Dispalying Text From Your Program on a QuickPanel

Tech Note #37

Description

The QuickDesigner Text Display object can be used to display alphanumeric character strings on a QuickPanel. The text you display can be developed by your program or entered by an operator using an IBM PC-AT compatible keyboard connected to the ORION™ Keyboard Interface.

String data is transferred to the QuickPanel using the same register system used for transferring numeric data. The data is placed in a number of consecutive two byte integer registers. The ASCII codes for the character in the text are placed in the registers as shown below:

<table>
<thead>
<tr>
<th>Register Number</th>
<th>Text</th>
<th>ASCII Value</th>
<th>Register Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Y</td>
<td>89</td>
<td>22895</td>
</tr>
<tr>
<td></td>
<td>o</td>
<td>111</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>u</td>
<td>117</td>
<td>30066</td>
</tr>
<tr>
<td></td>
<td>r</td>
<td>114</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>32</td>
<td>8276</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>e</td>
<td>101</td>
<td>25976</td>
</tr>
<tr>
<td></td>
<td>x</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>t</td>
<td>116</td>
<td>29728</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32</td>
<td></td>
</tr>
</tbody>
</table>

The register value is calculated by multiplying the ASCII code for the first character in each pair by 256 then adding the ASCII code for the second character. You should note that using this technique you can display characters with ASCII codes from 32 to 126. This includes all numbers, upper and lower case letters and most punctuation marks.

The remainder of this tech note describes how set up and use a QuickPanel Text Object to display text from a MotionBASIC® program.
Creating a QuickPanel Text Object

Using QuickDesigner with the panel on which you want to display text open, click on the Text Object toolbox icon and then place it on your panel.

After placing the Text Display on a panel, the Text Display Settings dialog box is displayed. The tag variable name defines the first register in the group of registers that will hold the string data. The MS (Message String) tag variable type designates it as string data (see below).

Clicking on the Text Format button displays the Text Format Settings dialog box shown below, which is used to configure string text display format. The Length parameter specifies the maximum number of characters to be displayed in the Text Display, this would normally be configured to be the same as the number of characters specified by the Tag Attributes Length parameter (explained later in this Tech Note). The maximum value for this number will depend on the size you make the text object and the Text Size you select.

Tag Attributes

To configure the number of characters read from the ORION, close the QuickDesigner panel editor and open the Tag Manager by clicking on the Tag button in QuickManager. Double
clicking on the Tag variable created for use with the Text Display opens the Tag Attribute dialog box shown below. The maximum Size is 127, which is equivalent to 64 mapped registers in the ORION (127 ÷ 2 rounded up to the nearest whole number). Since you will need to map one register for every two characters, it is more convenient if you make the size an even number.

The Size parameter is used to specify the number of characters to be read from the ORION. This number would be the same as the Length parameter specified in the Text Format dialog box for the Text Display. If the Tag Attribute Size is larger than the Text Display Length, more characters will be read from the ORION than will be displayed on the QuickPanel. This can waste processor time by requiring the system to communicate more registers than it will actually use.

**MotionBASIC Program**

The following MotionBASIC® does the following:

a. QuickPanel device  
b. Loads a QuickPanel file  
c. Maps registers  
d. Selects the panel with the text object  
e. Displays some text

```motionbasic
PROGRAM.START:
QP.CLOSE                 'close QuickPanel communications
QP.OPEN 1,"SRL1:"        're-open communication
QP@=ON                   'enable communications
QP.LED@=2                'flash LED on message receipt
WAIT 500                 'wait to make sure the QuickPanel is awake
QP.DNLQ "QPSTRING.QPL",0 'download the QuickPanel file
ERASE QPTEXT :DIM QPTEXT(18)  'create the register array
MAP ERASE                'erase any existing maps
FOR I=1 TO 18            'map 18 registers for the text
```
MAP I TO QPTEXT(I) ' the number 18 is equal to the
  the number 18 is equal to the
NEXT I ' Tag size attribute divided by 2

MAP 99 TO PANEL 'map a variable to select which panel to display. This
  register should be designated as the Panel Trigger Tag in
  the QuickPanel setup
PANEL=1 'display panel # 1

QP.MSG$="Your Text" 'This is the text we are going to display

START.REG=1 'this is the first register for the text object
NUM.REG=18 'this is the number of registers. it corresponds to
  the tag size attribute divided by 2
QP.PRINT 'send the message

WHILE TRUE :WEND 'Since we are done, loop here forever
END

'----- Displays text contained in QP.MSG$ on the QuickPanel in the text object
  mapped to the register number contained in START.REG. Thre the number
  of registers mapped to the text object should be contained in NUM.REG

QP.PRINT:
  TMP.MSG$ =MID$(QP.MSG$,1,2*NUM.REG) 'truncate string as necessary

  '----- 'pad string with spaces if necessary
  ACTUAL.LEN =LEN(TMP.MSG$)
  IF ACTUAL.LEN <(2*NUM.REG) THEN
    TMP.MSG$ =TMP.MSG$ +SPACE$(2*NUM.REG -ACTUAL.LEN)
  ENDIF

  '----- parse the string into 2 character segments
  END.REG =START.REG +NUM.REG -1 'calculate the last register in the block
  FOR I =START.REG TO END.REG 'loop through all the registers
    CHAR =2*(I-START.REG)+1 'this sets the character we are working on
    C1=ASC(MID$(TMP.MSG$,CHAR,1)) 'get the ASCII code of the first character
    IF C1<32 OR C1>127 THEN C1=32 'if it is non-printable, set it to a space
    C2=ASC(MID$(TMP.MSG$,CHAR+1,1)) 'get the ASCII code of the next character
    IF C2<32 OR C2>126 THEN C2=32 'if it is non-printable, set it to a space
    REG(I) =C1*256 +C2 'put them in the high and low bytes
  NEXT I
RETURN