

Please make the following corrections to your MultiI-0 manual:

1. Remove pages 135-136 and insert the new pages 135, 135A, and 136.

2. Add the following note to the bottom of page 155:

NOTE: The status byte received from a device may not be dependent on the secondary address used, if any. The status byte itself should indicate which function in the device requires service. Thus, the device may require service on a secondary address different from the one returned by the C\_SERPOL routine.

3. On page 58, second paragraph, fourth line from the end, change EOI.FLAG to LF.FLAG.

4. On page 98, left column, second line of equations at the top, first term, add a negation bar over ATN.

5. On page 100, right column, top state diagram, lower arc from SPAS state to TADS state, remove the negation bar over ATN.

EXAMPLE: GNOBRK:GPRINT#4,"DATA"

will cause the normal addressing and unaddressing overhead to be suspended for IEEE channel 4.

NOTES: 1. The GNOBRK command itself will not cause the associated device to be addressed. On the next GGET#, GINPUT#, or GPRINT# command the device will be addressed, and remain addressed until normal operation is restored.

#### 4.2.1.8 The GNOLF Command

PURPOSE: To specify if line feeds should be expected or sent after carriage returns when using the GINPUT# or GPRINT# commands, respectively.

SYNTAX: GNOLF# <ieee chan>

ARGUMENTS: <ieee chan> = a numeric expression which evaluates to a number from 0 to 9. This number will be used as the IEEE channel to be configured.

DISCUSSION: The GNOLF# command is used to specify whether line feeds should be expected or sent following carriage returns for a particular IEEE channel. This applies when inputting data with the GINPUT# command, or when outputting data with the GINPUT# command. The default is for line feeds to always follow carriage returns. In some situations it may be required that line feeds not follow carriage returns. In this case a GNOLF# command should be executed after the GASSIGN for that channel. It will also be necessary to use the GNOLF# command if you wish to control the sending of carriage returns and line feeds explicitly.

In the case of the GINPUT# command, expecting a line feed implies that one additional character will be read following the receipt of a carriage return. If this character turns out to be a line feed, then the GINPUT# command will discard the line feed and continue normally. If the character isn't a line feed, an error message will be issued. If a GNOLF# is executed, then inputting of a data item will stop on the carriage return. If a line feed follows this carriage return, the line feed will be taken as the first character of the next data item. This line feed will likely cause the inputted data to be processed incorrectly.

EXAMPLE: GNOLF# 4

will specify that line feeds are not expected or sent following carriage returns on IEEE channel 4.

NOTES: 1. The GNOLF# command must be executed following the GASSIGN command since the GASSIGN command always initializes the channel for adding or expecting line feeds after carriage returns.

#### 4.2.1.10 The GPRINT# Command

PURPOSE: To output data to an IEEE channel.

SYNTAX: GPRINT# <ieee chan> [ , <expression> [ ; [<expression>] ] ... ]

ARGUMENTS: <ieee chan> = a numeric expression which evaluates to a number from 0 to 9. This number will be used as the IEEE channel to which the output will be sent.

<expression> = an expression which evaluates to the data to be output.

DISCUSSION: The GPRINT# command is used to output data to an IEEE channel. Any numeric data specified will be converted to a string before being sent over the channel. After all the data has been sent, a terminating carriage return will be appended unless the GPRINT# command ends with a semicolon.

Normally a line feed will be sent immediately following each carriage return that occurs in the data sent by the GPRINT# command. This includes carriage returns which are explicitly embedded in the arguments (via CHR\$(13)). These automatic line feeds may be inhibited by executing a GNOLF# command on the IEEE channel. Also, the EOI signal will be asserted during the transfer of the last byte for the command, unless a GNOBRK command is in effect for the channel.

EXAMPLE: GPRINT# 5,D1\$;XN;L\$

will send the contents of the three data items to the device assigned to IEEE channel 5. The contents of D1\$ will be immediately followed by the value in XN converted to a string, which is in turn followed by the contents of L\$. A carriage return will be appended after the last character of L\$. A line feed will follow the carriage return unless a GNOLF#5 command has been executed since opening IEEE channel.

- NOTES: 1. If a GNOBRK command is in effect, the GPRINT# command will address the assigned device, if not already addressed. In addition, the GPRINT# will not unaddress the device after the data is sent.
2. The GPRINT# command returns the ST variable set appropriately. The value will equal the number of arguments output, minus 128 if EOI was sent. If the assigned device does not accept the data, 64 will be added to the current value of the ST variable and the GPRINT# command aborted.

#### 4.1.1.11 The GCMD Command

PURPOSE: To send an interface message on the IEEE-488 bus.

SYNTAX: GCMD <byte> [ , <byte> ] ...

ARGUMENTS: <byte> = a numeric expression which evaluates to a number from 0 to 255.

DISCUSSION: The GCMD command is used to send interface messages (i.e. messages sent with the ATN line asserted) on the IEEE-488 bus. This allows BASIC programs to send commands such as group execute trigger (GET) or parallel poll configure (PPC), etc.

4.2.1.9 The GOFF Command

PURPOSE: To have the MTU-130 release control of the IEEE-488 bus.

SYNTAX: GOFF

ARGUMENTS: none

DISCUSSION: The GOFF command is used make the MTU-130 release control of the IEEE-488 bus. This command must be executed if you wish to allow some other device or computer to act as the IEEE-488 bus controller. As long as the MTU-130 thinks it is the controller, it will continue to drive certain signals on the bus. Executing the GOFF command will essentially disconnect the MTU-130 from the IEEE-488 bus, thus allowing some other device to be the controller. It is recommended that this command be executed prior to exiting BASIC if a GINIT command has been previously executed.

EXAMPLE: GOFF

This statement will cause the MTU-130 to release control of the IEEE-488 bus.

NOTES: 1. The MTU-130 will also release control of the IEEE-488 bus if a cold or warm reset of the 130 is performed.

2. The IEEEEL library does not have the ability to pass control directly from the MTU-130 to another device capable of assuming control. You must execute a GOFF command first, then manually start the other controller.