Driving LCD with i2c interface

This module is part of the original MMBasic library. It is reproduced here with kind permission of Hugh Buckle and Geoff Graham. Be aware it may reference functionality which has changed or is deprecated in the latest versions of MMBasic.

Nice project to drive a standard HD4478 based LCD from an I^{2}C port expander.

HD4478, PCF8574

I2CLCD.bas

```
'
' http://letsmakerobots.com/node/4240 provided PIcAxe code
'
' Maximite modifications by John Gerrard
'
'  +-------------------------------------------------------------+
'  |                                                             |
'  | LCD DISPLAY    Hitachi HD44780 Standard                     |
'  |                                                             |
'  | 1   2   3   4   5   6   7   8   9   10  11  12  13  14 15  16 |
'  +-+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+-+
'    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
'   GND V+  CNT  RS  RW  E   D0  D1  D2  D3  D4  D5  D6  D7  LV+ LGND
'    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
'    =   +5  |   |   =   |                   |   |   |   |
'    |   +-+ |   |       |                   |   |   |   |
'    |     < |   |       |                   |   |   |   |
'    | 4k7 ><+   +-------------------+       |   |   |   |
'    |     <             |           |       |   |   |   |
'    +-----+             |           |       |   |   |   |
'                        |           |       |   |   |   |
'                        |           |       |   |   |   |
'                        |           |       |   |   |   |
'                        |           |       |   |   |   |
'                        |           |       |   |   |   |
'     to Maximite I2C    |           |       |   |   |   |
'           ^   ^        |           |       |   |   |   |
'       +3.3|   |       ++           |       |   |   |   |
'       |   |   |   |   |   |   |   |       |   |   |   |
'     +-+---+---+---+---+---+---+---+-+     |   |   |   |
'     | V+ SDA SCD INT  P7  P6  P5  P4|     |   |   |   |
'     |                               |     |   |   |   |
'      D    PCF8574 port expander     |     |   |   |   |
'     |                               |     |   |   |   |
'     | A0  A1  A2  P0  P1  P2  P3 GND|     |   |   |   |
'     +-+---+---+---+---+---+---+---+-+     |   |   |   |
'       |   |   |   |   |   |   |   |       |   |   |   |
'       +---+---+   |   |   |   +-----------------------+
'       |           |   |   +-----------------------+
'
```

```
'      gnd          |   +------------------------+
'                   +------------------------+
'
'



I2CEN 100,100 ' Enable I2C
I2CAddr  = &H20        ' this is the 8574 I2C address
                       ' A2=A1=A0=0 <-> x100 000x

'Name      8574 bit     LCD
'----      --------     ---
DB4       = 0          ' LCD Data Line 4 (pin 11)
DB5       = 1          ' LCD Data Line 5 (pin 12)
DB6       = 2          ' LCD Data Line 6 (pin 13)
DB7       = 3          ' LCD Data Line 7 (pin 14)
RS        = 4          ' 0 = Command   1 = Data (pin 4)
                       ' 5  free (to pin 15 for lcd bk light, for ex.)
                       ' 6  free
E         = 7          ' 0 = Idle      1 = Active (pin 6)

RSCMDmask = &B00000000 ' Select Command register
RSDATmask = &B00010000 ' Select Data register = High P4 on 8574
Emask     = &B11100000 ' Enable = P7 on 8574

Dim CNT(6)

CNT(0) = &H33           ' %0011---- %0011----   8-bit / 8-bit
CNT(1) = &H32           ' %0011---- %0010----   8-bit / 4-bit

' Byte commands - To configure the LCD

                        '
                        ' Display Format
                        ' 4bit mode, 2 lines, 5x7
                        '
                        '  001LNF00
CNT(2) = &B00101000     ' %00101000
                        ' L : 0 = 4-bit Mode    1 = 8-bit Mode
                        ' N : 0 = 1 Line        1 = 2 Lines
                        ' F : 0 = 5x7 Pixels    1 = N/A

                        '
                        ' Setup Display
                        ' Display ON, Cursor On, Cursor Steady
                        '
                        '  00001DCB
CNT(3) = &B00001100     ' %00001110
                        ' D : 0 = Display Off   1 = Display On
```

```
                        ' C : 0 = Cursor Off    1 = Cursor On
                        ' B : 0 = Cursor Steady 1 = Cursor Flash

                        '
                        ' Setup Cursor/Display
                        ' Inc Cursor Cursor Move
                        '
                        '  000001IS
CNT(4) =  &B00000110    ' %000001IS   Cursor Move
                        ' I : 0 = Dec Cursor    1 = Inc Cursor
                        ' S : 0 = Cursor Move   1 = Display Shift

CNT(5) =  &B00000001    ' Clear Screen


GoSub InitialiseLcd     ' Initialise the LCD


Menu:
line1$ = "2x20 LCD Test"       ' Add your text here
line2$ = "2nd Line"          ' And here

Gosub Line1            ' or gosub LCD_Mainloop if you want clear the screen

' Add key press stuff here

Goto menu


' I2C LCD

LCD_Mainloop:

    aByte = CNT(5)         'Clear Screen
    GoSub SendCmdByte

Line1:

aByte = &B00000010         ' Put cursor at start of Line 1
GoSub SendCmdByte

 For i = 1 To Len(Line1$)
    aByte = Asc(Mid$(Line1$, i, 1))
    GoSub SendDataByte
  Next i

If line2$ = "" Then Return
EndIf

Line2:
    aByte = &H80 Or &H40        ' Put cursor at start of Line 2
```

```
    GoSub SendCmdByte

 For i = 1 To Len(Line2$)
    aByte = Asc(Mid$(Line2$, i, 1))
    GoSub SendDataByte
  Next i

Return


'  INITIALIZE LCD
' ------------------------------------------------------------------
'
InitialiseLcd:

    For index = 0 To 5
      aByte =  CNT(index)
      GoSub SendInitCmdByte
    Next

Return

' SEND INIT CMD BYTE - SEND CMD BYTE - SEND DATA BYTE
' ------------------------------------------------------------------
'
SendInitCmdByte:

SendCmdByte:

    rsbit = RSCMDmask                ' Send to Command register

SendDataByte:

    '
    ' put MSB OUT 1st
    '
    temp = (aByte \ &B10000) Or rsbit
    GoSub DirectSendCmd
    '
    ' put LSB
    '
    temp = aByte And &H0F Or rsbit
    rsbit = RSDATmask               ' Send to Data register next

DirectSendCmd:
    temp = temp Xor Emask            ' E=1

I2CSEND i2caddr, 0, 1, temp          ' send to 8574
Pause 2
```

```
   temp = temp Xor Emask              ' E=0

I2CSEND i2caddr, 0, 1, temp
Return
```