

'-----Title-----'

' File.....4331\_encoder1.pbp  
' Started....10/28/09

' Microcontroller Used: Microchip Technology 18F4331  
' Available at:  
' <http://www.microchipdirect.com/ProductDetails.aspx?Category=PIC18F4331>  
' or <http://www.digikey.com/>  
' Motor Controller Used: Xavien 2 Motor Driver "XDDCMD-1  
' Available at: [http://encodergeek.com/Xavien\\_Amplifier.html](http://encodergeek.com/Xavien_Amplifier.html)  
' Motor and Encoder Used: Small Motor with Quadrature Incremental Encoder  
' Available at: [http://encodergeek.com/DCMtr\\_SMALL.html](http://encodergeek.com/DCMtr_SMALL.html)  
'  
' PicBasic Pro Code: micro-Engineering Labs, Inc.  
' [melabs.com](http://melabs.com)

'-----Program Description-----'

' Program uses encoder to track position of motor.  
' Display also gives motor power.

'---Review PicBasic Pro Command---

' The PicBasic Pro Compiler Manual is on line at:  
' <http://www.microengineeringlabs.com/resources/index.htm#Manuals>  
'  
' HPWM Channel,Dutycycle,Frequency  
'  
' Outputs a PWM signal using the PICs hardware which  
' is available on some PICs including the PIC18G4331.  
' Channel specifies which PWM channel to use.  
' Dutycycle ranges from 0 (0%) to 255 (100%).  
' Frequency - lowest frequency depends upon oscillator speed,  
' highest frequency at any oscillator speed is 32,767 Hz.  
' Look around page 75 in the PicBasic Pro Compiler Manual  
' for detailed discussion of the HPWM command.

'-----PIC Connections-----'

18F4331 Pin	Wiring
RA0(AN0)	Potentiometer, controls motor power
RA3	Signal 1 from Encoder
RA4	Signal 2 from Encoder
RB5	In Circuit Serial Programming (ICSP) PGM 100K Resistor to GND
RB6	ICSP PGC (Clock)
RB7	ICSP PGD (Data)
RC0	Brake Motor 1 on Xavien XDDCMD-1 (Pin 1)
RC1	PWM Motor 1 on Xavien XDDCMD-1 (Pin 2)
RC3	Direction Motor 1 on Xavien XDDCMD-1 (Pin 3)
RD4	LCD Data Bit 4
RD5	LCD Data Bit 5
RD6	LCD Data Bit 6

```

'          RD7          LCD Data Bit 7
'          RE0          LCD Register Select
'          RE1          LCD Enable
'          MCLR         4.7K Resistor to +5V & ICSP Vpp
'          VDD          +5V
'          VSS          GND
'          OSC1 & OSC2  4 MHz Crystal w/ 2-22 pF Cap. to GND

'-----Xavien XDDCMD-1 Connections-----

'  Xavien 2x5 Header Pin      Wiring      Pin Layout 2x5 Header
'  -----
'
'          Pin 1 Motor 1 Brake      RC0          0 0 0 0 0
'          Pin 2 Motor 1 PWM        RC1          0 0 0 0 0
'          Pin 3 Motor 1 Direction  RC3          1 3 5 7 9

' See schematic at:
' http://cornerstonerobotics.org/schematics/18f4331\_hpwm\_motor\_encoder.pdf

'--Sample POSCNTH, POSCNTL Values and Corresponding Position Counter--

'  position = 256 * POSCNTH + POSCNTL

'  POSCNTH      POSCNTL      Position Counter
'  -----
'  0            0            0
'  0            1            1
'  1            0            255
'  0            128          128
'  128          0            32768
'  0            255          255
'  255          0            65280
'  255          255          65535

'-----Defines-----

DEFINE LCD_DREG PORTD      ' Set LCD Data port
DEFINE LCD_DBIT 4          ' Set starting Data bit to 4
DEFINE LCD_BITS 4         ' Set LCD bus size to 4
DEFINE LCD_RSREG PORTE    ' Set LCD Register Select port to E
DEFINE LCD_RSBIT 0        ' Set LCD Register Select bit to 0
DEFINE LCD_EREG PORTE     ' Set LCD Enable port to E
DEFINE LCD_EBIT 1         ' Set LCD Enable bit to 1
DEFINE LCD_LINES 2       ' Set number of lines on LCD to 2
DEFINE LCD_COMMANDUS 2000 ' Set command delay time to 2000 us
DEFINE LCD_DATAUS 50      ' Set data delay time to 50 us
DEFINE ADC_BITS 8         ' Set number of bits in result to 8
DEFINE ADC_CLOCK 3        ' Set clock source (rc = 3)
DEFINE ADC_SAMPLEUS 50    ' Set sampling time in us
DEFINE CCP2_REG PORTC     ' Set HPWM Channel 2 port to C
DEFINE CCP2_BIT 1         ' Set HPWM Channel 2 bit to 1

'-----Variables-----

```

```

mot_pwr  VAR    BYTE    ' Declare mot_pwr variable, reserve byte
pot_val  VAR    BYTE    ' Declare pot_val, reserve byte
position VAR    WORD    ' Declare position, reserve word

'-----Initialization-----

CCP1CON = %00111111    ' Set Capture/Compare/PWM Module Control
                        ' Register CCP1CON in PWM mode (bits 0-3),
                        ' bits 4,5 set LSBs of 10-bit duty cycle,
                        ' see 18F4331 datasheet page 151 +/--.
ANSEL0 = %00000001    ' Set AN0 to analog, AN1-AN7 to digital,
                        ' see datasheet page 249 +/--.
ANSEL1 = %00000000    ' Set AN8 to digital, see datasheet
                        ' page 249 +/--.
TRISA = %00011111     ' Set TRISA register, RA7-RA5 as outputs,
                        ' RA4-RA0 as inputs, see datasheet
                        ' page 107 +/--.
LATA = %00000000      ' Set all LATA register bits to 0.
TRISB = %00000000     ' Set RB7-RB0 pins in PORTB as outputs.
TRISC = %00000000     ' Set RC7-RC0 pins in PORTC as outputs.
QEICON = %10001000    ' Set Quadrature Encoder Interface Control
                        ' Register. See page 171 +/- for
                        ' encoder set up.
PORTC.0 = 1           ' Turn on brake.
PORTC.1 = 0           ' Set PWM bit for Channel 2 of HPWM to LOW.

'-----Main Code-----

    PAUSE 1000         ' Pause to start up LCD
    PORTC.0 = 0        ' Turn off brake
    PORTC.3 = 0        ' Set direction of motor
                        ' If position value on LCD is in the 65,
000s
                        ' and counting down, then change the
                        ' motor direction: PORTC.3 = 1.
' Set counter starting position:

    POSCNTH = 0        ' Set counter for encoder, H bit
    POSCNTL = 0        ' Set counter for encoder, L bit
                        ' With POSCNTH and POSCNTL = 0,
                        ' position counter will start at 0.
                        ' See table above for more sample values.

start:

    position = 256*POSCNTH + POSCNTL ' Read position
    ADCIN 0, pot_val    ' Read AN0 and store result in pot_val.
                        ' This potentiometer (connected to AN0)
                        ' sets the motor power.

    mot_pwr = 11*pot_val/16 + 77
                        ' mot_pwr = 11/16 * pot_val + 77
                        ' (Can't write equation as 11/16 * pot_val
                        ' since interger division truncates: any
                        ' fractional part is discarded. Since 11
                        ' and 16 are integers, 11/16 would be
                        ' truncated to zero.)

```

```

                                ' 77 is the minimum power to start motor.
                                ' 11/16 is the slope of the line to give
                                ' mot_pwr values from 77 to about 255.
                                ' See graph & equation in schematic.
HPWM 2, mot_pwr, 20000          ' Send PWM signal from RC1 to Pin 2 on
                                ' the Xavien XDDCMD-1 DC motor driver.

GOSUB lcd                      ' Go to lcd subroutine
GOTO start                     ' Go to loop label
END

lcd:

LCDOUT $FE, $80, "mot_pwr = ",DEC3 mot_pwr,"    "
                                ' Display motor power on first line
                                ' in 3 decimal digits.
LCDOUT $FE, $C0, "position = ",DEC5 position
                                ' Display position on second line
                                ' Position will count to 65535, then
                                ' cycle back to 0 and continue counting.
                                ' in 5 decimal digits.
RETURN                          ' Go back to main routine that called us.
```