QTERM-II USER'S MANUAL

REVISION 14

BEIJER ELECTRONICS
2212 South West Temple #50
Salt Lake City, Utah 84115-2648
USA
Phone 801-466-8770
Fax 801-466-8792
Email info@beijerelectronicsinc.com
Web http://www.beijerelectronicsinc.com

93920E1 - Printed in USA

Manual Updated 29 March 2011

FCC COMPLIANCE STATEMENT

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Any modification to this device (including any changes to the recommended antenna configuration) that are not expressly approved by Beijer Electronics could void the user's authority to operate this device.

FOREWORD

The Beijer Electronics QTERM-II is a data-entry terminal for industrial applications. The QTERM-II is available in many different configurations; this manual discusses all versions and their operation.

This manual is written for Version 3.0 of the QTERM-II operation code. Be sure to read the file READ.ME on your distribution disk for any changes due to newer versions of code.

The QTERM-II is a CE certified product. It has been assessed against the requirements of EN 50082-1: 1992, EN 55022: 1987 and EN 60950 (including Amendments Nos. 1, 2, and 3). Based on conformity with these requirements, the QTERM-II is deemed in compliance with all applicable CE directives.

The sections of this manual are as follows:

Chapter 1 **Quick Start.** If you wish to start using your QTERM-II immediately, this chapter will show you how to connect to the terminal and how to run the example software.

Chapter 2 **QTERM-II Software.** This chapter provides a detailed listing of the QTERM-II commands and discusses how to use them.

Chapter 3 **Using The QDATA File.** The QDATA file is used to configure the QTERM-II. This chapter shows you how to modify the file and load it into your QTERM-II.

Chapter 4 **QTERM-II Hardware.** This chapter discusses the hardware of the QTERM-II, including dimensional drawings, interface specifications, connector pin assignments, performance and environmental specifications.

Appendix A **ASCII Chart.** This is a true 7-bit ASCII chart, along with mnemonic definitions.

Appendix B **QTERM-II Character Chart.** This is a 256-character chart showing how the QTERM-II handles every character it receives. The lower half is similar to, but not the same as, the true ASCII chart in Appendix A.

Appendix C **QTERM-II Command Summary.** This is an abbreviated summary of QTERM-II software commands.

Appendix D **Customizing the QTERM-II.** There are many ways the QTERM-II can be customized; this appendix discusses many of the options.

Quick Start 1
Power-On Setup 1
Connect the Communications Lines 1
Apply Power 2
Communicate 2
Running QDEMO 2
Using the QTERM-II 3
QTERM-II Software 5
Operation 5
Handshaking 5
Commands vs QDATA Parameters 5
Character, Key and LED Numbering 5
Software Commands 6
Bell (^G) - 07h 6
Backspace (^H) - 08h 6
Horizontal Tab (^I) - 09h 6
Line Feed (^J) - 0Ah 6
Vertical Tab (^K) - 0Bh 6
Form Feed (^L) - 0Ch 6
Carriage Return (^M) - 0Dh 7
XON (^Q) - 11h 7
XOFF (^S) - 13h 7
Delete - 7FH 7
Cursor Up - ESC A 7
Cursor Down - ESC B 7
Cursor Right - ESC C 7
Cursor Left - ESC D 8
Clear Screen - ESC E 8
Set Timeout Delay - ESC F # 8
Set Tab Spacing - ESC G # 8
Cursor Home - ESC H 8
Set Cursor Position - ESC I # * 8
Erase to End of Screen - ESC J 8
Erase to End of Line - ESC K 8
Set Contrast - ESC L # 8
Reset QTERM-II - ESC M 8
Query Version - ESC N 8
Buzzer On/Off/Beep - ESC O # 9
LEDs On/Off/Blink/Toggle - ESC P # 9
Insert/Replace Mode - ESC Q # 9
Auto Wrap Mode - ESC R # 9
Auto Scroll Mode - ESC S # 9
Auto Line Feed Mode - ESC T # 9
Display Off - ESC U 9
Backlight On/Off/Toggle - ESC V # 10
Query Status - ESC W 10
Query Cursor Position - ESC X 10
Query Character - ESC Y 10
Scroll Down - ESC Z 10
Key Click/Repeat Mode - ESC a # 10
Set Cursor Mode - ESC b # 10
Set Shift Mode - ESC c # 10
Control dig0 and dig1 - ESC d # 10
Set Buzzer Duration - ESC e # 11
Set Key Click Duration - ESC f # 11
Set Key Repeat Rate - ESC g # 11
Set LED Blink Rate - ESC h # 11
Save Parameters to EEPROM - ESC i 11
Bar Code Control - ESC j # 11
Transmit Buffer Flush - ESC k 12
XON/XOFF Mode - ESC l # 12
User Area Read/Write - ESC m # 12
Get Free User Area - ESC n 12
Using The QDATA File 17
QDATA Files 17
Using QSETUP 17
Loading a QDATA File 18
Loading a QCODE File 18
Retrieving a QDATA File 18
Basic QDATA File Concepts 18
Parameter and Key String Sections 19
Parameter Section 19
Key String Section 19
Custom Display Characters 20
QTERM-II Hardware 23
Handheld QTERM-II 23
Battery Back QTERM-II 23
Panel-Mount QTERM-II 24
Old Panel-Mount QTERM-II 24
New Panel-Mount QTERM-II 25
Interfaces 26
EIA-232 Interface 26
EIA-422 Interface 26
5-volt Buffered Interface 26
Other QTERM-II Hardware 26
Display 26
Keypad 26
Digital Outputs, dig0 and dig1 27
Buzzer 27
Regulator Option 27
Bar Code Option 27
The EEPROM Disable Jumper 28
Adding Another PC Board 28
QTERM-II Specifications 28

ASCII Chart 31
31
QTERM-II Character Chart 33
QTERM-II Command Summary 35
Customizing the QTERM-II 39
Standard Customization 39
Custom Customization 39

CHAPTER 1.

QUICK START

There are only four steps required to communicate with the QTERM-II:

- use *Power-On Setup* to set the contrast, baud rate and data format
- connect to your host transmit, receive and ground lines
- apply power
- transmit and receive with the QTERM-II

1.1 Power-On Setup

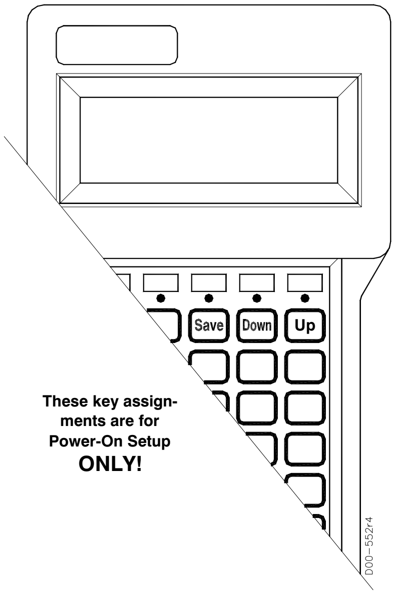
The *Power-On Setup* procedure is used to configure the QTERM-II's display contrast, baud rate and data format. You use three different QTERM-II keys to do this configuration (see Figure 1-1):

Top - Right:this is the *up* key
Top - 2nd from Right:this is the *down* key
Top - 3rd from Right:this is the *save* key

To perform the *Power-On Setup* follow these steps:

- Hold down any key and apply power to the QTERM-II (you do not need to connect the transmit and receive lines).
- The version of software in the QTERM-II will be displayed as long as you hold down the key, after which the you can adjust the display contrast.
- Set the desired display contrast using the *up* and *down* keys. When the display is at a contrast you like, press the *save* key.
- Set the desired baud rate (see Table 1-1) using the *up* and *down* keys. When the desired baud rate is displayed, press the *save* key.
- Set the desired data format (see Table 1-1) using the *up* and *down* keys. When the desired data format is displayed, press the *save* key.

Disconnect power and go to the next step.



These key assignments are for Power-On Setup **ONLY!**

Figure 1-1. Power-On Setup Key Assignments.

1.2 Connect the Communications Lines

The EIA-232 and 5-volt Buffered have one transmit and one receive line, while the EIA-422 has two transmit and two receive lines.

Table 1-2 shows the connector pin assignments for both the handheld and panel-mount QTERM-IIs. The receive and transmit directions shown in the table are with respect to the QTERM-II. Refer to this table to connect your host communications lines to the correct pins. Do not connect to the *dig0* and *dig1* pins; Chapter 2 discusses these pins and their uses.

Table 1-1: Available Baud Rates and Data Formats.

<u>Available Baud Rates</u>
9600, 4800, 2400, 1200, 600, 300, 150
<u>Available Data Formats</u>*
8n1, 8e1, 8o1, 8n2, 8e2, 8o2, 7n1, 7e1, 7o1, 7n2, 7e2, 7o2
*format is: data-parity stop

(If you are using an IBM-style PC, you cannot connect an EIA-422 or 5-volt Buffered QTERM-II directly to the computer's COM port; you must provide an interface device.)

1.3 Apply Power

Table 1-2 shows the pin assignments for the power and ground lines. Connect your DC power supply to the appropriate two pins.

WARNING: Power supplied to the QTERM-II must be from an SELV power source, and should have a current limit on its output of 5 Amperes. If you did not order the regulator option, the supply to the QTERM-II must provide a minimum of 4.8 volts DC and be limited to a maximum of 5.5 volts DC. If you did order the regulator option, the supply to the QTERM-II must provide a minimum of 5.5 volts DC and be limited to a maximum of 24 volts DC. Limiting may be inherent to the supply, or may be provided by supplementary overcurrent devices.

If you ordered the battery back option, you do not need to connect a power supply; just install six alkaline AA batteries in the battery compartment.

1.4 Communicate

At this point, characters which are transmitted by the host will be displayed on the QTERM-II display. If you press keys on the keypad, the QTERM-II will transmit the appropriate codes to the host computer.

1.5 Running QDEMO

To easily use some of the advanced features of the QTERM-II, run the QDEMO program that is included with this User's Manual:

- Connect your QTERM-II to either the COM1 or COM2 serial port on an IBM-style PC.

  (If you have an EIA-422 or 5-volt Buffered QTERM-II, you will need to provide an interface unit such as Beijer Electronics' QCOM-2; *you cannot directly connect a QTERM-II with one of these interfaces to the COM port on an IBM-style computer!)*

- Copy the file QDEMO.EXE to your hard disk.
- Apply power to the QTERM-II.
- Run the QDEMO program.
- Follow the on-screen instructions to use many of the features of the QTERM-II terminal.

QDEMO will allow you to exercise many of the features of the QTERM-II. It includes a terminal emulator so that you can also communicate directly with the terminal.

Table 1-2: QTERM-II Pin Assignments.

Handheld 6-pin Modular	Handheld DB9	Panel-Mount 2 mm	QTERM-II 232/5-volt Buffered Function	QTERM-II 422 Function
1	3	1	receive	+receive
2	6	2	dig1*	-receive
3	2	3	transmit	+transmit
4	1	4	dig0*	-transmit
5	9	5	+V^	+V^
6	5	6	ground	ground
Tx and *Rx* directions are with respect to the QTERM-II *these lines are optional; see section 4.5.3 ^no connection with battery back units				

1.6 Using the QTERM-II

There are three ways to "customize" the operation of the QTERM-II for your application:

- **Power-On Setup**

  To set the display contrast, baud rate and data format, see the beginning of this chapter.

- **Software Commands**

  If you want to send commands from your host computer to the QTERM-II, see Chapter 2 for a list of all commands and details on how to use them. Some of the commands allow you to control the display (such as cursor movement, clear screen and line feed), others control operation of the hardware (such as LED control and key click control), and others control external devices (bar code control and digital output control).

- **QDATA File Download**

  All configurable parameters of the QTERM-II can be controlled using software commands or by setting the proper values in the *QDATA file* and downloading it into the QTERM-II.

  The *only* way to modify the key strings (i.e. what the QTERM-II transmits when a key is pressed), or to modify the eight custom characters, is by modifying and downloading the QDATA file.

  Chapter 3 shows how to modify this file and how to load it into your QTERM-II. There are also several examples that utilize the powerful capabilities of key strings.

CHAPTER 2.

QTERM-II SOFTWARE

2.1 Operation

The operation of the QTERM-II is quite simple:

- Power is applied to the QTERM-II.
- Commands or data can be transmitted to the QTERM-II. Commands are executed as required; data is displayed.
- When a key is pressed, the QTERM-II transmits the appropriate character(s) to the host.

The QTERM-II has many additional capabilities which are accessed in three ways:

- Power-On Setup (see Chapter 1)
- Software commands from host (this chapter)
- Configuration-file download (see Chapter 3)

2.1.1 Handshaking

The QTERM-II has buffers for both receiving and transmitting characters. However, these buffers may not be large enough for some applications. If this is the case for your application, then your host must use XON/XOFF handshaking to make optimal use of the QTERM-II's capabilities.

The QTERM-II's transmit buffer (16 bytes long) is used when a host does not wish to receive characters for a period of time. The host sends an XOFF character to the QTERM-II. The QTERM-II starts placing keycodes (one character per keystroke, regardless of how many characters are in the string assigned to the key) into the transmit buffer rather than transmitting them. When the host is ready to receive characters, it sends an XON character, at which time the QTERM-II will start transmitting the characters in the buffer.

The QTERM-II's receive buffer is large enough to accept a full screen of text at the maximum baud rate (9600) without overrunning the buffer. However, some operations can take a longer time to execute, such as scrolling the entire screen down one line. In this case, the QTERM-II will send an XOFF to the host when it has only six bytes empty in its buffer. When it has processed input data to the point that the buffer only has six bytes used, it will send an XON, and the host can resume transmitting.

The command summary in Appendix C shows approximate execution time for each of the various operations of the QTERM-II.

XON/XOFF handshaking can also be disabled (by software command or the QDATA file) if you do not wish to use it. However, if you do not use handshaking, you must take extra care that you do not overrun the QTERM-II's receive buffer. The two easiest ways to do this are to use a slower baud rate (which gives the QTERM-II more time to process each byte) or add delays to your code after each write to the QTERM-II.

2.1.2 Commands vs QDATA Parameters

There are numerous parameters which are set by the QDATA configuration file (see Chapter 3) but which can be overridden by a software command. For example, tab spacing is set in the QDATA file, but can be changed by the **Set Tab Spacing** command.

In all such cases, the software command overrides the QDATA configuration file setting but does not store the new setting into EEPROM. Therefore, when power is turned off, the overriding value is lost, and when the terminal is turned on, the QDATA parameter will be used again.

If you wish to have the software command permanently override the QDATA configuration file setting, you can use the **Store Parameters to EEPROM** command. This saves all currently-set parameters to the EEPROM which makes them the current power-on defaults.

2.1.3 Character, Key and LED Numbering

Many of the commands refer to certain cursor locations, keys or LEDs. Figure 2-1 shows how these are numbered.

2.2 Software Commands

Once you have configured the QTERM-II using the power-on setup (see Chapter 1) and configuration-file download (see Chapter 3), you can program your host to control the QTERM-II using software commands.

Table 1-3 lists the commands available. Each command is discussed in detail below, and a command summary, including execution times, is given in Appendix C.

The notation "ESC" in this chapter always means the one-byte escape character (1Bh = 27 decimal) rather than the ASCII string of three letters.

2.2.1 Bell (^G) - 07h

This causes the buzzer to beep. The default beep duration (1/2 second) is set in the QDATA file; it can be overridden with the *Set Buzzer Duration* command (ESC e). Note that sending this command is identical to sending the *Buzzer On/Off/Beep* command for a beep (ESC O B).

2.2.2 Backspace (^H) - 08h

Causes a non-destructive backspace, i.e. characters are not erased as the cursor is backspaced over them. With auto wrap mode off, the backspace stops at the left edge of the current display line. With auto wrap mode on, the cursor will wrap to the last position on the previous line. The command is ignored if the cursor is at the home position. See also the *Delete* command (7Fh).

2.2.3 Horizontal Tab (^I) - 09h

Moves the cursor right to the next tab column. The default tab spacing is set in the QDATA file; it can be overridden with the *Set Tab Spacing* command (ESC G). (The QDATA default tab spacing is every 4 spaces.)

With auto wrap on, the cursor will wrap down to the first column in the line below when it is tabbed beyond the last column in the current line. If auto wrap is off, the cursor will stop at the end of the current line. If auto scroll and auto wrap are both on, then the display will scroll up as the cursor is tabbed beyond the last column in the last line.

2.2.4 Line Feed (^J) - 0Ah

Moves the cursor down one line without changing its horizontal position. When auto scroll mode is on and a line feed is performed on the last line, the display will scroll up with the horizontal cursor position unaltered.

2.2.5 Vertical Tab (^K) - 0Bh

Performs the same function as *Line Feed*.

2.2.6 Form Feed (^L) - 0Ch

Performs the same function as *Line Feed*.

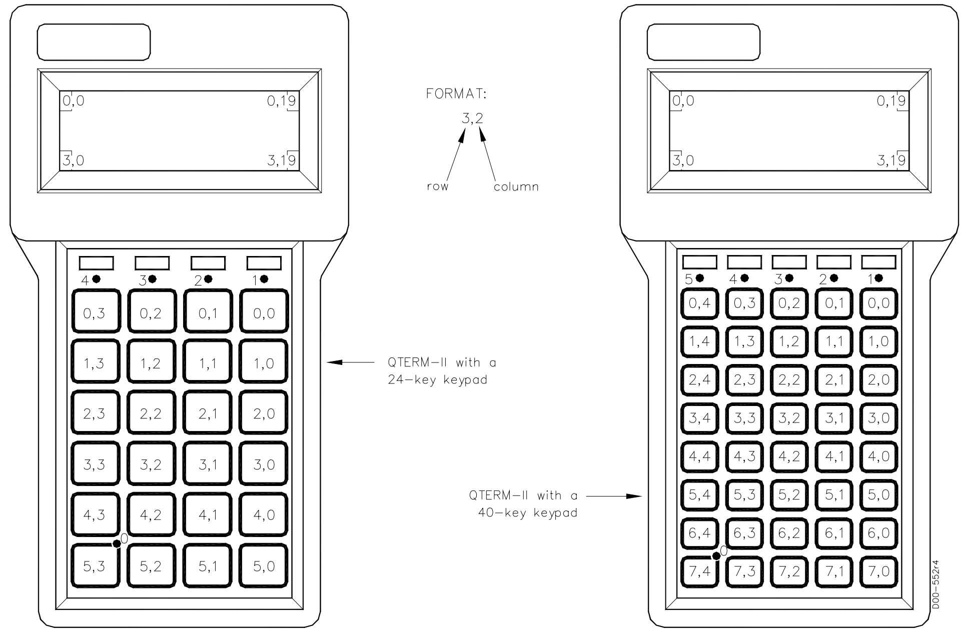


Figure 2-1. Numbering of Cursor Positions, LEDs and Keys.

Table 1-3: QTERM-II Software Commands.

Command	Code	Command	Code
Bell (^G)	07h	LEDs On/Off/Blink/Toggle	ESC P..
Backspace	08h	Insert/Replace Mode	ESC Q..
Horizontal Tab (^I)	09h	Auto Wrap Mode	ESC R..
Line Feed (^J)	0Ah	Auto Scroll Mode	ESC S..
Vertical Tab (^K)	0Bh	Auto Line Feed Mode	ESC T..
Form Feed (^L)	0Ch	Display Off	ESC U
Carriage Return (^M)	0Dh	Backlight On/Off/Toggle	ESC V..
XON (DC1 or ^Q)	11h	Query Status	ESC W
XOFF (DC3 or ^S)	13h	Query Cursor Position	ESC X
Delete	7Fh	Query Character	ESC Y
Cursor Up	ESC A	Scroll Down	ESC Z
Cursor Down	ESC B	Key Repeat/Click Mode	ESC a..
Cursor Right	ESC C	Set Cursor Mode	ESC b..
Cursor Left	ESC D	Set Shift Mode	ESC c..
Clear Screen	ESC E	Control *dig0* and *dig1*	ESC d..
Set Timeout Delay	ESC F..	Set Buzzer Duration	ESC e..
Set Tab Spacing	ESC G..	Set Key Click Duration	ESC f..
Cursor Home	ESC H	Set Key Repeat Rate	ESC g..
Set Cursor Position	ESC I..	Set LED Blink Rate	ESC h..
Erase to End of Screen	ESC J	Save Parameters to EEPROM	ESC i
Erase to End of Line	ESC K	Bar Code Control	ESC j..
Set Contrast	ESC L..	Transmit Buffer Flush	ESC k
Reset QTERM-II	ESC M	XON/XOFF Mode	ESC l..
Query Version	ESC N	User Area Read/Write	ESC m..
Buzzer On/Off/Beep	ESC O..	Get Free User Area	ESC n

2.2.7 Carriage Return (^M) - 0Dh

Moves the cursor to left edge of the display on the current line. If auto line feed mode is on, then the cursor moves to the left edge of the next line. If auto scroll and auto line feed are both on, a carriage return on the last line will cause the display to scroll up and the cursor to be positioned at the left edge of the last line.

2.2.8 XON (^Q) - 11h

Enables the QTERM-II to transmit keys pushed after receiving an XOFF. XON is used to re-enable QTERM-II transmission after an XOFF has disabled it, allowing handshaking with the host system.

2.2.9 XOFF (^S) - 13h

Disables all QTERM-II transmission except for information requested via the *Query Status* command (ESC W). After receiving an XOFF command, the QTERM-II stores characterstyped on the keypad in a transmit buffer. These characters will be transmitted when an XON is received. If the buffer becomes full before an XON is received, additional characters which are typed will be ignored.

2.2.10 Delete - 7FH

Delete works in the same way as *Backspace* (08h, ^H), except that characters are erased as the cursor moves over them.

2.2.11 Cursor Up - ESC A

Moves the cursor up one line without changing its horizontal position. It has no effect if the cursor is on the first line.

2.2.12 Cursor Down - ESC B

Moves the cursor down one line without changing its horizontal position. It has no effect if the cursor is on the last line.

2.2.13 Cursor Right - ESC C

Moves the cursor right one space without changing its vertical position. It has no effect if the cursor is at the rightmost position on the current line.

2.2.14 Cursor Left - ESC D

Moves the cursor left one space without changing its vertical position. It has no effect if the cursor is at the left-most position on the current line.

2.2.15 Clear Screen - ESC E

Clears the display and moves the display cursor to home (the left-most position in the top line of the display).

2.2.16 Set Timeout Delay - ESC F

This sets the time after which, if no activity has occurred, the QTERM-II will automatically turn off the display (in order to save power). Once the display has been turned off, it is turned back on by either of two events:

- If the operator presses a key, the display is turned back on. However, the key is *not* transmitted to the host.
- If the host sends a command or character to the QTERM-II, the display is turned back on, and the command or character will be processed normally.

If the QTERM-II has a backlight, and it is on when the timeout occurs, the backlight is turned off along with the display. When the display is turned back on, the backlight is turned on if it was on when the timeout occurred.

Anything displayed on the LCD screen when the timeout occurs is lost, i.e. when the display is turned back on, will be blank. Custom character definitions are *not* lost.

The format for this command is *ESC F #*, where # is a character in the range of 40h to 7Fh ('@' to DEL). 40h disables the timeout function. Other values set the timeout delay in 20 second increments, from 20 seconds through 21 minutes. See Table 1-4 (at the end of this chapter) for a complete list of codes and timeout delays.

The only thing the host can do that will not turn on the display is to issue the *Query Status* command (ESC W).

2.2.17 Set Tab Spacing - ESC G

This sets the width of the horizontal tab columns. The command has the form *ESC G #*, where # is a character in the range of 40h to 53h ('@' to 'S'). 40h causes tab characters to be ignored; 41h sets tabs at every column (which causes tabs to be treated as spaces), 42h sets tabs to every second column, etc. If # is outside this range, the command sequence is ignored.

For example, the three-byte command string

```
ESC G D
```

sets tab spacing to every four spaces. See Table 1-4 (at the end of this chapter) for a complete list of codes and tab spacings.

2.2.18 Cursor Home - ESC H

Moves the cursor to the home (top left) position on the display.

2.2.19 Set Cursor Position - ESC I # *

Positions the cursor to the specified location (Figure 2-1 shows location numbering). The command has the form *ESC I # **, where # sets the row and * sets the column. For example, the string

```
ESC I B D
```

sets the cursor to row 2 (third row) and column 4 (fifth column). See Table 1-4 for a complete list of valid codes and cursor positions.

2.2.20 Erase to End of Screen - ESC J

Erases from the current cursor position to the end of the screen. The cursor position is unchanged.

2.2.21 Erase to End of Line - ESC K

Erases all displayed characters from the current cursor position to the end of the line. The cursor position is unchanged.

2.2.22 Set Contrast - ESC L

This command sets the display contrast. It has the form *ESC L #*, where # is in the range of 40h to 7Fh ('@' to DEL). The smaller the ASCII value of the character, the lower the contrast. The higher the ASCII value of the character, the higher the contrast.

2.2.23 Reset QTERM-II - ESC M

Resets the QTERM-II to its power-up state. This includes clearing all input and output buffers and the display, and resetting all parameters to the configuration set by the QDATA file, or a subsequent *Save Parameters to EEPROM* command.

2.2.24 Query Version - ESC N

This tells the QTERM-II to transmit its software version to the host. The version will consist of four ASCII characters in the format *vx.y*, where x and y are single ASCII digits.

2.2.25 Buzzer On/Off/Beep - ESC O

This command controls the buzzer. It has the form *ESC O #,* where # is

@ - turn buzzer off
A - turn buzzer on
B - beep for fixed duration

The fixed duration beep command (ESC O B) is identical to sending a *Bell* command (07h, ^G). The duration used for beeping is set by the *Set Buzzer Duration* command (ESC e) and by the QDATA file (default is 1/2 second).

2.2.26 LEDs On/Off/Blink/Toggle - ESC P

This controls the LEDs on the QTERM-II keypad, including the shift key LED. Any of these LEDs can be set to on, off or blink (the blink rate is set in the QDATA file and by the *Set LED Blink Rate* command). If a shift key is used, pressing the shift key will cause the proper state of the shift LED to override whatever state is set by this command. If no shift key is used, the shift LED is controlled solely by this command.

See Table 1-4 for a list of valid values for # and the effect of each such value. As an example of this command, the string

ESC P R

sets the second LED from the right to blink.

2.2.27 Insert/Replace Mode - ESC Q

In insert mode, when a character is displayed at any position on the display, all following characters are moved right one position. The character in the current cursor location is also moved before the incoming character is displayed. In delete mode, the character displayed simply over writes whatever character was there previously.

Valid values for # are

@ - set replace mode
A - set insert mode

2.2.28 Auto Wrap Mode - ESC R

The auto wrap mode determines what happens when the cursor moves past the end of a line. With auto wrap off, the cursor stays at the last position in the line. With auto wrap on, the cursor moves down to the first position in the next line.

If the cursor moves past the end of the last line, and auto wrap is on, then the action depends on the auto scroll mode. If auto scroll is off, the cursor will wrap to the first position of the line, but the display will not scroll. Otherwise, the display will scroll, and the cursor will return to the first position in the last line.

Valid values for # are

@ - auto wrap off
A - auto wrap on

2.2.29 Auto Scroll Mode - ESC S

Auto scroll mode determines what happens when the cursor moves past the end of the last line. With auto scroll off, the cursor will stay in the last position. With auto scroll on, the display scrolls (i.e., all lines move up, and the last line becomes blank), and the cursor moves to the first position in the last line.

Valid values for # are

@ - auto scroll off
A - auto scroll on

2.2.30 Auto Line Feed Mode - ESC T

With auto line feed off, when a carriage return is received, the cursor returns to the first position in the current line. With auto line feed on, the cursor moves to the first position in the next line, i.e. it acts as if both a carriage return and a linefeed had been received.

Valid values for # are

@ - auto line feed off
A - auto line feed on

2.2.31 Display Off - ESC U

This turns off the display. Once the display has been turned off, it is turned back on by either of two events:

- If the operator presses a key, the display is turned back on. However, the key is *not* transmitted to the host.
- If the host sends a command or character to the QTERM-II, the display is turned back on, and the command or character will be processed normally.

If the QTERM-II has a backlight, and it is on when the timeout occurs, the backlight is turned off along with the display. When the display is turned back on, the backlight is turned on if it was on when the timeout occurred.

Anything displayed on the LCD screen when the timeout occurs is lost; i.e. when the display is turned back on it will be blank. Custom character definitions are *not* lost.

The only thing the host can do that will not turn on the display is to issue the *Query Status* command (ESC W). The only reason to use this command is that it substantially reduces the power consumption of the QTERM-II.

2.2.32 Backlight On/Off/Toggle - ESC V

This command turns the backlight on and off.

Valid values for # are

@ - backlight off
A - backlight on
B - backlight toggle

2.2.33 Query Status - ESC W

The Query Status command (ESC W) returns a character indicating two things: the number of characters presently in the QTERM-II transmit buffer (0 to 15), and whether the display is on or off. Refer to Table 1-4 at the end of this chapter for the exact values returned for various conditions.

This is the only command which you can send to the QTERM-II which does not cause the display to be turned back on if the display is off when this command is received.

2.2.34 Query Cursor Position - ESC X

Returns the cursor position as two ASCII characters. These two characters are defined in the same way as for the *Set Cursor Position* (ESC I) command. See Table 1-4 for a detailed listing.

2.2.35 Query Character - ESC Y

Returns the character value for the character at the current cursor position.

2.2.36 Scroll Down - ESC Z

This causes the top three lines on the display to move down one line. The top line is cleared, and the cursor remains at the same absolute location on the display; i.e. it does not scroll down with the text.

2.2.37 Key Click/Repeat Mode - ESC a

This selectively enables and disables both key repeat and key click. The key click duration and key click rate are set in the QDATA file.

Valid values for # are

@ - click off, repeat off
A - click off, repeat on
B - click on, repeat off
C - click on, repeat on

This command globally enables and disables the key repeat. Which specific keys repeat is defined in the QDATA file.

2.2.38 Set Cursor Mode - ESC b

The QTERM-II cursor can be an underline cursor, a block cursor, neither or both. When neither is selected, no cursor is visible to the user.

Valid values for # are

@ - block off, underline off (no cursor)
A - block off, underline on
B - block on, underline off
C - block on, underline on

2.2.39 Set Shift Mode - ESC c

The shift key on the QTERM-II can operate in one of two ways:

- FUNCTION MODE, where the shift key stays shifted for one additional key press only
- LOCK MODE, where the shift key stays shifted until it is pressed a second time

For either mode, the shift LED can be enabled or disabled, and, if enabled, properly reflects the state of the shift key.

Valid values for # are

@ - function mode, shift LED enabled
A - lock mode, shift LED enabled
B - function mode, shift LED disabled
C - lock mode, shift LED disabled

Note that the shift LED enable/disable does not affect the shift operation itself, only the LED indicator. If you do not wish to have a shift key at all, you can disable the shift function by a parameter in the QDATA file.

2.2.40 Control *dig0* and *dig1* - ESC d

These two digital lines can be used for general control purposes, or they can be used to drive an external buzzer or horn.

Valid values for # are

@ - dig0 off (low)
A - dig1 off (low)
H - dig0 on (high)
I - dig1 on (high)
P - dig0 blink
Q - dig1 blink
X - dig0 buzzer
Y - dig1 buzzer
' - dig0 toggle
a - dig1 toggle

The blink mode blinks the line at the rate defined for LED blinking in the QDATA file. The buzzer mode causes the line to go on (high) when the internal buzzer is on, and to go off (low) when the internal buzzer is off. The toggle mode reverses the current state of the line. If the line is blinking and a toggle command is sent, the line will be toggled from its current state and will remain there (i.e. it will stop blinking).

2.2.41 Set Buzzer Duration - ESC e

This sets the duration for a "beep" command. The beep command can either be a *Bell* (07h, ^G) or the *Buzzer On/Off/Beep* command for beep (ESC O B). The valid range for # is 41h through 7Fh ('A' to DEL). See Table 1-4 for the duration set by each character.

2.2.42 Set Key Click Duration - ESC f

This sets the length of the key click, which is actually a very short beep from the buzzer. The valid range for # is 41h through 7Fh ('A' to DEL). See Table 1-4 for the duration set by each character. Use the *Key Click/-Repeat Mode* command (ESC a) to disable the key click.

2.2.43 Set Key Repeat Rate - ESC g

This sets the key repeat rate. The valid range for # is 41h through 7Fh ('A' to DEL). See Table 1-4 for the rate set by each character. See the *Key Click/Repeat Mode* command (ESC a) to globally disable key repeats. Use the QDATA file to disable key repeats for selected keys.

2.2.44 Set LED Blink Rate - ESC h

This sets the LED blink rate. The valid range for # is 41h through 7Fh ('A' to DEL). See Table 1-4 for the rate set by each code. See also *LEDs On/Off/Blink* (ESC P).

2.2.45 Save Parameters to EEPROM - ESC i

This command causes all parameters to be stored to EEPROM. Any QDATA defaults which have been modified by software commands will be replaced by their modified values.

Note that the EEPROM used in the QTERM-II has a guaranteed write endurance of 10,000 write cycles. Although this is far more than most applications will ever require, if you use this command frequently in your application, you should take this endurance into consideration.

2.2.46 Bar Code Control - ESC j

This command is used to control a bar code wand (if one is installed). The bar code operation is mostly transparent to the QTERM-II: commands sent to the bar code wand are simply transmitted as received, and characters received from the bar code wand are transmitted directly to the host. The bar code serial port does not support handshaking, which requires that the host interface run at 1200 baud or faster.

There are three valid values for #:

@ - disable receive from bar code wand
A - enable receive from bar code wand
B - transmit a string to the bar code wand

DISABLE RECEIVE FROM BAR CODE: if # = '@,' anything which the bar code transmits to the QTERM-II will be ignored.

ENABLE RECEIVE FROM BAR CODE: if # = 'A,' anything which the bar code transmits to the QTERM-II will be transmitted to the host. Bytes are transmitted exactly as received, so the host will receive the data as if the bar code wand were transmitting directly to the host serial port.

TRANSMIT TO BAR CODE: if # = 'B,' then the host is transmitting a string to the bar code wand. The complete form of this command has this format:

```
ESC j B * . . . .
```

where * indicates how many bytes are to follow in the '. . . .' string.

The first four characters (ESC j B *) are *not* transmitted to the bar code wand; the '. . . .' string is transmitted without any modification.

The character * must have a value of 41h through 7Fh ('A' to DEL), which indicates that the string will be from 1 byte to 63 bytes long. See Table 1-4 for a listing of values for * and the corresponding string length.

As an example of using this QTERM-II command to send a command string to the bar code wand, assume we wish to send this string to the wand:

```
ESC - y 4 Y F A I L
```

(This command would tell a Hewlett-Packard SmartWand to send the string 'FAIL' if a read failure occurred.) This string is nine characters long, so the exact string to transmit to the QTERM-II would be:

```
ESC j B I ESC - y 4 Y F A I L
```

The first four characters tell the QTERM-II to transmit the following nine characters directly to the bar code wand.

2.2.47 Transmit Buffer Flush - ESC k

If the host has transmitted an XOFF to the QTERM-II, and the user has pressed any keys, this command will clear the buffer, so that when the host sends XON to the QTERM-II, there will be nothing in the buffer for the QTERM-II to transmit to the host.

2.2.48 XON/XOFF Mode - ESC l

This command enables or disables the XON/XOFF operation of the QTERM-II.

Valid values for # are

@ - disable XON/XOFF operation
A - enable XON/XOFF operation

If you disable XON/XOFF operation, then any keys pressed by the user will be sent to the host immediately. If the host sends data fast enough to the QTERM-II to fill up the receive buffer, additional characters will be ignored.

2.2.49 User Area Read/Write - ESC m

This command allows you to store up to 63 bytes of your own information (such as serial numbers or parameters) in the QTERM-II nonvolatile EEPROM, then later read them from the terminal.

There are two valid values for #:

@ - read user data
A - write user data (followed by data)

READ DATA: if # = '@', the QTERM-II will transmit the data in the user area to the host in the following format:

```
# . . . .
```

where # is a character in the range of 40h to 7Fh, and indicates that 0 to 63 bytes of user data will follow, and '. . . .' is the corresponding number of user bytes. These bytes will be exactly what was originally stored, so they may be any 8-bit value. If # = '@' (0 bytes to follow), then there was no data stored in the user area.

WRITE DATA: to write user data, use the format:

```
ESC m A # . . . .
```

where # is in the range of 41h to 7fh ('A' to DEL), and indicates that from 1 to 63 bytes of data are to follow, and '. . . .' are the data bytes to be stored.

These data bytes may be any 8-bit value. After the entire string has been received, the QTERM-II will respond by transmitting one character to the host:

06h - Acknowledge character (ACK), data stored propery

15h - Negative Acknowledge character (NAK), data not stored

The two reasons that the data would not be stored properly are if there was not enough area to store the data, or if there was a hardware failure. See the *Get Free User Area - ESC n* command for more information about storage area.

Refer to Table 1-4 for a complete list of values of # and the corresponding number of data bytes for both the read and write commands.

2.2.50 Get Free User Area - ESC n

This command tells the QTERM-II to transmit one character to the host. This character will be in the range of 40h to 7Fh ('@' to DEL) and will indicate the total size of the user area. This will always be 63 bytes (character 7Fh), unless your QDATA file is so large that there are fewer bytes available for the user area.

Note that the value returned by this command does not tell you how much *more* you can write to the User Area, but tells you how much *total* you can write to the User Area.

Table 1-4: Command Values for ASCII Characters.

Note: the column labeled “Time” refers to several commands:

Set Buzzer Duration	-T=duration of beep
Set Key Click Duration	-T=duration of key click
Set Key Repeat Rate	-2*T=period; 1 / (2*T) = rate
Set LED Blink Rate	-2*T=period; 1 / (2*T) = rate

ASCII Char	Hex Value	Decimal Value	Set Timeout Delay	Set Tab Spacing (spaces)	Query & Set Cursor Position		LEDs On/Off/Blink		Query Status		Time “T” (seconds)	Bar Code User Area String Length
					(row	col)	(led	state)	(ds	buf)		
@	40h	64	disable	disable	0	0	shft	off	on	0		
A	41h	65	0:20	1	1	1	1	off	on	1	0.05	1
B	42h	66	0:40	2	2	2	2	off	on	2	0.10	2
C	43h	67	1:00	3	3	3	3	off	on	3	0.15	3
D	44h	68	1:20	4		4	4	off	on	4	0.20	4
E	45h	69	1:40	5		5	5	off	on	5	0.25	5
F	46h	70	2:00	6		6			on	6	0.30	6
G	47h	71	2:20	7		7			on	7	0.35	7
H	48h	72	2:40	8		8	shft	on	on	8	0.40	8
I	49h	73	3:00	9		9	1	on	on	9	0.45	9
J	4Ah	74	3:20	10		10	2	on	on	10	0.50	10
K	4Bh	75	3:40	11		11	3	on	on	11	0.55	11
L	4Ch	76	4:00	12		12	4	on	on	12	0.60	12
M	4Dh	77	4:20	13		13	5	on	on	13	0.65	13
N	4Eh	78	4:40	14		14			on	14	0.70	14
O	4Fh	79	5:00	15		15			on	15	.075	15
P	50h	80	5:20	16		16	shft	blink			0.80	16
Q	51h	81	5:40	17		17	1	blink			0.85	17
R	52h	82	6:00	18		18	2	blink			0.90	18
S	53h	83	6:20	19		19	3	blink			0.95	19
T	54h	84	6:40				4	blink			1.00	20
U	55h	85	7:00				5	blink			1.05	21
V	56h	86	7:20								1.10	22
W	57h	87	7:40								1.15	23
X	58h	88	8:00				shft	toggle			1.20	24
Y	59h	89	8:20				1	toggle			1.25	25
Z	5Ah	90	8:40				2	toggle			1.30	26
[	5Bh	91	9:00				3	toggle			1.35	27
\	5Ch	92	9:20				4	toggle			1.40	28
]	5Dh	93	9:40				5	toggle			1.45	29
^	5Eh	94	10:00								1.50	30
_	5Fh	95	10:20								1.55	31
`	60h	96	10:40						off	0	1.60	32
a	61h	97	11:00						off	1	1.65	33
b	62h	98	11:20						off	2	1.70	34
c	63h	99	11:40						off	3	1.75	35

continued

Table 1-4: Command Values for ASCII Characters.

ASCII Char	Hex Value	Decimal Value	Set Timeout Delay	Set Tab Spacing (spaces)	Query & Set Cursor Position (row col)	LEDs On/Off/Blink (led state)	Query Status (ds buf)	Time "T" (seconds)	Bar Code User Area String Length
d	64h	100	12:00				off 4	1.80	36
e	65h	101	12:20				off 5	1.85	37
f	66h	102	12:40				off 6	1.90	38
g	67h	103	13:00				off 7	1.95	39
h	68h	104	13:20				off 8	2.00	40
i	69h	105	13:40				off 9	2.05	41
j	6Ah	106	14:00				off 10	2.10	42
k	6Bh	107	14:20				off 11	2.15	43
l	6Ch	108	14:40				off 12	2.20	44
m	6Dh	109	15:00				off 13	2.25	45
n	6Eh	110	15:20				off 14	2.30	46
o	6Fh	111	15:40				off 15	2.35	47
p	70h	112	16:00					2.40	48
q	71h	113	16:20					2.45	49
r	72h	114	16:40					2.50	50
s	73h	115	17:00					2.55	51
t	74h	116	17:20					2.60	52
u	75h	117	17:40					2.65	53
v	76h	118	18:00					2.70	54
w	77h	119	18:20					2.75	55
x	78h	120	18:40					2.80	56
y	79h	121	19:00					2.85	57
z	7Ah	122	19:20					2.90	58
{	7Bh	123	19:40					2.95	59
|	7Ch	124	20:00					3.00	60
}	7Dh	125	20:20					3.05	61
~	7Eh	126	20:40					3.10	62
del	7Fh	127	21:00					3.15	63

CHAPTER 3.

USING THE QDATA FILE

3.1 QDATA Files

A **QDATA file** is used to configure your terminal. There are three reasons you may want to use a QDATA file:

- You want to change what keys send when they are pressed or design your own display characters (this *must* be done with the QDATA file).
- You want to change some of the other defaults (such as beep length, tab settings, etc.).
- You want to protect your development investment by modifying the terminal from its factory-default operation.

If the third reason is important to you, be sure to consider modifying the <*qdata retrieve mode*> identifier, to ensure that your investment remains protected.

With your terminal you will have received a disk (PC-compatible) with a variety of files. Two of these files are

QDATA File	An ASCII (text) file that specifies values for all of the configurable parameters of the QTERM-II.
QSETUP.EXE	A program which loads the QDATA file into the QTERM-II.

There are actually several versions of the QDATA file on the disk:

QDATA24.V30	For 24-key terminals (version 3.0).
QDATA40.V30	For 40-key terminals (version 3.0).
QDATA24W.V30	For 24-key terminals with wide-temperature displays (version 3.0).
QDATA40W.V30	For 40-key terminals with wide-temperature displays (version 3.0).

You will only need to use the QDATA file that matches the hardware on your terminal. The QDATA file for your QTERM-II is already set up with default values for all parameters. You will modify only the parameters which you want to change from their default.

Your QTERM-II is delivered with the default QDATA file already installed. You only need to modify the appropriate QDATA file and load it into the terminal if you want to change any of the default values. If not, you can skip this entire chapter.

Note that all QTERM-II parameters can be modified using software commands (see Chapter 2), *except for key strings and custom display characters!* The only way to modify what a key sends when it is pressed, or to redefine the custom display characters, is by modifying the QDATA file and loading it into the QTERM-II using the QSETUP program.

3.2 Using QSETUP

Your distribution disk includes the QSETUP.EXE program, which is used to load the QDATA file into your QTERM-II. This program actually performs three tasks:

- load a QDATA file into the QTERM-II
- retrieve a QDATA file *from* the QTERM-II
- load the operation code file (QCODE file) into the QTERM-II

To use QSETUP, you *must* use power-on setup to set the QTERM-II's baud rate and data format to 9600/8n1. These are the only settings that will work with QSETUP.

You must also connect your terminal to your PC's COM port and supply power to it. If you do not have an EIA-232 terminal, you will need an interface unit. See Chapter 1 for more information.

When you first run QSETUP, you will see an introductory screen that shows the version number. This version *must* match the version of the QDATA file you are going to load. See the *Versions* section below for more information.

The next screen will give you a choice of COM ports, if your PC has more than one. Select the port to which you have connected your terminal.

3.2.1 Loading a QDATA File

To load a QDATA file from disk into the QTERM-II, select the *Load QDATA* menu item. You will be asked for the file name. Enter the file name, along with the extension, then press the *Enter* key.

QSETUP will now open the specified file, scan it for errors, and, if none are found, load it into the QTERM-II. This will take about 10 seconds.

If any errors are found, QSETUP will display an error message indicating the problem. Use your editor to fix the problem, then try to load it again.

3.2.2 Loading a QCODE File

Also included on your distribution disk is a file called QCODE.V30. This is the actually microprocessor code for running the QTERM-II. If the code in the QTERM-II somehow gets corrupted, you can load this file into the terminal to get it running again.

If you suspect your terminal has corrupted code, call Beijer Electronics for technical assistance in reloading the QCODE file.

3.2.3 Retrieving a QDATA File

If you select the third menu item on the QSETUP menu screen, you will be asked for a file name into which QSETUP will put the retrieved QDATA information. Specify a file name that does not exist.

QSETUP will retrieve the QDATA information from your terminal and put it into a disk file. This file is similar to the default files supplied with your terminal, except for two changes:

- The retrieved file will have no comments in it.
- The retrieved file will have exactly one identifier per line, so it will have many more lines than the default QDATA file (which puts several identifiers on one line in the key sections).

Since neither of these affects the parameter content of the retrieved QDATA file, this retrieved file can be downloaded into any QTERM-II without modification.

3.3 Basic QDATA File Concepts

Identifiers. The QDATA file contains "identifiers" used by QSETUP to determine how to configure the QTERM-II. These identifiers appear in two formats:

[....] and <....>

The most important rule about editing the QDATA file is NEVER DELETE OR EDIT ANY OF THE IDENTIFIERS, AND NEVER CHANGE THEIR ORDER!

You will edit the *values* assigned to the identifiers, but you may not edit the identifiers, delete the identifiers or change the order in which they appear.

Versions. You *must* use the same version of the QDATA file and the QSETUP program as the version of code in your QTERM-II.

- Your QDATA version is noted in the file name extension (i.e. QDATA40.V30 is version 3.0) and is the first identifier in the QDATA file.
- The QSETUP version is displayed on the first screen when you start the QSETUP program.
- The version of software in your QTERM-II is displayed if you hold down any key and apply power.

If you find that you have different QDATA, QSETUP and QTERM-II code versions, contact Beijer Electronics for updated software and instructions. Do *not* attempt to use different versions of these items together; you will probably corrupt the QTERM-II operational code.

QDATA File Size. The QDATA file which you edit can be as large as you want. When QSETUP sends the QDATA information to the QTERM-II, it only sends actual parameters and key strings. *The maximum amount of information from the QDATA file which can be stored by the QTERM-II is about 1000 bytes.*

The default QDATA files generate about 270 bytes of information, so you can add about 730 bytes of data to keystrings before you exceed the 1000 byte limit. QSETUP will tell you if there is too much data to be stored by the QTERM-II.

ASCII Editor. You may use any editor to modify the QDATA configuration file, but the output from the editor must be in ASCII format, without any embedded codes.

Included on your QTERM-II distribution disk is the TED3 editor (TED3.COM), which you can use to edit the

QDATA file. The file TED3.DOC provided documentation for the editor and its commands.

Uppercase/Lowercase. For keywords (such as "on" or "off"), case does not matter; i.e. "ON" is the same keyword as "on." For quoted text strings, (used for key strings), the string is assigned exactly as shown in the QDATA file.

Comments. Comments may be added to the file as desired. A comment starts with a semicolon and ends at the end of a line (at the carriage return).

Whitespace. Whitespace may be added to the file as desired. This means you can add spaces, tabs, carriage returns and line feeds anywhere you would like. The only exceptions are that you cannot split a keyword (such as "off") or identifier (such as [shift mode]) with whitespace, and that in quoted text strings (which are assigned to keys), anything within the quotes will be assigned to the key.

Since you can add and remove carriage returns, you can place each identifier on a separate line, or you can have several identifiers on one line, as long as the order in which they appear is not changed.

3.4 Parameter and Key String Sections

The QDATA file has two sections: the *parameter section* at the top, followed by the *key string section* at the bottom. The types of values you can assign to the identifiers is different in each section.

3.4.1 Parameter Section

In the parameter section, you will assign one of two items to each identifier:

- keywords (on, off, replace, df8n1)
- numbers (13, x1F)

Keywords. For every identifier to which you assign a keyword, the keywords which are permissible are listed in a comment following the identifier. The keyword you choose must be one of those listed in the comment. Never use quotes with keywords.

Numbers. For identifiers that accept numbers rather than keywords, the number can be specified in one of two ways:

- decimal numbers (49, 27)
- hexadecimal numbers (x0D, x8C)

These numbers are always positive integer numbers within the range specified by the comments. Hexadecimal constants start with the letter 'x'. Decimal and hexadecimal constants are equivalent; use whichever one is easier at any time.

3.4.2 Key String Section

This is the last section of the file, and follows the parameter section. The format for the identifiers for each key is

<**k**#*>	- unshifted key strings
<**sk**#*>	- shifted key strings
<**kr**#*	- unshifted key release strings
<**skr**#*>	- shifted key release strings

where:

# = row (top = row 0)
* = column (right = column 0)

Examples of this notation:

<k21> is for unshifted key string for row 2 and column 1

<skr32> is for shifted key release string for row 3 and column 2.

Since the QTERM-II electronics can support up to eight rows and six columns of keys, there are identifiers for 48 unshifted and 48 shifted key strings, as well as identifiers for 48 unshifted and 48 shifted key release strings. All of these must remain in the file; however, you will have no reason to assign key strings to keys that do not exist on your keypad.

In the key string section the type of values you can assign to a key are

- strings enclosed in single quotes ('abcde')
- decimal numbers (49, 27)
- hexadecimal numbers (x0D, x8C)
- keywords (\l1on, \d0off)
- a combination of the above, separated by commas

Strings. Text strings can contain any printable ASCII characters and are enclosed in single quotes. If you need to include a single quote in your text string, terminate the string and specify the ASCII value as a number (x27 hexadecimal or 29 decimal).

Numbers. Numbers are always positive integer numbers within the range 0 to 255. Hexadecimal numbers start with the letter 'x'. Decimal and hexadecimal numbers are equivalent; use whichever one is easier at any time.

Keywords. Keywords in key string assignments serve two purposes: inserting pauses (delays) in the transmission of a key string and controlling LEDs and digital outputs. A complete list of valid keywords is shown in Table 3-1.

Examples of Key Strings. The simplest example of a key string is where pressing the key sends just one character, regardless of whether the shift key has been pressed:

```
<k43> '1'  ; send ASCII '1'
<sk43> '1' ; send same if shifted
<kr43>     ; send nothing on release
<skr43>    ; send nothing on shifted release
```

The next example shows a key which sends a string when pressed but beeps if pressed when shifted (the number 7 is a 'bell' character):

```
<k27> 'Hello There'; send string
<sk27> 7   ; beep if shifted
<kr27>     ; nothing
<skr27>    ; nothing
```

Below is a key string which uses keywords to control the keypad LEDs and to pause (see Table 3-1). The key controls LED1 when unshifted (press and the LED goes on; release and it goes off), and controls LED2 when shifted (press and it goes on; release and it goes off ½-second later):

```
<k32> 'Turn 1 on',\l1on
                         ; send string, turn L1 on

<sk32> 'Turn 2 on',\l2on
                         ; send string, turn L2 on

<kr32> 'Turn 1 off',\l1off
                         ; send string, turn L1 off

<skr32> 'Turn 2 off',\p,10,\l2off
                         ; send string, pause ½-second, turn
                         L2 off
```

The final example shows how to combine strings, numbers and keywords on one key string assignment. Note the splitting the key assignment onto multiple lines (as done here) is perfectly acceptable.

```
<k62> 'Pause here ',\p,10,'then turn dig0 on ',
                 \d0on,'then beep ',7,'then send a cr/lf '
                 'to the host',x0D,10
```

Key String Restrictions. A key string may be from 0 to 255 bytes long. Each keyword requires two bytes.

Table 3-1: Valid Key String Keywords.

Keywords		Purpose
\10on	\10off	control LED0 (shift)
\11on	\11off	control LED1
\12on	\12off	control LED2
\13on	\13off	control LED3
\14on	\14off	control LED4
\15on	\15off	control LED5
\d0on	\d0off	control *dig0*
\d1on	\d1off	control *dig1*
\p,nnn		insert pause (nnn = 50 ms ticks)

Pause constants may be from 1 to 255 which corresponds to 50 ms to 12.75 second pauses. Multiple pause constants can be specified in one string.

Be sure to separate each component of a key string (text strings, numbers and keywords) by a comma, as shown in the examples above.

3.5 Custom Display Characters

The QTERM-II is capable of displaying eight user-defined characters. Each of these characters consists of an eight row by five column dot matrix, where you specify which of the dots are on or off.

You also define what code will cause the character to be displayed by the QTERM-II. You may choose any 8-bit value (code) from 00h through FFh, *except* values from 07h through 1Bh (see Appendix B for a chart of this). This allows you to redefine up to eight of the standard ASCII character bit patterns. For example, you could redefine the hex value x62, which normally displays a 'b', to display a small box (□).

Each custom character is defined in the QDATA file as a string of eight byte values, where each byte specifies the dot pattern for one row of dots. For each byte, the upper three bits are set to 0 and the lower five bits correspond to the five dots on the row. If a bit is set to 1, the dot will be dark (on); if the bit is set to 0, the dot will be light (off).

As an example, Figure 3-1 shows the default dot pattern of custom character 6, which is a little humanoid figure. From this dot pattern, the following bit assignments are derived:

Row	7	6	5	4	3	2	1	0	Hex

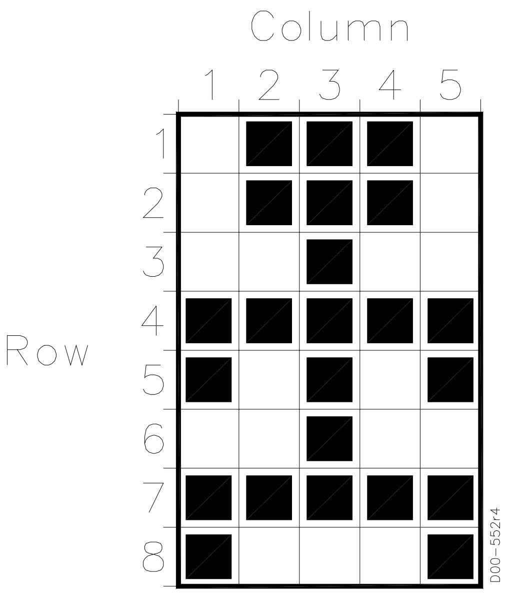


Figure 3-1. Example Custom Character.

As you can see in your default QDATA file, the hexadecimal values listed in the right column above are the values which are used to define the custom character. This particular dot patter is assigned to the ASCII character x06 in the default QDATA file.

Default dot patterns and character assignments for all eight custom characters are shown in the chart in Appendix B.

CHAPTER 4.

QTERM-II HARDWARE

4.1 Handheld QTERM-II

The dimensions of the handheld version of the QTERM-II are shown in Figure 4-1. This figure also shows the standard keypad legend for the bottom five rows of the 24-key keypad (the top row is labeled to user-specifications).

The housing is made from impact-resistant ABS and is colored black. (See Appendix D if you need a different color housing.) The housing is not waterproof, but it can be subjected to moderate rain or splash without harm.

The handheld QTERM-II uses either a 6-pin modular connector or an (optional) DB9 female for connection to the external cable. The unit can be ordered with either connector exiting from the top or the bottom of the housing. If desired, the 6-pin modular connector can easily be switched from one end of the housing to the other.

The pin assignments for the connectors for each of the available interfaces are shown in Table 4-1. (The lines *dig0* and *dig1* are discussed in section 4.5.3.) Figure 4-2 shows the pin numbering for each of the connectors used in the QTERM-II. For reference, Table 4-5 (at the end of this chapter) shows the pin assignments used by the COM ports on PC-style computers for both DB25 and DB9 connectors.

4.2 Battery Back QTERM-II

When ordered with the battery back, the handheld QTERM-II does not have any power connection, i.e. the +V pin on the external connector is not used.

The battery back uses six "AA" size alkaline batteries. Typical battery life is shown in Table 1-3. Dimensions of the handheld with battery back are shown in Figure 4-1.

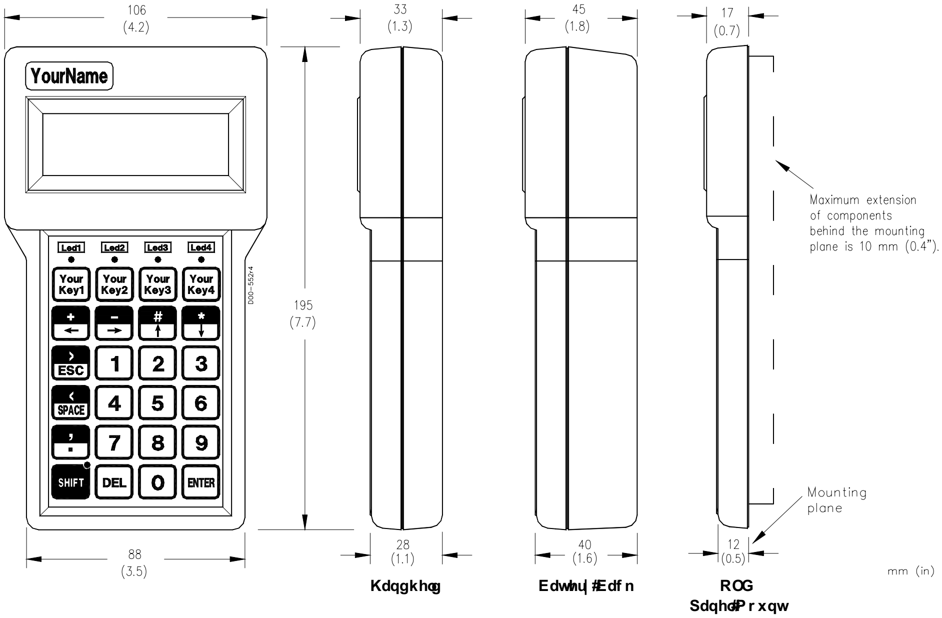
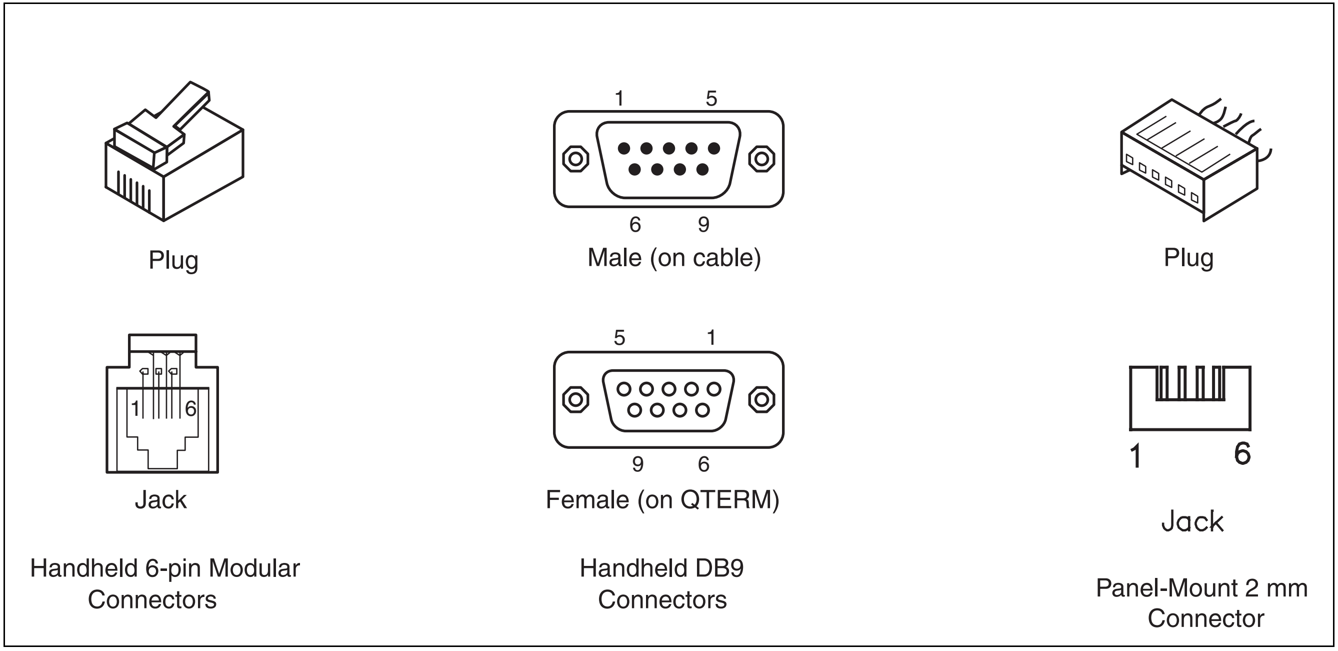


Figure 4-1. Dimensions of the Handheld, Old Panel-Mount and Battery-Back QTERM-II.



The battery back includes a power-supply board with an on/off switch. This board generates 5 volts DC from the batteries, so the regulator option should *not* be ordered with the battery back option.

Because of the battery compartment, the connectors on a battery-back unit can only exit from the top of the case.

4.3 Panel-Mount QTERM-II

4.3.1 Old Panel-Mount QTERM-II

The old panel-mount QTERM-II is mounted directly onto your instrument or enclosure. Figure 4-1 shows the dimensions of the old panel-mount QTERM-II. Figure 4-4 shows the mounting cutout required to install the QTERM-II.

The old QTERM-II is mounted to the panel via four self-tapping screws that insert from the rear of the panel and screw into bosses on the QTERM-II enclosure. Screws are included with the QTERM-II; these are usable with panels up to about 4 mm thick. If you are using a thicker panel, use M3 (#4) self-tapping screws that are about 8 mm longer than the panel thickness.

A panel-mount gasket is available as an accessory.

Connection to the old panel-mount QTERM-II is made via a 2-mm terminal strip connector on the internal PC board, which is accessible from the back of the mounting panel.

Table 4-2. QTERM-II Pin Assignments.

Handheld 6-pin Modular	Handheld DB9	Panel-Mount 2 mm	QTERM-II 232/5-volt Buffered Function	QTERM-II 422 Function
1	3	1	receive	+receive
2	6	2	dig1*	-receive
3	2	3	transmit	+transmit
4	1	4	dig0*	-transmit
5	9	5	+V^	+V^
6	5	6	ground	ground
Tx and *Rx* directions are with respect to the QTERM-II *these lines are optional; see section 4.5.3 ^no connection with battery back units				

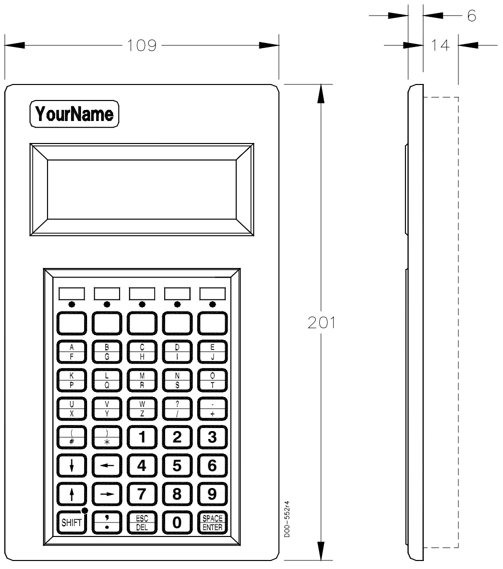


Figure 4-3. Dimensions of the New Panel-Mount.

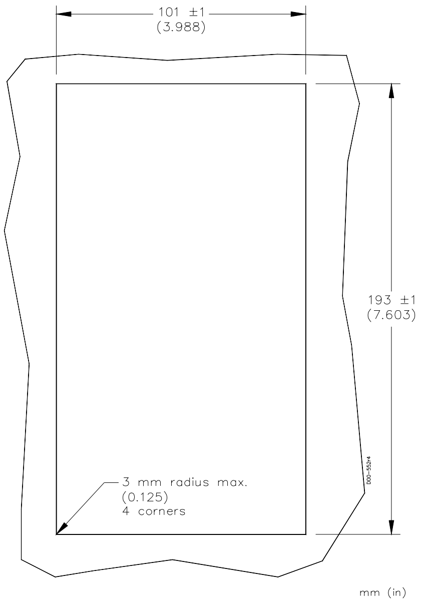


Figure 4-4. Mount the Panel-Mount QTERM-IIs.

The pin assignments for the old panel-mount QTERM-II connector are shown in Table 4-1; the lines *dig0* and *dig1* are discussed in section 4.5.3.

The connector pin numbering is shown in Figure 4-2.

4.3.2 New Panel-Mount QTERM-II

The new panel-mount QTERM-II is mounted directly onto your instrument or enclosure. Figure 4-3 shows the dimensions of the new panel-mount version. Figure 4-4 shows the mounting cutout required to install the new panel-mount QTERM-II.

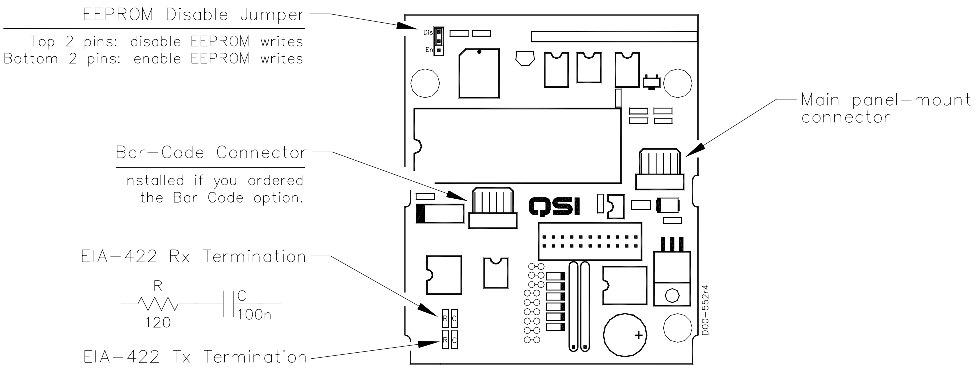

EEPROM Disable Jumper
Top 2 pins: disable EEPROM writes
Bottom 2 pins: enable EEPROM writes
Dis
En
Bar-Code Connector
Installed if you ordered the Bar Code option.
EIA-422 Rx Termination
R
120
C
100n
EIA-422 Tx Termination
Main panel-mount connector
QSI
D00-552r4

Figure 4-5. Installing the EIA-422 Termination Resistor; Locating the Bar Code Connector.

The new panel-mount QTERM-II is mounted to the panel with two brackets that attach from the rear with four self-tapping screws.

Connection to the new panel-mount QTERM-II is done in the same fashion as the old panel-mount version.

4.4 Interfaces

4.4.1 EIA-232 Interface

With proper cables and good grounding, the EIA-232 QTERM-II can communicate up to about 15 meters at its top speed of 9600 baud.

The QTERM-II does not monitor or control any of the EIA-232 modem-control lines (such as RTS and DTS). Handshaking between the host and the QTERM-II is done using software XON/XOFF commands. The XON/XOFF operation can be disabled (via software commands or the QDATA file) if you do not wish to have it operate.

4.4.2 EIA-422 Interface

With proper cables and grounding, the EIA-422 interface can communicate up to a distance of about 1000 meters. Since the EIA-422 version of the QTERM-II requires four communication lines and two power lines, the *dig0* and *dig1* lines are not available.

The EIA-422 unit does not have the 120S line termination resistor often used to terminate an EIA-422 interface. If you need this resistor, you can install it on the main PC board (for handheld units, remove the back of the case). Figure 4-5 shows where the resistor is to be soldered in. Do *not* attempt to remove the PC board from the case!

For volume orders, Beijer Electronics can supply the QTERM-II with this resistor installed.

4.4.3 5-volt Buffered Interface

The 5-volt Buffered interface is the same as a EIA-422 interface, except that it only uses the *+transmit* and *+receive* lines. The two remaining pins are used for the *dig0* and *dig1* lines, as with the EIA-232 version.

When used to communicate with another 5-volt Buffered device, the QTERM-II can operate at distances up to about 5 meters. The advantage of the 5-volt Buffered QTERM-II is that it is lower power than the EIA-232 version and uses the same 5-volt logic levels as come directly out of a UART (i.e. the idle state is at 5 volts, and the active state is at 0 volts).

Table 4-3. Bar Code Connector Pin Assignments.

Handheld DB9	Panel-Mount 2 mm	Function
2	1	receive from wand
3	3	transmit to wand
9	5	5 volts to wand
7	6	ground

Although the signal levels are the same as at a UART, the transmit and receive lines are buffered and can be connected to live signals whether or not the QTERM-II is powred (turned on).

4.5 Other QTERM-II Hardware

4.5.1 Display

The QTERM-II display is a 4-line by 20-character supertwist unit. The ASCII character set and eight user-defined characters can be displayed. Hosts which transmit 8-bit data can also display an additional 64 characters including Greek letters, katakana characters, non-English alphabetic characters and math symbols.

Appendix B is a chart which shows what the QTERM-II does with every possible 8-bit value it can receive. Note that the ASCII portion of the chart (the first 128 characters) is similar, *but not identical*, to the true ASCII chart shown in Appendix A.

If you ordered the lighted display option, the software commands shown in Chapter 2 allow you to turn the backlight on and off. Without the lighted display, these commands have no effect.

If you have a wide-temperature display, the backlight will operate identically to the lighted display.

4.5.2 Keypad

The lower-left key on both the 24- and 40-key keypads is a shift key and has an associated LED to indicate the shift function (this can by changed by the QDATA file; see Appendix D). Above the top row of keys is a row of user-programmable LEDs (four on the 24-key keypad, and five on the 40-key keypad).

Software commands and the QDATA file allow you to control both key clicks (on or off) and key repeat (on or off, independently programmable for each key). Every key can be programmed to return an unshifted and a shifted character or string (up to 255 characters, including delays, subject to the limitation noted in Chapter 3). Every key can also be programmed to return an unshifted and shifted release character or string.

4.5.3 Digital Outputs, *dig0* and *dig1*

The QTERM-II has two programmable digital output bits (*dig0* and *dig1*) which can be used to control external devices. (Note that these are not available on the EIA-422 version due to lack of connector pins).

Software commands can be used to set these lines high or low, or to have one or both function as an external buzzer or horn signal. Both lines are 74HC bus-driver outputs and can sink or source up to about 20 mA of current at 0 or 5 volts, respectively.

If either *dig0* or *dig1* is programmed to act as an external buzzer signal, it acts as an enable signal: when the line is low (0 volts), it should shut off the external buzzer or horn; when the line is high (5 volts), it should turn on the external buzzer or horn. These lines should only be used to drive an external transistor, which in turn drives the external buzzer.

4.5.4 Buzzer

The QTERM-II includes an audio buzzer, which is used for key clicks, for beeping in response to a "bell" character (^G, 07h), and for programmable-duration buzzing. For applications in environments that are too loud for the buzzer to be heard, the digital outputs (described above) can be programmed to trigger an external device when buzzer commands are received. (These signals are not available on EIA-422 units due to lack of connector pins.)

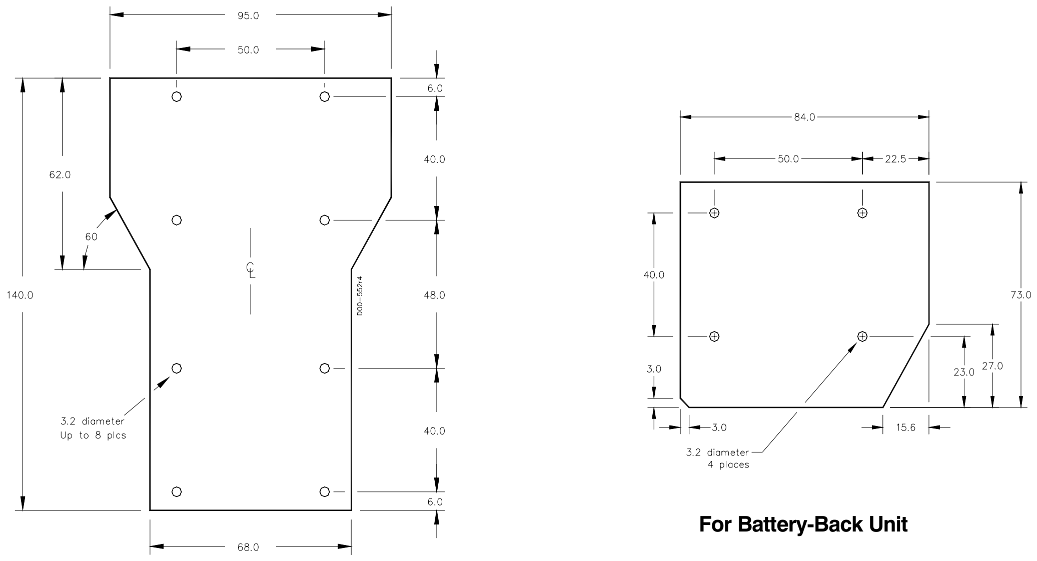
4.5.5 Regulator Option

The standard QTERM-II requires a 4.8- to 5.5-volt regulated, SELV, DC power source. An optional regulator allows the QTERM-II to be operated from a 5.5- to 24-volt SELV, DC source. The power supply should have a current limit on its output of 5 Amperes. Limiting may be inherent to the supply or may be provided by supplementary devices.

4.5.6 Bar Code Option

If you ordered the bar code option with your QTERM-II, you have an additional connector. On handheld units, this is a DB9 connector which exits from the same end of the case as the 6-pin modular connector; on panel-mount units, this is a 6-pin, 2 mm connector (see Figure 4-2). Table 1-2 shows the pin assignments for both the panel-mount and handheld connectors. The handheld DB9 female will mate directly with the DB9 male on an HP SmartWand bar code reader.

The bar code serial port operates at 1200 baud with a data format of 8n1. The QTERM-II host port must operate at 1200 baud or faster when using the bar-code option. This serial port does not support XON/XOFF handshaking.



For Battery-Back Unit

For Handheld Unit

Figure 4-6. Size of Additional PC Board.

4.5.7 The EEPROM Disable Jumper

Figure 4-5 shows the location of the EEPROM disable/ enable jumper. This is shipped in the *enable* position, which allows the EEPROM to be modified with QDATA downloads or the *Save Parameters to EEPROM* command.

If you do not want the EEPROM to be modified under any conditions, move this jumper from the *enable* to the *disable* position. This will prevent any writes to the EEPROM, including downloading of QCODE or QDATA files.

4.5.8 Adding Another PC Board

If you are using a handheld or battery back QTERM-II, you can install an additional PC board in the back of the case.

This board can be used, of course, for whatever purpose your application requires.

Figure 4-6 shows the physical size of the printed circuit board, as well as the allowable component heigth. There are plastic bosses molded into the back of the case; you can use all or just some of these for mounting your PC board. The ones you do not wish to use can be easily clipped off. For the bosses that you do want to use, place a 3.2 mm diameter hole in your PC board. After the PC board is installed over the bosses, you can either press on a retainer snap ring (Eaton-Tinnerman part number T99220SS-67-01-10780) or heat sink the bosses.

4.6 QTERM-II Specifications

Table 1-3 lists various specifications for the QTERM-II. Note that all components of the QTERM-II, other than the display, will operate over the industrial temperature range of -40 to 85 °C. Table 4-4 lists current consumption of the various versions of the QTERM-II.

Table 4-4. Environmental and Power Specifications.

Parameter	Limits
Operating Temperature Range (except for display)	-40 to 85°C
Standard/Lighted Display usable temperature range	-10 to 60°C
Wide-temperature Display usable temperature range	-20 to 70°C
Storage temperature, all components	-40 to 85°C
Maximum humidity range, all components	0 to 95%, non-condensing
Operating voltage range, without regulator option	4.8 to 5.5 volts
Operating voltage range, with regulator option	5.5 to 24 volts
Battery Back Option Battery life, EIA-232 Battery life, EIA-422 Battery life, 5-volt Buffered Battery life, backlight on	 150 hours 190 hours 190 hours 43 hours

Table 4-5. QTERM-II Current Consumption (mA).

Version	Ireg Active	Ireg Disp Off
EIA-232 version	25	15
5-volt Buffered version	22	12
EIA-422 version	22	12
Add for backlight (when on)	56	-
Add for wide-temp	2.0	0.0
Add for each LED on keypad (when on)	4.3	4.3
Add for regulator	0.4	0.4
Add for bar code wand (w/o wand)	3.0	3.0

Table 4-6. Pin Assignments for PC-Style COM Ports.

COM Ports with DB25 Male Connectors		COM Ports with DB9 Male Connectors	
Pin	**Function**	**Pin**	**Function**
2	PC transmit	1	CD *
3	PC receive	2	PC receive
5	CTS *	3	PC transmit
6	DSR *	5	ground
7	ground	6	DSR *
8	CD *	7	CTS *
22	RI *	8	RI *

*These lines normally can be left unconnected. Some PCs may require that one or more of them be pulled to 5 volts through a pullup resistor (about 300 ohms)

APPENDIX A.

ASCII CHART

		Most Significant Digit (hex)							
		3	4	5	6	7	8	9	:
Least Significant Digit (hex)	3	QXO	GOH	VS	3	C	S	c	s
	4	VRK	GF4	$	4	D	T	d	t
	5	VW[	GF5	%	5	E	U	e	u
	6	HW[	GF6	&	6	F	V	f	v
	7	HRW	GF7	’	7	G	W	g	w
	8	HQT	QDN	(	8	H	X	h	x
	9	DFN	V\ Q	)	9	I	Y	i	y
	:	EHO	HWE	*	:	J	Z	j	z
	;	EV	FDQ	+	;	K	[	k	{
	<	KW	HP	,	<	L	\	l	|
	D	OI	VXE	-	=	M	]	m	}
	E	YW	HVF	.	>	N	^	n	~
	F	II	IV	/	?	O	_	o	¡
	G	FU	JV	0	@	P	‘	p	¢
	H	VR	UV	1	A	Q	a	q	£
	I	VL	XV	2	B	R	b	r	GHO

0345R3

NUL = blank
SOH = start of header
STX = start of text
ETX = end of text
EOT = end of transmission
ENQ = enquiry
ACK = acknowledge
BEL = bell
BS = backspace
HT = horizontal tab
LF = line feed
VT = vertical tab
FF = form feed
CR = carriage return
SO = shift out
SI = shift in
SP = space
DLE = data link escape
DC1 = device control 1 (XON)
DC2 = device control 2
DC3 = device control 3 (XOFF)
DC4 = device control 4
NAK = negative acknowledge
SYN = synchronization
ETB = end of text block
CAN = cancel
EM = end of medium
SUB = substitute
ESC = escape
FS = file separator
GS = group separator
RS = record separator
US = unit separator
DEL = delete/rubout

APPENDIX B.

QTERM-II CHARACTER CHART

The chart on the next page shows how the QTERM-II responds to each of the 256 possible values of characters which it can receive.

Where a dot pattern is shown, sending the corresponding code will cause the QTERM-II to display the dot pattern at the current cursor location.

Numbers in circles refer to these notes:

ŒThese bytes are always ignored. They cannot be used for remapping custom characters.
œThese bytes are normally ignored but may be used for remapping custom characters if desired.
–This is a space character.

Custom characters, and the codes to which they are assigned, are defined in the QDATA file (see Chapter 3). The following restrictions apply to assigning the codes:

- Custom characters may *not* be mapped to any code from 07h through 1Bh (inclusive). This includes the Œ bytes as well as the control codes which are used by the QTERM-II.
- Custom characters *may* be mapped to any other code, i.e. 00h through 06h and 1Ch through FFh.
- The characters shown in the chart from 00h through 06h and at 1Ch are the default custom character definitions. You may change the character dot patterns and map them to different codes using the QDATA setup file.

Other notations in the chart are

BEL	bell (beep) command
BS	backspace
HT	horizontal tab
LF	line feed
CR	carriage return
XON	XON character
XOFF	XOFF character
ESC	escape character
DEL	delete character

Note that, although the left half of this chart is similar to the ASCII chart in Appendix A, there are differences.

	Most Significant Digit (hex)															
Least Significant Digit (hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	↑	①	③	0	@	P	`	p	②	②	③	ー	タ	ミ	α	p
1	↓	XON	!	1	A	Q	a	q	②	②	。	ア	チ	ム	ä	q
2	↘	①	"	2	B	R	b	r	②	②	「	イ	ツ	メ	β	θ
3	⌒	XOFF	#	3	C	S	c	s	②	②	」	ウ	テ	モ	ε	∞
4	[illegible]	①	$	4	D	T	d	t	②	②	、	エ	ト	ヤ	μ	Ω
5	±	①	%	5	E	U	e	u	②	②	・	オ	ナ	ユ	σ	ü
6	[illegible]	①	&	6	F	V	f	v	②	②	ヲ	カ	ニ	ヨ	ρ	Σ
7	BEL	①	'	7	G	W	g	w	②	②	ァ	キ	ヌ	ラ	g	π
8	BS	①	(	8	H	X	h	x	②	②	ィ	ク	ネ	リ	√	x̄
9	HT	①	)	9	I	Y	i	y	②	②	ゥ	ケ	ノ	ル	⁻¹	y
A	LF	①	*	:	J	Z	j	z	②	②	ェ	コ	ハ	レ	j	千
B	LF	ESC	+	;	K	[	k	{	②	②	ォ	サ	ヒ	ロ	×	万
C	LF	¿	,	<	L	¥	l	|	②	③	ャ	シ	フ	ワ	¢	円
D	CR	③	-	=	M	]	m	}	②	③	ュ	ス	ヘ	ン	Ⱡ	÷
E	①	③	.	>	N	^	n	→	②	③	ョ	セ	ホ	゛	ñ	③
F	①	③	/	?	O	_	o	DEL	②	③	ッ	ソ	マ	゜	ö	█

(Notes Œ, œ and – are on the previous page.)

APPENDIX C.

QTERM-II COMMAND SUMMARY

This appendix is an abbreviated summary of all of the available QTERM-II software commands. More detailed descriptions of the commands are in Chapter 3.

Refer to Table 2-4 for a complete list of acceptable parameter values for applicable commands. If a command parameter is out of the valid range, the QTERM-II ignores the entire command.

Note that the timing shown for the execution of the various commands is only approximate. Many factors can affect the execution time, so it is impossible to give exact figures.

Command	String	Timing (ms)	Notes & Parameters
display character		0.7 14.0 37.0	typical in replace mode typical in insert mode worst case in insert mode
Bell (^G)	07H	0.6	other buzzer commands not executed until beep is over
Backspace (^H)	08H	0.8	
Horizontal Tab (^I)	09H	0.8	1.1 ms worst case
Line Feed (^J)	0AH	0.8	up to 35 ms if auto scroll is on
Vertical Tab (^K)	0BH		same as line feed
Form Feed (^L)	0CH		same as line feed
Carriage Return (^M)	0DH	0.8	up to 35 ms if auto scroll is on
XON (^Q)	11H	0.4	
XOFF (^S)	13H	0.4	
Delete	7FH	1.2	
Cursor Up	ESC A	0.8	
Cursor Down	ESC B	0.8	
Cursor Right	ESC C	0.8	
Cursor Left	ESC D	0.8	
Clear Screen	ESC E	4.4	
Set Timeout Delay	ESC F#	1.2	see Table 2-4
Set Tab Spacing	ESC G#	1.2	#= @ causes tabs to be ignored A to S for tab set every 1 to 19 spaces
Cursor Home	ESC H	3.7	

Command	String	Timing (ms)	Notes & Parameters
Set Cursor Position	ESC I#*	1.5	#= @ to C for row 0 to 3 *= @ to S for column 0 to 19
Erase to End of Screen	ESC J	6.0 12.0 18.0 24.0	if cursor is on row 3 if cursor is on row 2 if cursor is on row 1 if cursor is on row 0
Erase to End of Line	ESC K	3.0 5.8	typical maximum (cursor in column 0)
Set Contrast	ESC L#	1.2	#= @ to DEL for lightest to darkest
Reset QTERM-II	ESC M	300.0	
Query Version	ESC N	0.5	time to load characters into transmit buffer
Buzzer On/Off/Beep	ESC O#	1.2	other buzzer commands not executed until beep is over #= @ turn buzzer off = A turn buzzer on = B beep for 1/2 second
LED Control On/Off/Blink/Toggle	ESC P#	1.2	#= @ to U; see Table 2-4
Insert/Replace Mode	ESC Q#	1.2	#= @ set replace mode A set insert mode
Auto Wrap Mode	ESC R#	1.2	#= @ auto wrap off A auto wrap on
Auto Scroll Mode	ESC S#	1.2	#= @ auto scroll off A auto scroll on
Auto Line Feed Mode	ESC T#	1.2	#= @ auto line feed off A auto line feed on
Display Off	ESC U#	2.0	key press or character received turns display back on
Backlight Mode	ESC V#	1.2	#= @ backlight off A backlight on B backlight toggle
Query Status	ESC W	0.5	time to load characters into transmit buffer seeTable 2-4 for returned values
Query Cursor Position	ESC X	0.5	time to load characters into transmit buffer see Table 2-4 for returned values
Query Character	ESC Y	0.5	time to load character into transmit buffer
Scroll Down	ESC Z	35.0	
Key Repeat/Click Mode	ESC a#	1.2	#= @ click off, repeat off #= A click off, repeat on #= B click on, repeat off #= C click on, repeat on

Command	String	Timing (ms)	Notes & Parameters
Set Cursor mode	ESC b#	1.2	#= @ block off, underline off (no cursor) A block off, underline on B block on, underline off C block on, underline on
Set Shift Mode	ESC c#	1.2	#= @ function mode, shift indicator enabled A lock mode, shift indicator enabled B function mode, shift indictator disabled C lock mode, shift indicator disabled
Control *dig1* and *dig2*	ESC d#	1.2	see section 2.2.40 for values
Set Buzzer Duration	ESC e#	1.2	see Table 2-4
Set Key Click Duration	ESC f#	1.2	see Table 2-4
Set Key Repeat Rate	ESC g#	1.2	see Table 2-4
Set LED Blink Rate	ESC h#	1.2	see Table 2-4
Save Parameters to EEPROM	ESC i	6.0	see note in section 2.2.45
Bar Code Control	ESC j#	1.2 1.2 *	#= @ disable receive from bar code A enable receive from bar code B send string, *= about 8 ms per character
Transmit Buffer Flush	ESC k	1.2	
XON/XOFF Mode	ESC l#	1.2	#= @ disable XON/XOFF operation A enable XON/XOFF operation
User Area Read/Write	ESC m#		timing depends on baud rate # = @ read user; returns byte count and data bytes; see section 2.2.49 for details A write user data; follow “A” with byte count and data bytes; see section 2.2.49 for details
Get Free User Area	ESC n	2.2	see Table 2-2

APPENDIX D.

CUSTOMIZING THE QTERM-II

There are many ways in which the QTERM-II can be customized for your application. In fact, there are so many ways that we have "standard" customization and "custom" customization.

D.1 Standard Customization

There are four different standard customizations for the QTERM-II called Level 1, Level 2, Level 3 and Level 4.

LEVEL 1: included with every QTERM-II, for no extra charge, is custom silkscreening of a company name, LED labels and the top row of keys. This no-charge screening has the following restrictions:

- All text will be typeset in a Helvetica-style font.
- No "reversed" text can be used, although two words can be screened onto the top row of keys.
- Silkscreening will be done with black ink.

LEVEL 2: if you need "reversed" text on the top keys, or you want a graphic logo, special font or different color ink, you need Level 2 customization:

- Silkscreening can include your actual graphic logo.
- Top keys can included reversed (shifted) functions.
- You can specify the type font which you want to use.
- You can specify any color of ink.

There is a nominal one-time art charge for the special artwork and/or for the special color of ink. After this one-time charge, there is *no* extra charge on the QTERM-II units, regardless of the quantity ordered.

LEVEL 3: If you need custom legends on more than the top row of keys, then you need Level 3 customization:

- The entire keypad legend (24 or 40 keys) can have your custom text, including shifted keys anywhere you desire.
- The logo label will include your company name and graphic logo to your specifications.
- Due to the process used, legends are restricted to just two colors: white/black or white/blue (PMS 280).

For Level 3 customization, there is a nominal one-time charge for the required artwork, then a small additional charge for each unit for producing the keypad legend and logo label. Level 3 customization can be done in any quantity.

LEVEL 4: If you need an all-custom QTERM-II, then Level 4 customization is for you. Here you can receive:

- Full custom keypad art, in multiple colors, made with the same embossing process as our standard keypad legends.
- Full custom company name and logo, in the colors you specify, on the logo label.

There is a one-time NRE charge for the custom art for these all-custom legends, and the legends must be purchased in minimum quantities of 25 pieces. *Note that you do **not** have to buy 25 terminals to have Level 4 customization!* If desired, you can order the custom keypad legends and logo labels, which Beijer Electronics will then stock. We then install the keypad and logo labels on units as you order them. There is no additional charge for this service, beyond your original cost for art and keypad/logo labels.

CUSTOM ENCLOSURE COLOR: If you wish to have a custom-color enclosure, you can select from 23 "standard" colors and pay only a one-time color charge. There is no additional unit charge, regardless of the number of terminals purchased (even just one unit!).

If one of the 23 "standard" colors does not suit your requirements, contact Beijer Electronics for information on a non-standard color enclosure.

D.2 Custom Customization

We recognize that no matter how flexible we make the QTERM-II software or how many color and labeling options we offer, there will be applications where a customer may need something more.

There are many additional items which can be modified on the QTERM-II to suit your needs. These include:

- **Custom Keypads.** The keypad for the QTERM-II is a plug-in unit. If your application requires, we can design a different unit, a different number of keys, different size keys, more or fewer LEDs or even some small and some large keys.
- **Larger Memory.** If you need many long key strings, you may exceed the 1000-byte limit for your QDATA file. The QTERM-II can be configured to provide about 20 Kbytes of QDATA memory if your application needs this.
- **Custom Additional PC Board.** As indicated in Chapter 1, the handheld QTERM-II has a provision for an additional board to be mounted in the back of the case. If you have a need for such a board, we would be happy to provide a quote for its design and construction.

Of course, the cost-effectiveness of any of these will depend on the design time involved and the quantity ordered. If you feel that, for your application, the volume or necessity may warrant such all-custom design, contact Beijer Electronics directly for more details and a quotation.