'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* www.ideastechnology.com \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

'

' \* Nombre proyecto:

' SISTEMA DE ADQUISICION Y ALMACENAMIENTO DE DATOS CON MATLAB

' \* Descripcion:

' Este proyecto consiste en leer un dato analogico proveniente de un potenciometro

' para controlar el nivel de PWM que le aplicamos al motor por medio del puente H

' la señal digital que proviene del encoder optico es procesada y tranformada

' a revoluciones por minuto (RPM) y posteriormente enviada por comunicacion serial

' RS232 al computador donde MATLAB se encargara de tomar los datos de (PWM y RPM).

' \* Configuraciones:

' Microcontrolador: PIC16F886

' Oscillator: INT\_RC\_OSC\_NOCLKOUT, 4MHz

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

'\*\*\*\*\* Nombre proyecto \*\*\*\*\*

program Adquisicion

'\*\*\*\*\* Declaración de Variables \*\*\*\*\*

dim mtempin,mtempout as word

dim tempin,tempout,n,receive as byte

dim txt as string[4]

dim recivio as byte

dim velocidad as word

dim automatico as byte

dim salto as byte

dim x\_coord128, x\_coord as word

''unsigned int GetY() {

''//reading Y

'PORTC.F0 = 0; '// DRIVEA = 0 (LEFT drive off , RIGHT drive off , TOP drive on)

'PORTC.F1 = 1; '// DRIVEB = 1 (BOTTOM drive on)

'Delay\_ms(5);

'return ADC\_read(2); '// reading Y value from RA1 (from LEFT)

'y\_coord64 = 64 -((y\_coord \*64) / 1024);

''}

'\*\*\*\*\* Programa principal main \*\*\*\*\*

'-----------------------------------------

sub procedure interrupt ' Interrupt service routine

if (UART1\_Data\_Ready() = 1) then

porta.2=1

delay\_ms(300)

porta.2=0

delay\_ms(300)

receive = UART1\_Read()

velocidad=receive

'select case salto

' case 1

' if receive = 0xA0 then

' salto =2

' else

' salto =1

' end if

' case 2

' salto =3

' velocidad= (receive\*254)/9

'' portA=receive

' recivio = 1

' case 3

' salto =1

' automatico = receive ' automatico = 1 y no automatico = 0

'

' end select

end if

end sub

'-\*---------------------------------------

main:

OPTION\_REG=%10000000

INTCON = %11000000 ' Enable external interrupts bit(7) and bit(6) de permiso q no se controlan con INTCON

'T1CON = $30

'PIR1.TMR1IF = 0 ' clear TMR1IF

'TMR1H = $0B ' initialize Timer1 register

'TMR1L = $DB

PIE1 = %00100000 ' enable Timer1 interrupt

' Configuracion de Puertos analogicos / digitales AN0 - AN13

ANSEL = %00000111 'REGISTRO CONFIG. O DIGITAL, 1 ANALOGICO

ANSELH = %00000000 'REGISTRO CONFIG. O DIGITAL, 1 ANALOGICO

' Configuracion de Puertos como entradas / salidas

trisa = %00000111 ' Configuracion IN/OUT PORTA

trisb = %00000000 ' Configuracion IN/OUT PORTB

trisc = %10000000 ' Configuracion IN/OUT PORTC

' Reset de salidas

PORTA = %00000000 ' Inicializacion PORTA

PORTC = %00000000 ' Inicializacion PORTC

'USART

Uart1\_Init(9600) ' Initialize USART module

' Lazo infinito Bucle While

PWM1\_Init(1000) ' Initialize PWM1 module at 5KHz

PWM2\_Init(1000) ' Initialize PWM2 module at 5KHz

PWM1\_Start() ' start PWM1

PWM2\_Start() ' start PWM2

PWM1\_Set\_Duty(0)

PWM2\_Set\_Duty(0)

while true

velocidad=ADC\_read(0)>>2

' unsigned int GetX() {

'//reading X

PORTB.0 = 1'; ' DRIVEA = 1 (LEFT drive on, RIGHT drive on, TOP drive off )

PORTB.1 = 0'; ' DRIVEB = 0 (BOTTOM drive off )

Delay\_ms(5)';

'; ' reading X value from RA0 (BOTTOM)

x\_coord128 = ADC\_read(1) / 10';

porta=velocidad

if x\_coord128 > 5 then

if x\_coord128 > 40 then '35 + 5

portb.1=1

velocidad=(x\_coord128-40)\*3

PWM1\_Set\_Duty(velocidad)

portb.2=1

PWM2\_Set\_Duty(velocidad)

else

if x\_coord128 < 36 then '35-5

portb.1=0

velocidad=(36-x\_coord128)\*3

PWM1\_Set\_Duty(velocidad)

portb.2=0

PWM2\_Set\_Duty(velocidad)

else

PWM1\_Set\_Duty(0)

PWM2\_Set\_Duty(0)

end if

end if

else

PWM1\_Set\_Duty(0)

PWM2\_Set\_Duty(0)

end if

UART1\_Write\_text("\*")

WordToStr(x\_coord128, txt)

UART1\_Write\_text(txt)

wend

end.



