TMAC Programs 4-1
ISRI 3-13
Superframe 9-218
Shared Channel Feedback Response 9-223
SIGN Command 2-8, 3-74
Signal Strength Meter Commands 6-112
Simulation Commands
AMPS/NAMPS 6-132
DCCH Mobile Station 9-334
E-BCCCH 9-248
F-BCCCH 9-224
FDCC 9-178
FOCC 9-157
FVC 9-170
RDCCH 9-334
RDT 9-387
RVC 9-388
SPACH 9-291
SIN (Sine) Command 2-8, 3-75
SINAD Meter Commands 6-109
SLEEP Command 2-15, 3-75
Soft Function Key 2-20
Displays 2-20
Example 2-20
SOUND Command 2-22, 3-75
Sound Control, Display and 2-13
SOUNDS See Audio Tones
SPACH Simulation Commands 9-387
Special Editing Commands, TMAC 9-395
Special Test (Sp Tst) 9-395
Quick Reference List 8-1
Specific TMAC Commands 9-1
Specific TMAC Commands
HOST 6-1
Sp Tst 6-1
Spectrum Analyzer 6-79
Remote Example 6-91
Speech Parameter 9-48
SQR Command 2-8, 3-76
*SRQ Command and query 2-25, 2-30, 2-34, 3-18
SRQ Generation, 2-30
HOST RTS Bit and 2-31
Standard Event Status Register
S (cont)
Status (HOST Only)
Byte 2-30
Registers 2-28
Configuration 2-29
Instrument 2-32
Instrument Summary 2-33
Questionable 2-32
Standard Event 2-31
Subsystem 2-28
STATUS:OPERation:INSTRument
Queries and command 2-32, 3-76
STATUS:OPERation Queries and command 2-31, 3-77
STATUS:PRESet Command 2-30, 3-77
STATUS:QUESTIONable:INSTRument:
SUMmary Queries and command 2-33, 3-78
STATUS:QUESTIONable:INSTRument
Queries and command 2-32, 3-33, 3-79
STATUS:QUESTIONable Queries and command 2-32, 3-33, 3-80
*STOP Command 2-30, 3-23, 3-84, 3-18
STOP Command 3-80
Storing Loaded TMAC Program into Flash Memory 4-6
STR Command 2-10, 3-81
STRING Command 2-9, 3-81
String Variables and Functions 2-9
STRPOS Command 2-10, 3-82
SUB Channel RACH Message 9-337, 9-346
Summary Status Register, Instrument 2-33
Superframe 9-220
Setup 9-218
Syntax 2-1
System
Commands 2-23
Error 2-23
Key 2-23
Faults (HOST Only) 2-23
Keyboard Lock Command (HOST Only) 2-24
Plot Commands (HOST Only) 2-23
RS-232 Configure Commands 2-24
Time and Date Commands 2-23
SYSTEM:COMM unicate:GPIB:
ADDRESS Command 2-25, 2-26, 3-82
SYSTEM:COMM unicate:GPIB:CMD
Command 2-26, 3-83
SYSTEM:COMM unicate:GPIB:CONT rol
Command 2-26, 3-83
SYSTEM:COMM unicate:GPIB:DCl
Command 2-26, 3-83
SYSTEM:COMM unicate:GPIB:GEl
Command 2-26, 3-83
SYSTEM:COMM unicate:GPIB:LONy
Command 2-26, 3-84
SYSTEM:COMM unicate:GPIB:PRINTER
Command 2-27, 3-84
SYSTEM:COMM unicate:GPIB:SLAVE
Command 2-26, 3-84
SYSTEM:COMM unicate:GPIB:SPOLL?
Query 2-26, 3-85
SYSTEM:COMM unicate:GPIB:SRQ?
Query 2-26, 3-86
SYSTEM:COMM unicate:GPIB:TONy
Command 2-26, 3-86
SYSTEM:COMM unicate:SERial
Commands 2-24, 3-87
Index-17
S (cont)

SYSTem:CUR:sor:DEFaults Command 2-23, 3-88
SYSTem:DATE Command and query 2-23, 3-88
SYSTem:DEFaults Command 2-23, 3-88
SYSTem:DISPLAY:DEFaults Command 2-23, 3-89
SYSTem:ERROR? Query 3-89
SYSTem:FREQ:LIST Command and query 2-27, 3-89
SYSTem:FREQ:LOCK Command and query 2-27, 3-90
SYSTem:KEY Command and query 2-27, 3-90
SYSTem:KEY:DEFine Command 2-23, 3-91
SYSTem:KEY:DELete Command 2-23, 3-91
SYSTem:MILLisecond? Query 2-23, 3-92
SYSTem:PLOT Commands 2-23, 3-92
SYSTem:PRTHrough:GPiB Command and query 2-26, 3-92
SYSTem:PRTHrough:SERial Command and query 2-24, 3-93
SYSTem:PRTHrough:SERial:QUEue? Query 2-24, 3-93
SYSTem:TIME Command and query 2-23, 3-93

T

TAB Command 2-10, 3-94
Taikback 9-206
Talker Only, Test Set Operating as 2-26
TASK Command 2-15, 3-94
TMKA Transmission Overview 9-334
Terminal
Emulation Program Setup, RS-232 4-2
Executing TMAC Program from RS-232 4-7
Test Commands (HOST Only), Individual Self 2-27
Test Set
Frequency Control (HOST Only) 2-27
Operating as Talker Only 2-26
Text, Graphics and 2-21
TICKS? Query 2-23, 3-94
Time and Date Commands, System 2-23
TMAC (General) Commands 3-1
TMAC Commands
HOST Specific 6-1
Special Test (Sp Tst) Specific 9-1
TMAC Overview 1-1, 2-1
TMAC Program
Creating 4-3
Example 4-5
with Multiple Macros 4-8
Executing 4-4
Uploading 4-4
TMAC Program into Flash Memory, Storing 4-6
Uploaded 4-6
TMAC Special Editing Commands 9-395
Tones, Audio 2-22
TPAUSE Command 2-15, 3-95
Transmission Overview, TDMA 9-334
Transmitter Commands, Duplex 6-41
*TRG Command 2-34, 3-18
TRUE Command 2-8, 3-95
TSTOP Command 2-15, 3-95
*TST? Query 2-34, 3-18

U

Units, DMM Ranges and 3-116
Uploading TMAC Program 4-4
USER Command 2-19, 3-96
User Data Message Diagram 9-342
User-Defined Optional Message Types and 9-283
Information Elements for the F-BCH and
E-BCH

V

VAL Command 2-10, 3-96
VAR Command 2-3, 3-97
Variables and Arrays 2-14
in Macros 2-3
Numeric 2-4
Saving (HOST Only) 2-9
Variables and Functions, String 2-9
Video Page (HOST Only) 2-22
Control 3-99
Settings 3-99
VIDEOpage:COPY Command 2-22, 3-98
VIDEOpage:SET Command 2-22, 3-99

W

*WAI Command 2-11, 3-19
WAKE Command 2-15, 3-99
WCLOSE Command 2-19, 3-100
WEND, WHILE Command 2-15, 3-100
WHITE WEND Command 2-15, 3-100
WINDOW? Query 2-19, 3-101
Window Example 2-19
Windows 2-19
WMOVE Command 2-19, 3-101
WOPEN Command 2-19, 3-102
WSEL Command 2-19, 3-103

X

XOR Operation Results, Bitwise 3-6
XY Command 2-22, 3-103
XYPRINT Command 2-21, 3-104

Index-18