Finally got around to figuring out how to paste code here using [SyntaxHighlighter Evolved](http://wordpress.org/extend/plugins/syntaxhighlighter/) plugin will test it out with the Nokia 84×48 LCD Code still need to look into an auto copy feature.

[?](http://www.raspians.com/testing-source-code-plugin-for-the-site/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136 | #!/usr/bin/python  # -\*- coding: utf-8 -\*-  # using wiringPI GPIO method  # also using PWM on gpio pin 12  # now using wiringpi and not BCM ID method  # this should work with both v1 and v2 Pi boards    import wiringpi    import time    #gpio's :  SCE = 2 # gpio pin 13 = wiringpi no. 2  SCLK = 5 # gpio pin 18 = wiringpi no. 5  DIN =  4 # gpio pin 16 = wiringpi no. 4  DC =  3 # gpio pin 15 = wiringpi no. 3  RST =  0 # gpio pin 11 = wiringpi no. 0  LED = 1 # gpio pin 12 = wiringpi no. 1    font =[  0x7E, 0x11, 0x11, 0x11, 0x7E, # A  0x7F, 0x49, 0x49, 0x49, 0x36, # B  0x3E, 0x41, 0x41, 0x41, 0x22, # C  0x7F, 0x41, 0x41, 0x22, 0x1C, # D  0x7F, 0x49, 0x49, 0x49, 0x41, # E  0x7F, 0x09, 0x09, 0x09, 0x01, # F  0x3E, 0x41, 0x49, 0x49, 0x7A, # G  0x7F, 0x08, 0x08, 0x08, 0x7F, # H  0x00, 0x41, 0x7F, 0x41, 0x00, # I  0x20, 0x40, 0x41, 0x3F, 0x01, # J  0x7F, 0x08, 0x14, 0x22, 0x41, # K  0x7F, 0x40, 0x40, 0x40, 0x40, # L  0x7F, 0x02, 0x0C, 0x02, 0x7F, # M  0x7F, 0x04, 0x08, 0x10, 0x7F, # N  0x3E, 0x41, 0x41, 0x41, 0x3E, # O  0x7F, 0x09, 0x09, 0x09, 0x06, # P  0x3E, 0x41, 0x51, 0x21, 0x5E, # Q  0x7F, 0x09, 0x19, 0x29, 0x46, # R  0x46, 0x49, 0x49, 0x49, 0x31, # S  0x01, 0x01, 0x7F, 0x01, 0x01, # T  0x3F, 0x40, 0x40, 0x40, 0x3F, # U  0x1F, 0x20, 0x40, 0x20, 0x1F, # V  0x3F, 0x40, 0x38, 0x40, 0x3F, # W  0x63, 0x14, 0x08, 0x14, 0x63, # X  0x07, 0x08, 0x70, 0x08, 0x07, # Y  0x61, 0x51, 0x49, 0x45, 0x43, # Z  ]    def main():    start = time.time()    begin(0xa4) # contrast - may need tweaking for each display ( b3 )    gotoxy(28,0)    text("TAKE A")    gotoxy(18,2)    text("RASPBERRY")    gotoxy(26,4)    text("PI BYTE!")    finish = time.time()    print ("Total time : ",finish - start)    print    def gotoxy(x,y):    lcd\_cmd(x+128)    lcd\_cmd(y+64)    def text(words):    for i in range(len(words)):      display\_char(words[i])    def display\_char(char):    index=(ord(char)-65)\*5    if ord(char) >=65 and ord(char) <=90:      for i in range(5):        lcd\_data(font[index+i])      lcd\_data(0) # space inbetween characters    elif ord(char)==32:        lcd\_data(0)        lcd\_data(0)        lcd\_data(0)        lcd\_data(0)        lcd\_data(0)        lcd\_data(0)    def cls():    gotoxy(0,0)    for i in range(84):      for j in range(6):        lcd\_data(0)    def setup():    # set pin directions    wiringpi.wiringPiSetup()    wiringpi.pinMode(SCE, 1) # output    wiringpi.pinMode(DIN, 1) # output    wiringpi.pinMode(SCLK, 1) # output    wiringpi.pinMode(DC, 1) # output    wiringpi.pinMode(RST, 1) # output    wiringpi.pinMode(LED, 2) # LED set up as PWM    print ("setup ok")    wiringpi.pwmWrite(LED, 255)    def begin(contrast):    setup()    # toggle RST low to reset    wiringpi.digitalWrite(SCE, 0)    wiringpi.digitalWrite(RST, 0)    wiringpi.digitalWrite(LED, 0)    time.sleep(0.100)    wiringpi.digitalWrite(RST, 1)    lcd\_cmd(0x21) # extended mode    lcd\_cmd(0x14) # bias    lcd\_cmd(contrast) # vop    lcd\_cmd(0x20) # basic mode    lcd\_cmd(0xc) # non-inverted display    cls()    def SPI(c):    # data = DIN    # clock = SCLK    # MSB first    # value = c    for i in xrange(8):      wiringpi.digitalWrite(DIN,((c & (1 << (7-i))) > 0))      wiringpi.digitalWrite(SCLK, 1)      wiringpi.digitalWrite(SCLK, 0)    def lcd\_cmd(c):    wiringpi.digitalWrite(DC, 0)    SPI(c)    def lcd\_data(c):    wiringpi.digitalWrite(DC, 1)    SPI(c)    if \_\_name\_\_ == "\_\_main\_\_":    main() |