

# 8720

**ALPHANUMERIC DISPLAY  
FOR MODICON PLC's  
M84 / 584 / 884 / 984**

**PRODUCT MANUAL**

PM-8720  
REVISION 2

**QUARTECH**  
  
**Corporation**

The product described in this document can have a variety of uses, the user and those responsible for applying this equipment must satisfy themselves as to the acceptability of each application and the use of the unit. Under no circumstances will QUARTECH CORPORATION be responsible or liable for any damage, including indirect or consequential losses resulting from the use, misuse, or application of the unit.

The text, illustrations, charts, and examples included in this document are intended solely to help explain applications of the product. Due to the many variables associated with specific uses or applications, QUARTECH CORPORATION cannot assume responsibility or liability for actual use based upon the data provided in this document.

No patent liability is assumed by QUARTECH CORPORATION with respect to the use of circuits, information, equipment, or software described in this document.

No part of this document may be reproduced, stored in a retrieval system, transmitted in any form or by any means, including electronic, mechanical, photocopying or otherwise, without the prior express written permission of QUARTECH CORPORATION.

This document is printed in the U.S.A. and is subject to change without notice.

© 1992 - QUARTECH® CORPORATION - ALL RIGHTS RESERVED

## **Quartech Corporation**

15943 Angelo Drive  
Macomb Twp, MI 48042

### **New Numbers**

Phone: 810-781-0373  
Fax: 810-781-0378

# TABLE OF CONTENTS

<u>Subject</u>	<u>Page</u>
1.0 Introduction.....	2
2.0 Triggering Messages.....	2
2.1 Register Immediate.....	2
2.2 Register Stack.....	3
2.3 Bit Triggered.....	3
3.0 Using the Keyboard.....	3
3.1 Entering Messages (F1 key).....	4
3.2 Deleting Messages (F2 key).....	6
3.3 Editing Messages (F3 key).....	6
3.4 Static Display (F4 key).....	7
3.5 Message Transfer (F5 key).....	8
3.6 Print Document (F6 key).....	8
3.7 Trigger Mode Configure (F7 key).....	9
3.8 Printer Setup (F8 key).....	12
3.9 Entering PLC Mode (F9 key).....	12
3.10 Clearing Memory (F10 key).....	13
4.0 Embedding Variables In Messages.....	13
4.1 Decimal Display of 2, 3, 4, or 5 Digits.....	14
4.2 Binary Display.....	15
4.3 Bit Status Defined.....	16
4.4 Bar Graph.....	17
4.5 Displaying Delimiter Characters.....	17
4.6 Variable Definition Syntax Table.....	18
5.0 Programming Examples.....	19
6.0 Message Logging.....	20
7.0 Error Messages.....	21
8.0 Communication Cable.....	22
9.0 Electrical Requirements.....	23
10.0 Installation Notes.....	23
11.0 ASCII Character Set.....	24
12.0 Mechanical Dimensions.....	25

## **1.0 INTRODUCTION**

The DataMate 8720 is a 40 character alphanumeric display designed specifically for the Gould MODBUS family of Programmable Controllers. The 8720 communicates with the processor through its MODBUS port eliminating the need for I/O modules and special communications driver programs. Over 32 thousand bytes of memory are available for message storage. With a maximum of 512 messages allowed. This allows an average of 60 bytes/message.

The 8720 includes an easy to use message compiler which requires only the addition of an IBM PC/XT or compatible keyboard which plugs directly into the 8720. An important feature of the 8720 is the ability to embed variable data within the message. Additional standard features include 8720 to 8720 file transfers, 8720 to/from IBM PC or compatible file transfer and printer document support.

## **2.0 TRIGGERING MESSAGES**

There are three methods available for triggering messages: Register Immediate, Register Stack, and Bit Triggered.

### **2.1 REGISTER IMMEDIATE**

When set into this triggering mode the 8720 will poll an assigned holding register (pointer register) waiting for a message number to be placed into it. When a message number is found, the corresponding message will be displayed. Once triggered, the message will remain on the display until another message number is placed into the pointer register. It is important to note that the 8720 is continually polling the pointer register, so any time the value of this register changes a new message will be triggered, overwriting the old message.

If the message number 0 is placed into the pointer register one of two displays will result. If a message was present on the display at the time the 0 was placed into the pointer register, the message will continue to be displayed. If a 0 was placed in the pointer at the time the unit is connected to the PLC, the following message will be displayed:

**WAITING FOR MESSAGE**

If a message number is placed in the pointer register for which a message was never constructed, the following message will be displayed:

**NO MESSAGE #XXX**

Where XXX is the message number that could not be found.

## **2.2 REGISTER STACK**

This mode is identical to the Register Immediate mode except after the first message is displayed it must be acknowledged in order to display the next message. If a new message is triggered before the previous message has been acknowledged the new message will not overwrite the old one. Instead, the message number will be saved in the 8720's RAM. Thus, this function will allow messages to be stacked up and displayed only after the operator acknowledges the prior message. This is useful when multiple messages may be triggered over a short period of time or if the operator is not always present to receive the displayed messages.

The message stack is a first in first out (FIFO) type and will store a maximum of 25 messages. Since the 8720's RAM is not battery backed, it will lose the stored message numbers during power outage.

As messages are removed from the stack through the use of the acknowledge contact, the messages will be displayed in a FIFO order. When the final message is acknowledged, the following message is displayed indicating that the stack is empty:

WAITING FOR MESSAGE

If a 0 is placed in the pointer register, it will not be added to the stack since it is not a valid message number.

## **2.3 BIT TRIGGERED**

In this mode, the 8720 polls an assigned group of holding register bits. When a low to high transition is detected on one of the bits, the corresponding message is displayed. Once a message has been triggered it will remain on the display, regardless of the status of the trigger bit, until another message is triggered.

A table of holding registers are used for triggering messages. The number of holding registers (blocks) needed depends on the number of messages used. Each holding register contains 16 bits. Each one of these 16 bits are used to trigger 1 of 16 messages. Therefore, a table with a length of 4 blocks can trigger up to 64 messages (4x16).

For more information on this topic see TRIGGER MODE CONFIGURE (section 3.7).

## **3.0 USING THE KEYBOARD**

Messages are programmed into the 8720 via an IBM or compatible PC/XT keyboard. Keyboards such as the IBM AT will not work because different control codes are used by this keyboard. The Keytronic 5151 keyboard will work if the unit is not placed in its enhanced mode where the specifically designated cursor keys are used. If the keyboard is set to the normal IBM mode, it can be used. We suggest that if you do not have an extra keyboard that one be purchased from QUARTECH. Our keyboard is designated as 8785.

The keyboard is plugged into the DIN connector located on the bottom of the 8720. Plug the keyboard into the unit and then apply power. Do not connect or disconnect the keyboard with power applied to the 8720.

When the 8720 is powered up with a keyboard attached, a key must be held down in order for the 8720 to detect that it is attached. We recommend that you use the Space Bar. If a key is not pressed, the 8720 will enter the PLC mode.

The following function keys are used to program messages into the 8720:

F1	ENTER MESSAGE	F2	DELETE MESSAGE
F3	EDIT MESSAGE	F4	STATIC DISPLAY
F5	MESSAGE TRANSFER	F6	PRINT DOCUMENT
F7	MODE CONFIGURE	F8	PRINT SETUP
F9	ENTER PLC MODE	F10	CLEAR MEMORY

### 3.1 ENTERING MESSAGES (F1 KEY)

The [F1] key allows the entry of a new message. The final or displayed message to be shown to an operator is limited to a maximum length of 40 characters including spaces; however, embedded variables generally require more characters for definition than are actually displayed. Consequently, even though the final (displayed) message is limited to 40 characters, the constructed or stored messages may be larger. The 8720 will allow the entry of 512 messages with an average constructed message length of 60 characters. The maximum constructed message length is 110 characters. If the constructed message is greater than 40 characters, it is scrolled during the entry and editing stages to allow it to be displayed to the programmer. When the [F1] key is pressed, the following screen will appear:

```
***** ENTRY *****  
ENTER #
```

Messages are identified by message numbers. Before a message can be entered a message number must be assigned. Valid message numbers are 1-512. Messages may be entered in any numerical order. If the message number has already been used, the following message will be displayed:

```
***** # USED *****  
SELECT FUNCTION KEY
```

If the number entered has not been used, the screen will be blank except for a cursor character (solid block). At this point the message can be entered. Both upper and lower case characters are available.

To edit prior to saving the message, the [BACKSPACE] key , may be used. The [BACKSPACE] key is destructive, i.e. it removes any character that the cursor moves to.

Variables may be entered at any place in the message. See EMBEDDING VARIABLES IN MESSAGES (section 4.0) and PROGRAMMING EXAMPLES (section 5.0) for more details. The number of variables allowed for a given message is only limited by the 40 character screen.

Any existing message may be Cloned into a new message. This is very helpful when the new message is very similar to an existing message and it would be easier to edit a clone of the existing message than to construct a new one from scratch. The procedure to clone a message is as follows:

Press [F1] and enter the new message number. The block cursor is in the first character position of the first line. The cursor must be in this position and no other key must have been pressed. If you have moved the cursor and even repositioned it back to the original location, this function will not be allowed. At this point the [CTRL] key is pressed and the following message will be displayed:

```
***** CLONING *****  
ENTER #
```

Enter the message number to be cloned and press the [ENTER] key. The cloned message will now be duplicated into the new message number and you are now in the Edit mode, refer to EDITING MESSAGES (section 3.3).

If the message number to be cloned is not found, the following message will be displayed:

```
NO MESSAGE #XXX  
SELECT FUNCTION KEY
```

When entering a long list of messages, it is easy to forget which message you are working on. If at any time during the entry of a message, the message number that you are working on is forgotten; pressing the [TAB] key will display the following message:

```
MESSAGE #XXX
```

This message will remain for approximately two seconds and will then return to the original display.

Once the [ENTER] key has been pressed, the message is stored and the following screen will appear:

```
** MESSAGE STORED **  
SELECT FUNCTION KEY
```

If the [ESC] key is pressed at any time prior to pressing the [ENTER] key, the message will not be saved.

### 3.2 DELETING MESSAGES (F2 KEY)

The [F2] key is used to delete or remove a message from memory. When the [F2] key is pressed, the following screen will appear:

```
***** DELETE *****  
ENTER #
```

Entering the number that corresponds to the message that is to be removed and pressing the [ENTER] key will remove the message and display the following screen:

```
**** PLEASE WAIT ***  
DELETING MESSAGE
```

After the message has been deleted the following screen will appear:

```
* MESSAGE DELETED *  
SELECT FUNCTION KEY
```

The deleted message number can now be reused.

Pressing the [ESC] key prior to pressing [ENTER] key will back you out of this function without deleting the message.

### 3.3 EDITING MESSAGES (F3 KEY)

The [F3] key will allow you to edit any message that has already been entered. When the [F3] key is pressed, the following screen will appear:

```
***** EDIT *****  
ENTER #
```

Enter the message number to be edited and press the [ENTER] key. The message will be displayed for editing. Notice that a cursor is blinking and it may be moved by using the arrow keys, [<--] [-->]. If the message is longer than 40 characters, the message will be scrolled. Three methods of editing are available at this point; TYPE OVER, INSERT, and DELETE.

To TYPE OVER the existing message with corrections, simply move the cursor to the location of the type over and begin entering the new characters.

To INSERT new characters into the message, move the cursor to the location of the insert, press the [INS] key, a space will be created and the new character may now be typed in. The insert is a push type and the letter after the cursor will be moved to the right as the new character(s) are entered. After the insert has been finished, press the [INS] key a second time to exit the insert mode and the space that was created to allow the insert will be removed. If you wish to insert characters at the end of the message, move to the end and start adding the new characters. The [INS] key does not have to be pressed at this location; but if you do, it will function as stated.



If the [DEL] key is pressed while doing an insert, the insert function will be cancelled. The [ENTER] key is inactive when Insert is active.

To DELETE an existing character(s) from the message, move to the location of the delete, press the [DEL] key and the character under the cursor will be deleted. Each time the [DEL] key is pressed a character will be removed. If the key is held down, it will repeat.

If at any time during the editing of a message, the message number that you are working on is forgotten; pressing the [TAB] key will display the following message:

MESSAGE #XXX

This message will remain for approximately two seconds and will then return to the original display.

When editing is complete, press the [ENTER] key and the following message will appear:

\*\*\*\* PLEASE WAIT \*\*\*  
STORING MESSAGE

When the message has been saved the following screen will appear:

\*\* MESSAGE STORED \*\*  
SELECT FUNCTION KEY

If the [ESC] key is pressed, the edit change will not be stored and you will be able to select another function key.

### **3.4 STATIC DISPLAY (F4 KEY)**

The [F4] key is used to display a given message as it will be seen by the operator. This display is very useful in determining the arrangement of characters that will be seen by the operator. When the [F4] key is pressed, the following screen will appear:

\*\*\*\*\* DISPLAY \*\*\*\*\*  
ENTER #

Enter the message number to be displayed and press the [ENTER] key. The message will be displayed as it will be seen by an operator. Variables will be shown with block characters (solid) and messages that use the BIT STATUS DEFINED (section 4.3) display only the longest label that has been defined. At this point you may press any function key or if the [ESC] key is pressed it will take you back to the Main Screen.

### **3.5 MESSAGE TRANSFER (F5 KEY)**

The [F5] key allows the user to transfer messages between two 8720's, or between an 8720 and the model 8780 File Transfer Software. In addition to messages being transferred all data for TRIGGER MODE CONFIGURE (section 3.7) and PRINTER SETUP (section 3.8) are transferred. When the [F5] key is pressed, the following screen will appear:

S = SEND TO DEST.  
R = REC. FROM SOURCE

If [S] is pressed, the 8720 will transfer its messages to the receiving unit and the screen will change to:

SEND DATA TO  
DESTINATION DEVICE

If [R] is pressed, the 8720 will receive messages from the transferring unit and the screen will change to:

RECEIVE DATA  
FROM SOURCE DEVICE

To transfer the messages between two 8720's, connect an 8792-10 cable (or compatible cable) between the two units. The unit that has the messages should be put in the SEND mode and the second unit should be put in the RECEIVE mode.

To transfer the messages between an 8720 and disk storage, the model 8780 File Transfer Software must be used. In order to do this, an 8794-10 or 8795-10 cable must be connected between the 8720 and the serial port on the computer. To store the messages on disk, the 8720 will SEND the messages to the computer and the computer will RECEIVE the messages from the 8720. The reverse is true when sending messages from the computer to the 8720. Refer to the 8780 software for further information.

When the transfer is complete, the screen will change to:

SELECT FUNCTION KEY

NOTE: Messages can be transferred from an 8720 version 1.2 or lower to a version 1.3 or higher however, messages cannot be transferred from an 8720 version 1.3 or higher to a version 1.2 or lower.

### **3.6 PRINT DOCUMENT (F6 KEY)**

The [F6] key is used to print a listing of the messages in the 8720. This printout is used to document the messages only. The listing will be as follows:

MESSAGE TRIGGER METHOD: XXXXXXXXXXXX  
REGISTER ADDRESS -- XXXXX (If this function is used)  
POINTER REGISTER ADDRESS -- XXXXX (If this function is used)  
ACKNOWLEDGE CONTACT ADDRESS -- XXXXX-XX (If this function is used)

001 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
003 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX (Messages are printed in  
004 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX numerical order, and if  
: message is not used then  
: it is not printed)  
:  
512 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

MEMORY BYTES REMAINING : XXXXX

The listing will print out all messages that have been stored in the 8720. When the [F6] key is pressed, the following screen will appear:

ENTER DATE MM/DD/YY

As you type the date in, the numbers will be displayed. If an error is made prior to entering the last digit, press the [ESC] key to exit and press [F6] to reenter the date. If an error is made while entering the last digit, power will have to be removed from the 8720 and reapplied since the keyboard is ignored as soon as the last digit has been entered.

Once the date has been correctly entered, the 8720 will pause for a few seconds then display the following screen:

MESSAGE PRINTING  
CTS IS XXX Where XXX is either OFF or ON.

As data is being sent to the printer, the CTS message will toggle ON and OFF as the printer accepts the data.

See PRINTER SETUP (section 3.8) for printer baud rate, parity and auto line feed requirements.

### 3.7 TRIGGER MODE CONFIGURE

The [F7] key is used to determine the method that messages are triggered from the PLC. Messages are triggered one of three ways; Register Immediate, Register Stack, or Bit Triggered. For more information see TRIGGERING MESSAGES (section 2.0).

When the [F7] key is pressed, the following message will appear:

```
SELECT TRIGGER MODE
B=BIT R=REG. [r]
```

NOTE: The selection of upper or lower case will determine if messages are logged out to a printer through the keyboard port. If a lower case 'r' or 'b' is used, then the messages are not logged out to a printer. If an upper case 'R' or 'B' is used, then the messages are logged out to a printer. See MESSAGE LOGGING (section 6.0).

If you desire to trigger messages using the Register Immediate or the Register Stack mode then select REG. If you desire to trigger messages using the Bit Triggered mode then select BIT.

REGISTER IMMEDIATE/STACK: Upon selecting 'r' or 'R', the following screen will be displayed:

```
ENTER POINTER REG.
> [40001]
```

For the M84 and 484, the pointer register and the acknowledge address are entered as a five digit number even though these PLC's will not normally accept a value of that size. Example: Pointer Register at 4001 is entered as 40001, Acknowledge Address at 1001 is entered as 10001.

The Pointer Register is the holding register that the 8720 looks at for the number of the message to display. The Pointer Register can be any legal holding register number. The default value is 40001. If a different value is desired, enter the number for the Pointer Register and press the [ENTER] key. The following screen will then be displayed:

```
ENTER ACK. ADDRESS
> [00000]
```

If no acknowledge bit is assigned (i.e. ack. address = 00000), then the 8720 will be in Register Immediate mode. The default value is 00000, so pressing [ENTER] will put the 8720 in Register Immediate mode. However, if an acknowledge bit is assigned the 8720 will be put into Register Stack mode.

In Register Stack mode, the acknowledge bit is used by an operator to acknowledge receipt of a message. When the acknowledge bit is set and then cleared, the next message in the stack will be displayed.

The acknowledge bit can be any address including input and output addresses. To enter the acknowledge address, enter the desired value and press [ENTER]. This will complete the trigger mode set up.

BIT TRIGGERED: Upon selecting 'b' or 'B', the following screen will be displayed:

```
ENTER BEGINNING REG.
> [40001]
```

The Beginning Register is the holding register or output coil designated as the starting address of the message trigger table. The default value is 40001. If a different value is desired, enter the number and press [ENTER].

The following screen will then be displayed:

```

ENTER NUMBER OF BIT
BLOCKS >          [01]
  
```

The number of bit blocks (trigger bit blocks) determines the number of messages that can be triggered. 1 block is capable of triggering 16 messages. Therefore divide the number of messages programmed by 16 and round off to the next highest number. This number will be the number of blocks needed.

The following table indicates the relationship between register addresses, message numbers, and trigger bits.

X = Beginning Register.

TRIG BIT BLOCK ADDR	BIT NUMBER															
	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
X	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
X+1	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
X+2	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
X+3	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
:																
:																
X+32	512	511	510	509	508	507	506	505	504	503	502	501	500	499	498	497

← MESSAGE NUMBERS →

Message trigger bits are scanned by the 8720 from lowest to highest number. The first OFF to ON transition detected will cause a new message to be displayed. The scanning processor causes a natural prioritizing. If two messages are triggered simultaneously, the lowest number will be displayed first. Since the highest number message would remain on the display, it should be of most importance.

In addition, it is a wise practice to not leave gaps between assigned message numbers. The reason is that the 8720 will have to scan those unused bits looking for a triggered bit and will thus increase the amount of time that a triggered bit must be ON to insure that the 8720 will see it.

If a message request bit is set but no message was programmed, the 8720 will display the message:

NO MESSAGE #XXX

At any time during this setup mode, the [ESC] key may be pressed and you will be returned to the Main Screen.

### 3.7 PRINTER SETUP (F8 KEY)

The [F8] key allows the operator to set up the printer defaults. These defaults include baud rate, parity, and auto line feed. Eight (8) data bits are always sent. When the [F8] key is pressed, the following screen will appear:

```
BAUD  >A=300   C=4800
RATE  B=1200  D=9600
```

This screen is the baud rate selection screen. The default setting is noted by the > and is 300 baud, press [ENTER] key to accept this value. If you desire an alternate setting, enter the value by pressing the key corresponding to the value (i.e. [D] for 9600 baud). Upper and lower case characters are allowed. The next screen is as follows:

```
N=NONE E=EVEN O=ODD
SELECT PARITY  [N]
```

If the parity desired is the default value, then press the [ENTER] key. If not select another value by pressing the key corresponding to the value (i.e. [E] for Even). The next screen is as follows:

```
AUTO LINE FEED?
Y=YES   >N=NO
```

If your printer automatically executes a line feed after receiving a carriage return, press [Y] for yes. If the default of N (no) is desired, press [ENTER] key. If you are unsure about this, then select [N]. If the printed document is double spaced then enter as [Y].

### 3.9 ENTERING PLC MODE (F9 KEY)

The [F9] key allows the programmer to enter PLC mode while the keyboard is attached to the 8720. This is useful to test messages, because it is impossible to fully test variable definitions without actually going into the PLC.

If errors are encountered, control will automatically be returned to the keyboard. You can then edit the message and retest it.

### **3.10 CLEARING MEMORY (F10 KEY)**

The [F10] key allows the programmer to clear the memory of the 8720. When the [F10] key is pressed, the following screen will appear:

```
*** CLEAR MEMORY ***
ARE YOU SURE?  Y=YES
```

Pressing any character other than [Y] will abort this function. Pressing [Y] will clear all messages from memory, set defaults for the triggering and printing modes and display the following screen:

```
**** PLEASE WAIT ***
CLEARING MEMORY
```

When memory has been cleared or if this function was aborted the following screen will appear:

```
SELECT FUNCTION KEY
```

### **4.0 EMBEDDING VARIABLES IN MESSAGES**

One of the most powerful features of the 8720 message display, is the ability to display variable data - such as a counters accumulator - in real time. To do this you need only to supply the address of the counter, and specify how the data is to be displayed (i.e. decimal, binary, or bar graph).

Several characters have been reserved for the purpose of specifying how the data is to be displayed; these characters also act as delimiters for the address information. For example: Assume the address 40100 is the accumulator of a counter. To display 5 digits of decimal data in a message, you would include the following string in the message: \$40100\$

The \$ delimiter specifies 3 digit decimal data, and also denotes the beginning and ending of the address information.

The variable data to be displayed is not restricted to timers and counters. By using other delimiter characters you can display I/O and bit status. The correct entry format (syntax) must be used when specifying embedded variables. It is also important to note that variable addresses are not tested until individual messages are triggered during PLC operation. Refer to VARIABLE DEFINITION SYNTAX TABLE (section 4.6).

The following examples indicate the proper syntax for the various data displays.

NOTE: Variable address do not require that leading zeros to be entered.

## 4.1 DECIMAL DISPLAY OF 2, 3, 4, OR 5 DIGITS

This display format can reference a 30000 or 40000 series addresses.

The \* delimiter will result in the least significant two (2) digits of the specified register to be displayed.

The # delimiter will result in the least significant three (3) digits of the specified register to be displayed.

The ! delimiter will result in least significant four (4) digits of the specified register to be displayed.

The \$ delimiter will result in five (5) digits of the specified register to be displayed.

The following format can be used to display variable data:

2 digit data:       \* $\alpha\beta\beta$ \*

3 digit data:       # $\alpha\beta\beta$ #

4 digit data:       ! $\alpha\beta\beta$ !

5 digit data:       \$ $\alpha\beta\beta$ \$

where;

$\alpha\alpha$  = decimal position (optional).

$\beta\beta$  = 30000 or 40000 series address.

A decimal point may be included (optional) in the display by immediately following the start delimiter by a decimal point and then a single digit that indicates position.

$\alpha\alpha$ FIELD (OPTIONAL)	DECIMAL POSITION
.0	XXXXX.
.1	XXXX.X
.2	XXX.XX
.3	XX.XXX
.4	X.XXXX
.5	.XXXXX



Assume the contents of register 40030 is 56789, then the resulting displayed values will be as follows for the associated delimiter format:

2 DIGIT	
DELIMITER FORMAT	RESULTING DISPLAY
*40030*	89
*.040030*	89.
*.140030*	8.9
*.240030*	.89

3 DIGIT	
DELIMITER FORMAT	RESULTING DISPLAY
#40030#	789
#.040030#	789.
#.140030#	78.9
#.240030#	7.89
#.340040#	.789

4 DIGIT	
DELIMITER FORMAT	RESULTING DISPLAY
!40030!	6789
!.040030!	6789.
!.140030!	678.9
!.240030!	67.89
!.340030!	6.789
!.440030!	.6789

5 DIGIT	
DELIMITER FORMAT	RESULTING DISPLAY
!30!	56789
\$.040030\$	56789.
\$.140030\$	5678.9
\$.240030\$	567.89
\$.340030\$	56.789
\$.440030\$	5.6789
\$.540030\$	.56789

## 4.2 BINARY DISPLAY

This display format can reference 00000, 10000, 30000, or 40000 series addresses.

The & delimiter will result in a 0/1 status display of 8 specified bits or I/O points.

Binary display:     & $\beta\beta\tau$ &

where;

$\beta\beta$  = 00000, 10000, 30000, or 40000 series address.

$\tau$  = display lower (0) or upper (1) 8 bits.

The & delimiter will result in the binary (1's & 0's) display of the specified address, or the binary display of the specified I/O point as well as the next seven points. If a 30000 or 40000 series address is used, you must specify a 0 or 1 for the  $\tau$  field to display the lower or upper bits.

Assume the contents of register 40030 is 6789, then the resulting displayed values will be as follows for the associated delimiter format:

DELIMITER FORMAT	RESULTING DISPLAY
&400300& &400301&	10000101 00011010
&00012&	Status of outputs 12 thru 19

### 4.3 BIT STATUS DEFINED

This display format can reference 00000, 10000, 30000, or 40000 series addresses.

The ^ delimiter will result in user defined labels for the 0 and 1 status of a specified bit. The first user defined label (up to 40 characters) is displayed if the bit is 0. If the bit is 1, the second label (up to 40 characters) is displayed.

Bit status: ^ββφφOFFLABEL:ONLABEL^

where;

ββ = 00000, 10000, 30000, or 40000 series address.  
φφ = bit position (if 30000 or 40000 address).  
OFFLABEL = label (up to 40 characters) if bit is zero.  
ONLABEL = label (up to 40 characters) if bit is one.

In this example address 40030 bit 16 is being looked at. After the colon the user defined label (up to 40 characters) is displayed if bit 16 is 0. If bit 16 is 1, the second label (up to 40 characters) is displayed.

DELIMITER FORMAT	BIT 16 VALUE	RESULTING DISPLAY
^4003016The bit is zero:The bit is one^	0	The bit is zero
^4003016The bit is zero:The bit is one^	1	The bit is one

Note: labels may use any character except the colon.

## 4.4 BAR GRAPH

This display format can reference a 40000 series address.

The @ delimiter will cause a bar graph to be produced based on the value in the specified register.

Bar graph: @ $\beta\beta\pi$ @

where;

$\beta\beta$  = 40000 series address.

$\pi$  = display graph on top line (0) or bottom line (1).

The full scale value of the bar graph is 20. Values greater than 20 will be considered full scale. The 8720 does not scale the value. To scale, divide the full scale raw data value by 20 to obtain the scaling factor. The raw data is then divided by the scaling factor to obtain the data that is assigned to the bar graph register.

Assume the contents of register 40030 is 5 and the contents of register 40035 is 17, then the resulting displayed values will be as follows for the associated delimiter format:

DELIMITER FORMAT	RESULTING DISPLAY
@400301@ (bar on bottom line)	-----
@400350@ (bar on top line)	-----

The unused display line may be used for additional information and/or data displays.

## 4.5 DISPLAYING DELIMITER CHARACTERS

The delimiters are also characters which can be displayed. Instructing the 8720 to display a delimiter character is accomplished by pressing the [ALT] key prior to pressing the delimiter key. A degree symbol (°) is produced when the [ALT] key is pressed.

If the [ALT] key is pressed twice the degree symbol is produced in the message (this character is useful in displaying temperatures). Note that when the messages are sent to a printer, the degree symbol may be replaced by some other character usually the approximate (~) character.

## 4.6 VARIABLE DEFINITION SYNTAX TABLE

The following table lists all the variable delimiters, there associated functions, and some typical variable definition syntax.

DELIMITER CHARACTER	DELIMITER DEFINITION	DELIMITER FORMAT	RESULTING DISPLAY
*	2 Digit Display - least significant 2 digits of specified register.	*40010* *.140010*	XX X.X
#	3 Digit Display - least significant 3 digits of specified register.	#40010# #.240010#	XXX X.XX
!	4 Digit Display - least significant 4 digits of specified register.	!40010! !.340010!	XXXX X.XXX
\$	5 Digit Display - least significant 5 digits of specified register.	\$40010\$ \$.440010\$	XXXXX X.XXXX
&	Binary Display - display 8 bit binary data.	&00017& &3001015&	XXXXXXXXX XXXXXXXXX
^	Bit Status Defined - 40 character message relative to status of a given bit for a specified address. Each label is up to 40 characters.	^00017OFF:ON^	AAA
@	Bar Graph Display - 20 position bar graph relative to the value in a specified register	@400300@ top @400351@ bot	----- -----

## 5.0 PROGRAMMING EXAMPLES

**EXAMPLE 1:** The first example will combine characters and variables to display Pressure, Temperature, Motor status and Motor RPM into message number 25. The keystrokes are as follows:

With the 8720 powered down, plug the 8785 or equivalent keyboard into the 8720. Applying power will result in the following screen.

```
KEYBOARD ACTIVE
SELECT FUNCTION KEY
```

Press [F1].

```
***** ENTRY *****
ENTER #
```

Press [2] [5] [ENTER].

Type in the message as follows:

```
PRESS *30002* LBS M2 ^00010OFF:ON^ TEMP #.140003#[ALT][ALT]
RPM !30003!
```

Press [ENTER] to store message.

```
** MESSAGE STORED **
SELECT FUNCTION KEY
```

Press [F4].

```
***** DISPLAY *****
ENTER #
```

Press [2] [5] [ENTER].

```
PRESS XX LBS M2 XXX
TEMP XX.X° RPM XXXX
```

Pressure is in register address 30002.  
Motor M2 is monitored by output 00010.  
Temperature is in register address 40003.  
RPM is in register address 30003.

The message would appear as shown below: (values are arbitrary)

```
PRESS 50 LBS M2 OFF
TEMP 57.2° RPM 0000
```

**EXAMPLE 2:** An interesting use of the bit status defined function is to flash a word in a message. The programmed message may be as follows:

```
MOTOR 15 is ^00013:HOT^
```

Note that no label is defined for the 0 state. The word HOT will be displayed when coil 3 is in the 1 state. By using a flashing contact in series with the sensor contact, the monitored bit will turn on and off on a regular basis. The result is that the word HOT will alternately be displayed and go blank.

**EXAMPLE 3:** It is often helpful to display information to an operator graphically. In this example, the gallons of liquid being placed in the tank are displayed as a decimal value as well as a graph. The message is programmed as follows:

```
TANK VOLUME *40100* GALS @40100I@
```

If 40100 contains the decimal value 12, it will look like this on the display:

```
TANK VOLUME 12 GALS  
-----
```

If a the full scale value is greater than twenty, the data sent to the graph will have to be scaled. To scale, divide the full scale raw data value by 20 to obtain the scaling factor. The raw data is then divided by the scaling factor to obtain the data that is assigned to the delimiter register.

## 6.0 MESSAGE LOGGING

The 8720 can log messages out to a printer as they are triggered. Messages are sent out through the keyboard port, to an 8514 Message Logger Module. One end of the Message Logger Module plugs in to the keyboard port, the other end has a 25 pin D-subminiature connector. The 8514 has both a RS-232C serial port and a RS-422/485 port through its 25 pin connector, which allows for either a local or remote printer.

As each message is triggered, it is first sent to the logger before being sent to the display. If any variables are embedded within the message, the instantaneous value of the variable, at the time the message is triggered, is sent to the logger with the message. The message is sent to the logger only once every time it is triggered.

The printer baud rate and serial parameters for log mode, are defined through PRINTER SETUP (section 3.8). To use message logging, it must be enabled in the TRIGGER MODE CONFIGURE (section 3.7). An upper case R or B used when defining trigger mode, will set the 8720 to logging mode.

## 7.0 ERROR MESSAGES

The following is a list in alphabetical order of all messages that may appear on the 8720.

<u>MESSAGE</u>	<u>CAUSE</u>
CTS OFF, CHECK CABLE	Cable between processor is incorrect or disconnected. Processor is not powered up.
RAM READ / WRITE ERROR	RAM read / write workspace error. Recycle power, if problem continues replace unit.
ROM CHECKSUM ERROR	ROM checksum error. Recycle power, if problem continues replace unit.
READ / WRITE MEMORY ERROR	Read / write memory storage error. May occur during memory clear, retry if problem continues consult factory.
PLC NOT SUPPORTED	PLC not supported. Consult QUARTECH.
ERROR 5	Cycle test error. Consult QUARTECH.
ERROR 6	Cycle test communications failure. Consult QUARTECH.
PARITY ERROR	Parity error. Recycle power, if problem continues replace unit.
TRIGGER MODE NOT SET	Trigger mode not set. Set trigger mode (F7).
SERIAL PORT NOT CONFIGURED	Serial printer port has not been configured.
ILLEGAL ACKNOWLEDGE CONTACT ADDRESS	Illegal address location or address entered incorrectly.
ILLEGAL RESPONSE	Communications failure. Unit will automatically retry.
ILLEGAL VARIABLE IN MESSAGE #	Invalid variable address or incorrectly entered address.
INVALID TRIGGER ADDRESS	Trigger address is not a valid address or incorrectly entered. Change address.
LOAD ABORTED SELECT FUNCTION KEY	Will occur only during file transfer. Retry transfer.
NO MESSAGE #	Message # does not exist.
NO PC RESPONSE	Communications error. Incorrect or broken cable.
OPEN DELIMITER ERROR SELECT FUNCTION KEY	Variable entered incorrectly. Re-enter variable.

OPEN DELIMITER  
MESSAGE #

Variable end delimiter not found. Re-enter  
variable.

PC NOT RUNNING

Use programmer to place PLC into run mode.

EEPROM CHKSUM ERROR

Unit has encountered a checksum error within a  
message or a setup parameter. Power unit down  
and connect keyboard. Check all messages and  
setup parameters and correct any problems  
found. If problem cannot be found, the unit  
will have to be re-initialized (F10 key).

SYNTAX ERROR  
IN MESSAGE #

Variable entered incorrectly.

WAITING FOR MESSAGE

First message has not been received or message  
stack is empty.

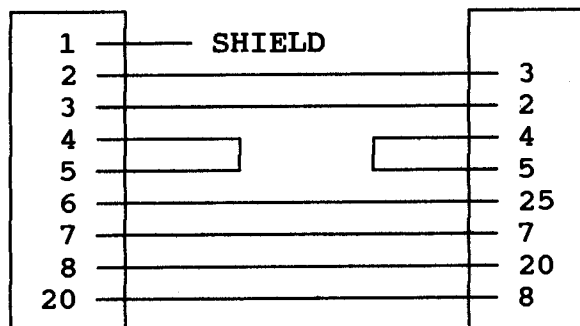
## 8.0 COMMUNICATION CABLE

Cable Type: Multi-Conductor shielded, 20 - 24 AWG

Example: Beldon No. 9538, Alpha No. 1218

Quartech cable 8566-10  
984-A/B, 884 PLC'S 8720 END

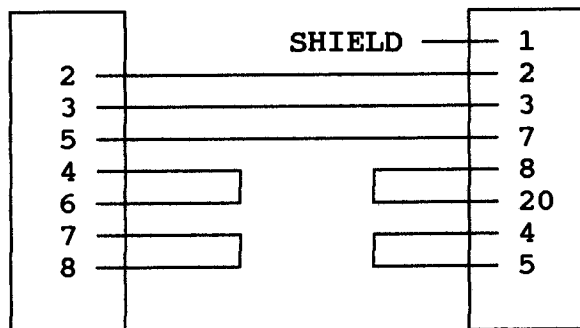
Male 25 pin  
D-type with  
slide latch  
hardware and  
Rt. angle hood



Female 25 pin  
D-type with  
screw lock  
hardware and  
straight hood

Quartech cable 8570-10  
984-680/480/380 PLC'S 8720 END

Male 9 pin  
D-type with  
screw lock  
hardware and  
45° hood



Female 25 pin  
D-type with  
screw lock  
hardware and  
straight hood

NOTE: If the 8720 is used with the 8517 multiplexer the cable from the  
8720 to the 8517 is the 8569-10 cable. A 2 foot cable from the 8517 to  
the PLC is supplied with the 8517. If a 584 PLC is used, contact  
Quartech for the cable that is used between the 8517 and the PLC.



## 9.0 ELECTRICAL REQUIREMENTS

Power to the DataMate 8720 is provided by an external AC source meeting the following requirements:

Supply Voltage: 120 VAC 50/60 HZ  
Supply Current: 1.0 AMP

## 10.0 INSTALLATION NOTES

### WIRING CONSIDERATIONS:

To minimize the effects of noise related system problems, care should be taken when routing the communication cable. Keep the cable at least 1 foot from 120 VAC lines, and at least 2 feet away from higher voltage lines. If the cable must cross AC power lines, cross them at right angles. If you route the cable through conduit, the conduit should contain only other communication cables or low voltage DC signals. Keep the cable away from sources of high energy fields such as arc welders, AC motors, motor starters, servo controllers, generators, induction heaters, and transformers.

### GENERAL:

Inspect the unit prior to installation for visible damage that may have occurred during shipment.

### NOTICE

ANY CLAIM FOR DAMAGE SHOULD BE FILED WITH THE CARRIER OR HIS AGENT.  
ALSO NOTIFY YOUR DISTRIBUTORS SO THAT CORRECTIVE ACTION CAN BE TAKEN.

### MOUNTING:

The 8720 is designed to be mounted in the door or on an operators console for ease of use. The attached template may be used to assist in the drilling and cutting of the mounting holes for the unit.

Care should be taken to protect the unit from metal chips and conductive particles. Failure to protect the unit may cause failure when power is applied and may void the warranty.

A minimum clearance of six inches should be kept between the unit and any other device that generates heat. In the event that the internal enclosure temperatures periodically exceeds 60 Deg. C (140 Deg. F), fans or a purge air system should be used to increase the air flow and eliminate "Hot Spots" that occur within the panel. Under extreme conditions, air conditioning of the panel may be required.

**ELECTRICAL:**

The 8720 is designed and constructed to minimize the effects of harsh industrial environments. However, when any control system operates under extreme conditions, additional measures must be taken. Additional steps may be required to suppress major sources of interference, should interference become a problem. Obvious sources are inductive items such as relays, motor starters, solenoids and "hard contacts". If suppression is used, suppressors should be placed as close to the item as possible for maximum effectiveness.

**CAUTION**

THE POWER TO THE 8720 MUST BE TURNED OFF WHILE CHANGING ANY CABLES OR WORKING ON ANY ELECTRICAL TERMINATION; TO PROTECT THE OPERATOR, EQUIPMENT AND THE UNIT FROM POSSIBLE HARM.

**11.0 ASCII CHARACTER SET**

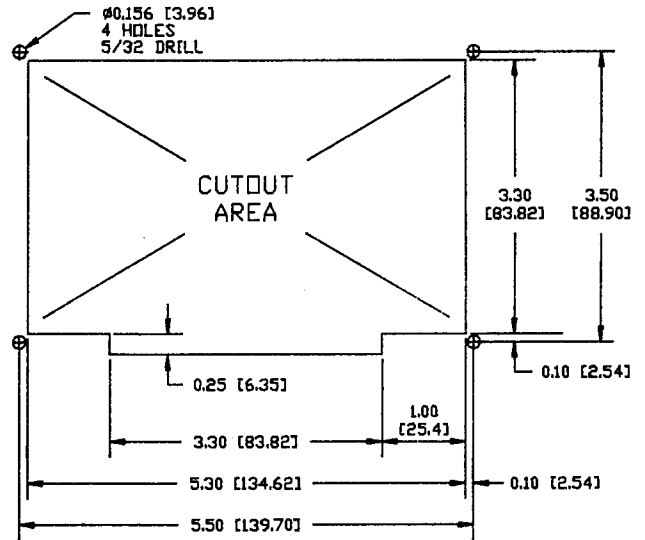
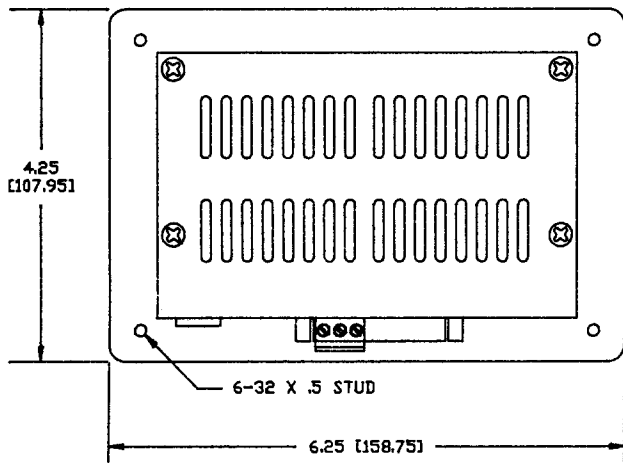
The following is the ASCII Character Set which is supported by the 8720. In general all keys that are available on a standard IBM type keyboard are available. The (°) symbol is obtained by pressing the [ALT] key twice.

**LOWER BYTE**

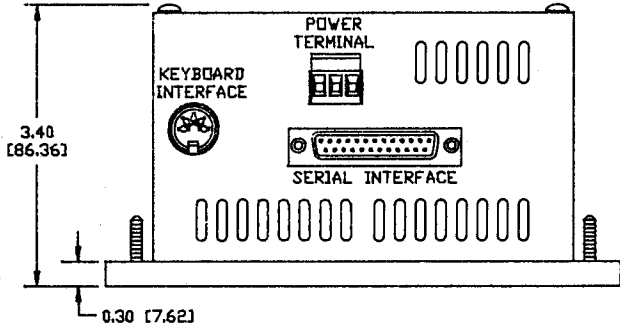
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
<b>U P P E R  B Y T E</b>	<b>2</b>	SPC	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
	<b>3</b>	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
	<b>4</b>	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	<b>5</b>	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
<b>6</b>	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
<b>7</b>	p	q	r	s	t	u	v	w	x	y	z	{		}	°		

# 12.0 MECHANICAL DIMENSIONS

REAR VIEW



BOTTOM VIEW



ALL DIMENSIONS IN INCHES [MILLIMETERS]