PROYECTO: ASISTENTE PARA NO VIDENTES

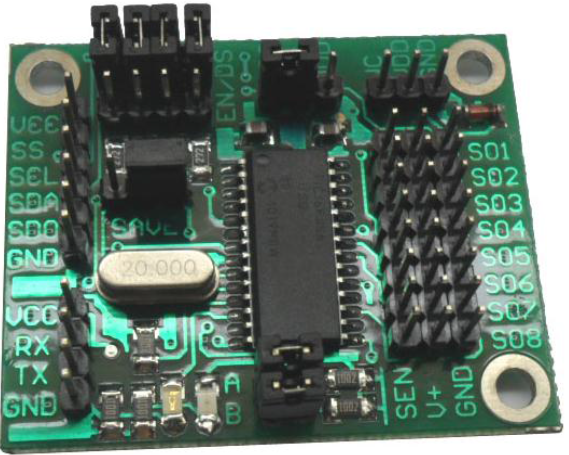
 |

PROYECTO DE TESIS

2014

**IMPLEMENTACION TECNICA**

1. **MATERIALES UTILIZADOS**
   1. **TARJETA DE CONTROL CON PIC 16F886 TIPO SMD**



ESPECIFICACIONES

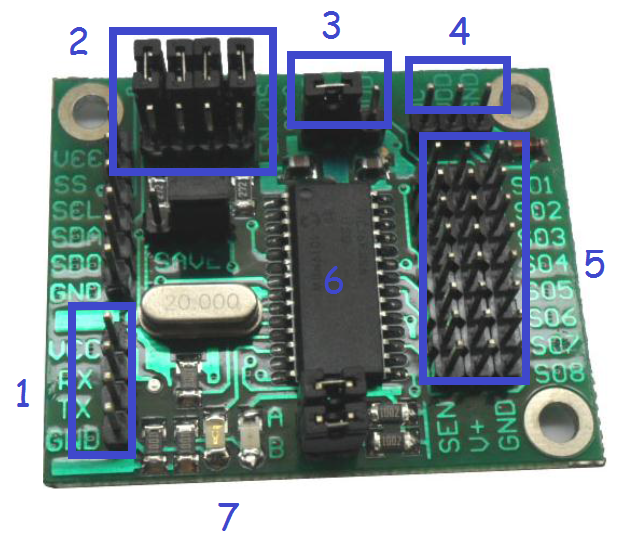
* Comunicación por protocolo UART con antena Xbee y antena Bluetooth.
* 24 entradas y salidas digitales para botoneras.
* 1 led indicador de presencia de voltaje.
* Programación de PIC por protocolo ICSP.
* Conectores de voltaje de alimentación desde 4.5v hasta 5.2v
* Fuente de alimentación de corriente continua EXT.

CONECTOR DE PROGRAMACION

La mayoría de los microcontroladores usan el protocolo ICSP para programar el código desarrollado dentro del espacio de memoria las funciones a ejecutar con los periféricos conectados.

El código del programa puede ser desarrollado en cualquier plataforma orientada a programación de microcontroladores.

DIAGRAMA DE PINES



1. Conexión con antena Xbee y antena Bluetooth
2. Switch de resistencias para botoneras máximo 8.
3. Selector de alimentación de voltaje externo.
4. Conector de alimentación de voltaje externo
5. Banco de botoneras.
6. Microcontrolador 16f886
7. Led de encendido y alimentación de voltaje DC.
   1. **SENSORES DE DISTANCIA ULTRASONICO SR04**



Este módulo es capaz de entregar información de la distancia que existe entre el mismo y un objeto ubicado dentro del rango de 2cm hasta 400cm, los ciclos repetitivos de programación dentro del controlador permiten obtener un dato mucho más preciso de longitud con unidad de media que es el centímetro regulado por el Sistema Internacional de Medidas.

Se alimenta de 5 voltios de corriente continua, la señal de control es de nivel TTL y es capaz de entregar medidas en un rango 2cm<X<400cm.

El consumo máximo de energía es de 20mA concluyendo un promedio

de 100mW de potencia absorbida.

* 1. **ANTENA DE BLUETOOTH HC06**



Los datos de distancia obtenidos en el microcontrolador son enviados por Protocolo Serial UART a través del pin de TX, dependiendo del botón presionado desde el mando inalámbrico colocado en la mano del no vidente.

Se alimenta de 5 voltios de corriente continua, la señal de control es de nivel TTL y es capaz de mantener la conexión con el teléfono hasta 100 metros de distancia (línea de vista) por ser de un dispositivo de clase 1.

El consumo máximo de energía es de 80mA durante el enlace de datos y

40 mA de consumo de corriente con enlace estable y sin transmisión.

Esta determinado para este dispositivo un consumo de 400mW de potencia consumida durante enlace de datos y 200mW de potencia absorbida sin envío de datos manteniendo la comunicación.

* 1. **ANTENA XBEE PRO**



Un evento que implica presionar un botón de control, será receptado por el controlador para enviar un comando por el PIN TX hasta el circuito de control receptor por el PIN RX, cada comando posee un identificador para cada botonera programada previamente para evitar mal interpretación de datos.

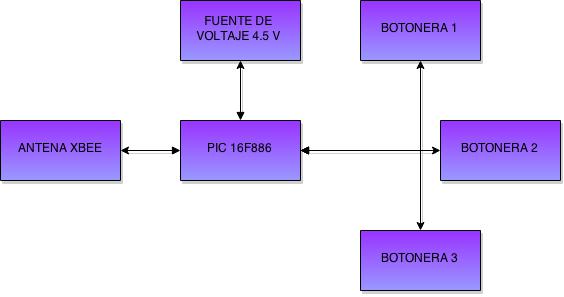
El circuito que contiene las botoneras se mantendrá como MASTER, únicamente se dedicará al envío de datos por Xbee, el circuito que contiene la antena de Xbee en el PIN RX, los 3 sensores de distancia y la antena de Bluetooth en el PIN TX se mantendrá como ESCLAVO, únicamente se dedicará al envío de datos controlado por el circuito MASTER.

Se alimenta de 3.3 voltios de corriente continua, la señal de control es de nivel LV-TTL.

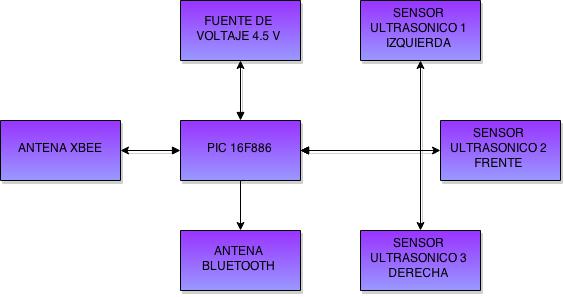
Consume de 10mW hasta 50mW de potencia dependiendo del área en donde se esté utilizando.

La tasa de trasmisión de bits (BAUDRATE) es de 9600 para este proyecto configuración preestablecida desde fábrica tanto para la antena de Bluetooth, la antena de Xbee y las tarjetas de control, puesto que de no ser sincronizadas con este tipo de parámetro la comunicación no fuese posible.

1. **DIAGRAMA ESQUEMÁTICO CIRCUITO MASTER.**

****

1. **DIAGRAMA ESQUEMÁTICO CIRCUITO ESCLAVO.**

****

1. **TABLA DE INTERCONEXION CIRCUITO MASTER.**

|  |  |
| --- | --- |
| BOTONERA | MICROCONTROLADOR |
| PIN RB0 | PIN RA0 |
| PIN RB1 | PIN RA1 |
| PIN RB2 | PIN RA2 |

|  |  |
| --- | --- |
| ANTENA XBEE | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| PIN RX | PIN TX |

1. **TABLA DE INTERCONEXION CIRCUITO ESCLAVO.**

|  |  |
| --- | --- |
| SENSOR DISTANCIA 1 | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| TRIG | PIN RB0 |
| ECHO | PIN RB1 |

|  |  |
| --- | --- |
| SENSOR DISTANCIA 2 | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| TRIG | PIN RA3 |
| ECHO | PIN RA4 |

|  |  |
| --- | --- |
| SENSOR DISTANCIA 3 | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| TRIG | PIN RA1 |
| ECHO | PIN RA2 |

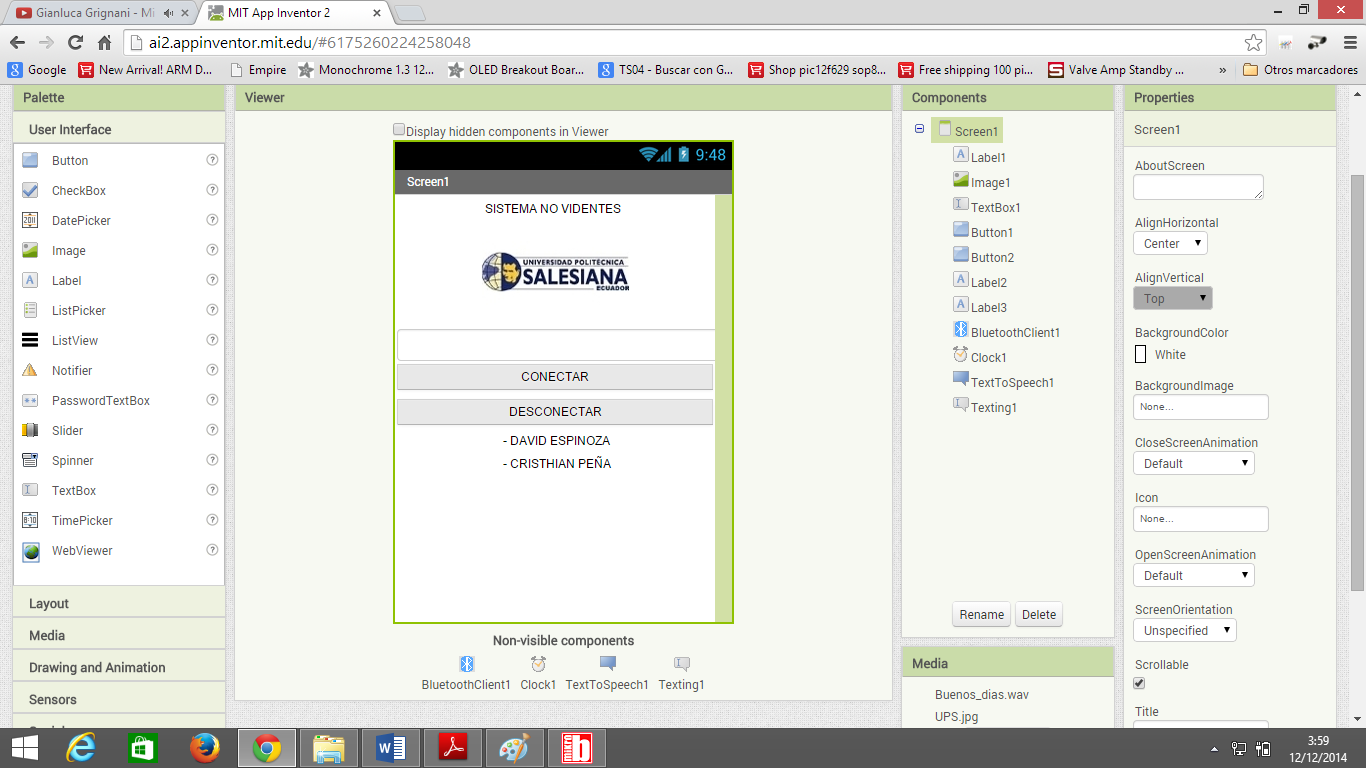
|  |  |
| --- | --- |
| ANTENA XBEE | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| PIN TX | PIN RX |

|  |  |
| --- | --- |
| ANTENA BLUETOOTH | MICROCONTROLADOR |
| - | GND |
| + | + 5V |
| PIN RX | PIN TX |

1. **APLICACIÓN ANDROID**

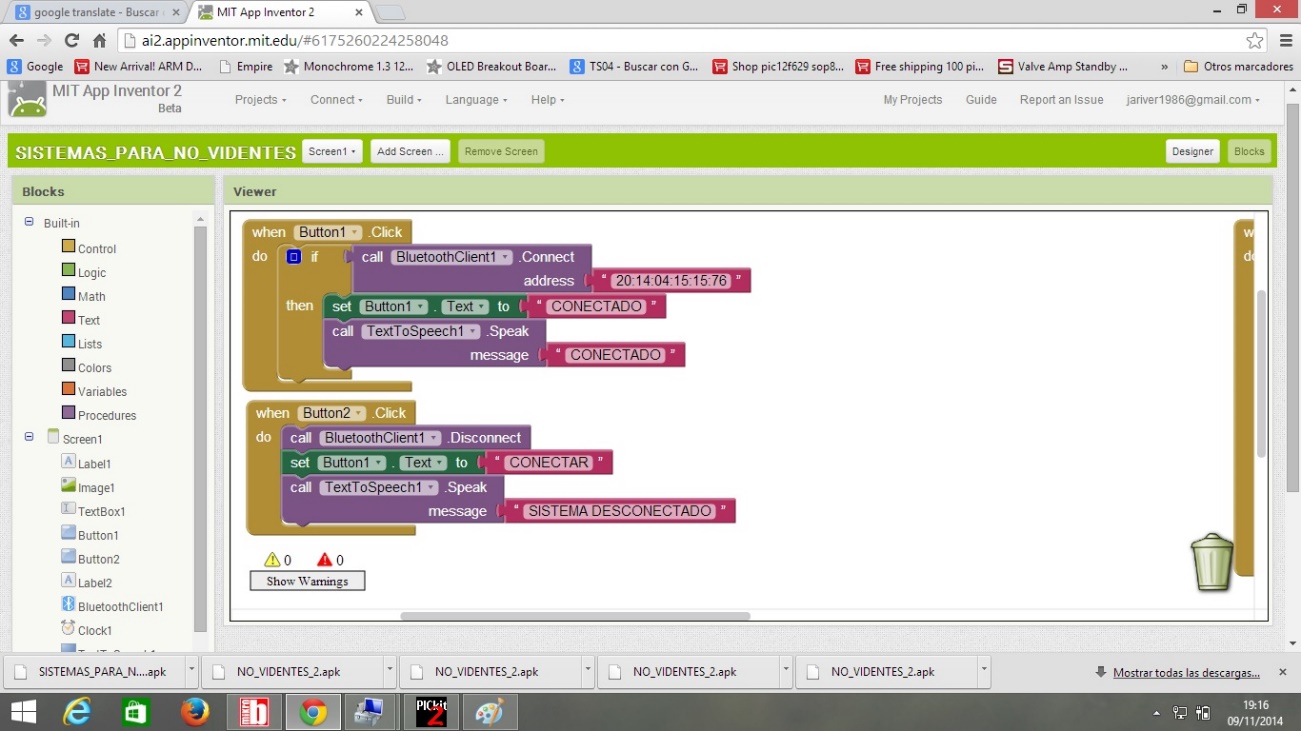
El Instituto Tecnológico de Massachusetts creo una interfaz para el desarrollo de aplicaciones que reducen la curva de aprendizaje con respecto al tiempo, tomando en cuenta muchos requerimientos y exigencias por parte de muchos desarrolladores en el mundo, las aplicaciones van desde un simple hola mundo hasta los sistemas de control muy complejos. Se necesita tener una cuenta de Gmail e ingresar a la página WEB de AppInventor 2.

Sus bloques interactivos de programación poseen una gran acogida en muchas universidades en el mundo e instituciones militares de Estados Unidos y demás países.



Su interfaz grafica de en la figura anterior presenta una paleta de componentes, vista , lista y propiedades, que hacen posible el desarrollo de nuestra aplicación.

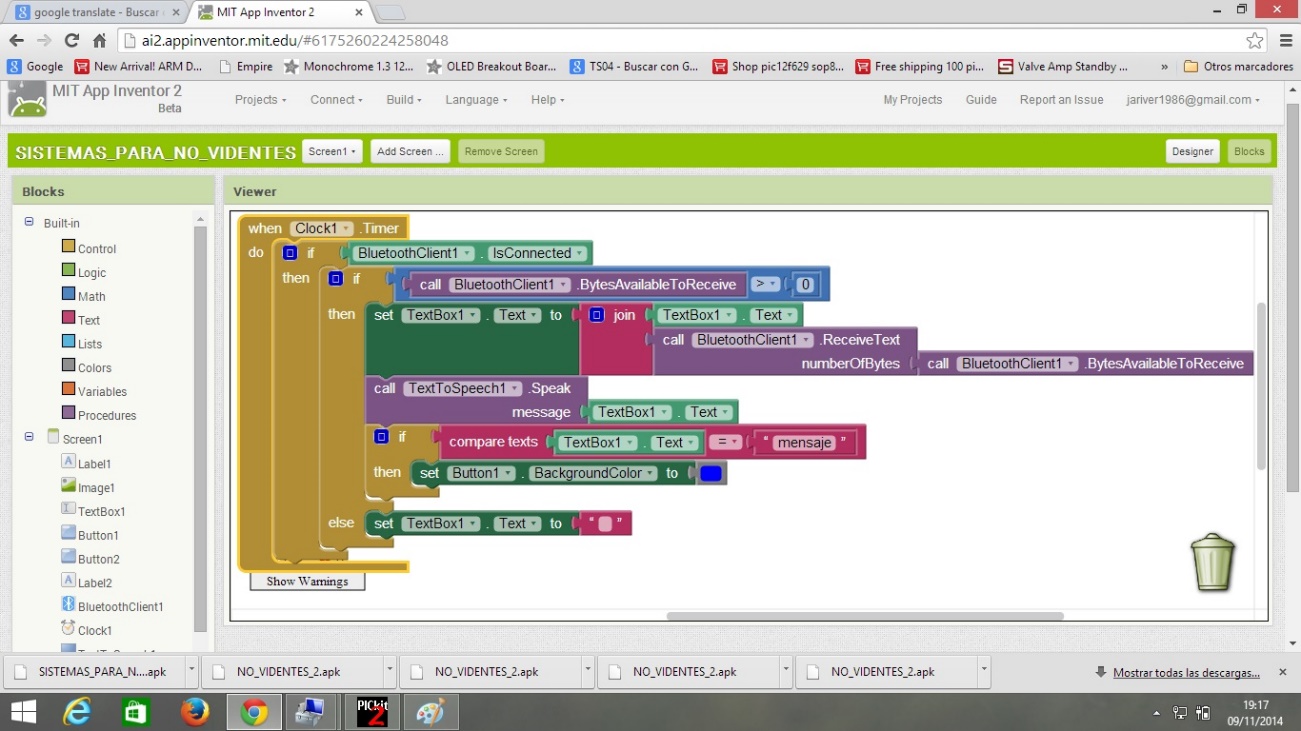
Las siguientes funciones permiten hacer el enlace de datos con el dispositivo de control esclavo.



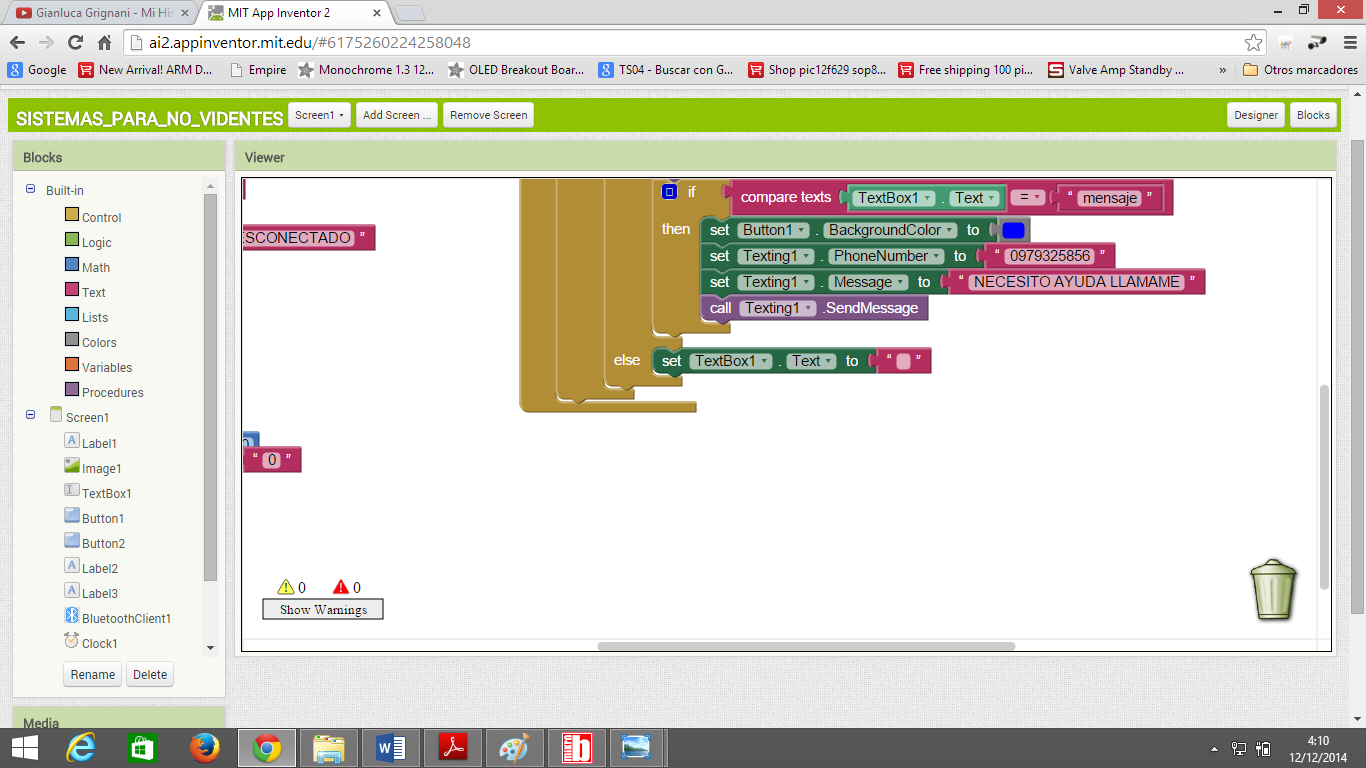
El botón de CONECTAR permite que el celular se comunique con el dispositivo.

El botón de DECCONECTAR permite que el celular rompa la comunicación con el dispositivo.

La siguiente grafica nos permite recibir los datos enviados por el circuito esclavo.



La siguiente figura está programada para enviar un mensaje de texto urgente y manifestar un estado de emergencia hacia un numero de celular previamente establecido.



El programa fue desarrollado online y cualquier cambio se debe de ingresar previamente y descargar el nuevo archivo .apk para ser instalado en el telefono con android con versión mínima de 4.

1. **CODIGO DE PROGRAMACION CIRCUITO MASTER**

program no\_videntes\_botoneras

DIM KP AS BYTE

sub procedure TECLADO()

KP=0

PORTA=%00000110

if PORTB.0=0 then

while(PORTB.0=0) wend

KP= 0X31

end if

if KP=0 then

PORTA=%00000101

if PORTB.1=0 then

while(PORTB.1=0) wend

KP= 0X32

end if

ELSE

RETURN

END IF

if KP=0 then

PORTA=%00000011

if PORTB.2=0 then

while(PORTB.2=0) wend

KP= 0X33

end if

ELSE

RETURN

end if

end sub

main:

OSCCON = 0X75

PORTA = 0

PORTB = 0

PORTC = 0

TRISA = 0

TRISB = %00000111

TRISC = 0

ANSEL = 0

ANSELH= 0

OPTION\_REG = 0X07

WPUB = %00000111

UART1\_Init(9600)

while (1)

TECLADO()

IF KP>0 THEN

if (KP = 0x31) then

UART1\_Write\_Text("F")

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

end if

if (KP = 0x32) then

UART1\_Write\_Text("D")

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

end if

if (KP = 0x33) then

UART1\_Write\_Text("I")

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

PORTA.RA3=1

Delay\_ms(200)

PORTA.RA3=0

Delay\_ms(200)

end iF

Delay\_1sec

Delay\_1sec

Delay\_1sec

END IF

wend

end.

1. **CODIGO DE PROGRAMACION CIRCUITO ESCLAVO**

program no\_videntes

DIM TIEMPO,DATO,TURNO AS BYTE

DIM DISTANCIA1, DISTANCIA2 ,DISTANCIA3 AS WORD

DIM TEXTO AS STRING [10]

sub function ultrasonico1( ) as word

dim flanco as byte

dim distan,tiempo as word

flanco = 0

PORTB.RB0 = 1

Delay\_us(150)

PORTB.RB0 = 0

DO

IF( PORTB.RB1 = 1 )THEN

TMR0 = 0

flanco = 1

END IF

LOOP UNTIL( flanco = 1 )

DO

IF( PORTB.RB1 = 0 )THEN

tiempo = TMR0

flanco = 0

END IF

'if(TMR0>254)THEN

' FLANCO=0

' END IF

LOOP UNTIL( FLANCO = 0 )

distan = (tiempo\*128)/58

Delay\_ms (50)

result = distan

end sub

sub function ultrasonico2( ) as word

dim flanco as byte

dim distan,tiempo as word

flanco = 0

PORTA.RA3 = 1

Delay\_us(150)

PORTA.RA3 = 0

DO

IF( PORTA.RA4 = 1 )THEN

TMR0 = 0

flanco = 1

END IF

LOOP UNTIL( flanco = 1 )

DO

IF( PORTA.RA4 = 0 )THEN

tiempo = TMR0

flanco = 0

END IF

' if(TMR0>254)THEN

' FLANCO=0

' END IF

LOOP UNTIL( FLANCO = 0 )

distan = (tiempo\*128)/58

Delay\_ms (50)

result = distan

end sub

sub function ultrasonico3( ) as word

dim flanco as byte

dim distan,tiempo as word

flanco = 0

PORTA.RA1 = 1

Delay\_us(150)

PORTA.RA1 = 0

DO

IF( PORTA.RA2 = 1 )THEN

TMR0 = 0

flanco = 1

END IF

LOOP UNTIL( flanco = 1 )

DO

IF( PORTA.RA2 = 0 )THEN

tiempo = TMR0

flanco = 0

END IF

' if(TMR0>254)THEN

' FLANCO=0

' END IF

LOOP UNTIL( FLANCO = 0 )

distan = (tiempo\*128)/58

Delay\_ms (50)

result = distan

end sub

main:

OPTION\_REG = %00000110

OSCCON = 0x65

PORTA = 0

TRISA = %00010100

PORTB = 0

TRISB = %00101010

PORTC = 0

TRISC = %10000000

ANSEL =0

ANSELH = 0

UART1\_Init(9600)

TURNO=0

WHILE (1)

if UART1\_Data\_Ready=1 then

DATO = UART1\_Read

end if

if DATO = "F" then

distancia1 = ultrasonico1()

WordToStr(DISTANCIA1,TEXTO)

UART1\_Write\_Text(" DISTANCIA OBJETO EN FRENTE ")

UART1\_Write\_Text(TEXTO)

UART1\_Write\_Text(" centimetros")

DATO=0

end if

if DATO = "I" then

distancia2 = ultrasonico2()

distancia3 = ultrasonico3()

WordToStr(DISTANCIA2,TEXTO)

UART1\_Write\_Text(" DISTANCIA OBJETO A LA IZQUIERDA ")

UART1\_Write\_Text(TEXTO)

UART1\_Write\_Text(" centimetros")

WordToStr(DISTANCIA3,TEXTO)

UART1\_Write\_Text(" DISTANCIA OBJETO A LA DERECHA ")

UART1\_Write\_Text(TEXTO)

UART1\_Write\_Text(" centimetros")

Delay\_1sec

Delay\_1sec

DATO=0

end if

if DATO = "D" then

UART1\_Write\_Text("mensaje")

Delay\_1sec

Delay\_1sec

DATO=0

end if

Delay\_ms(200)

WEND

end.

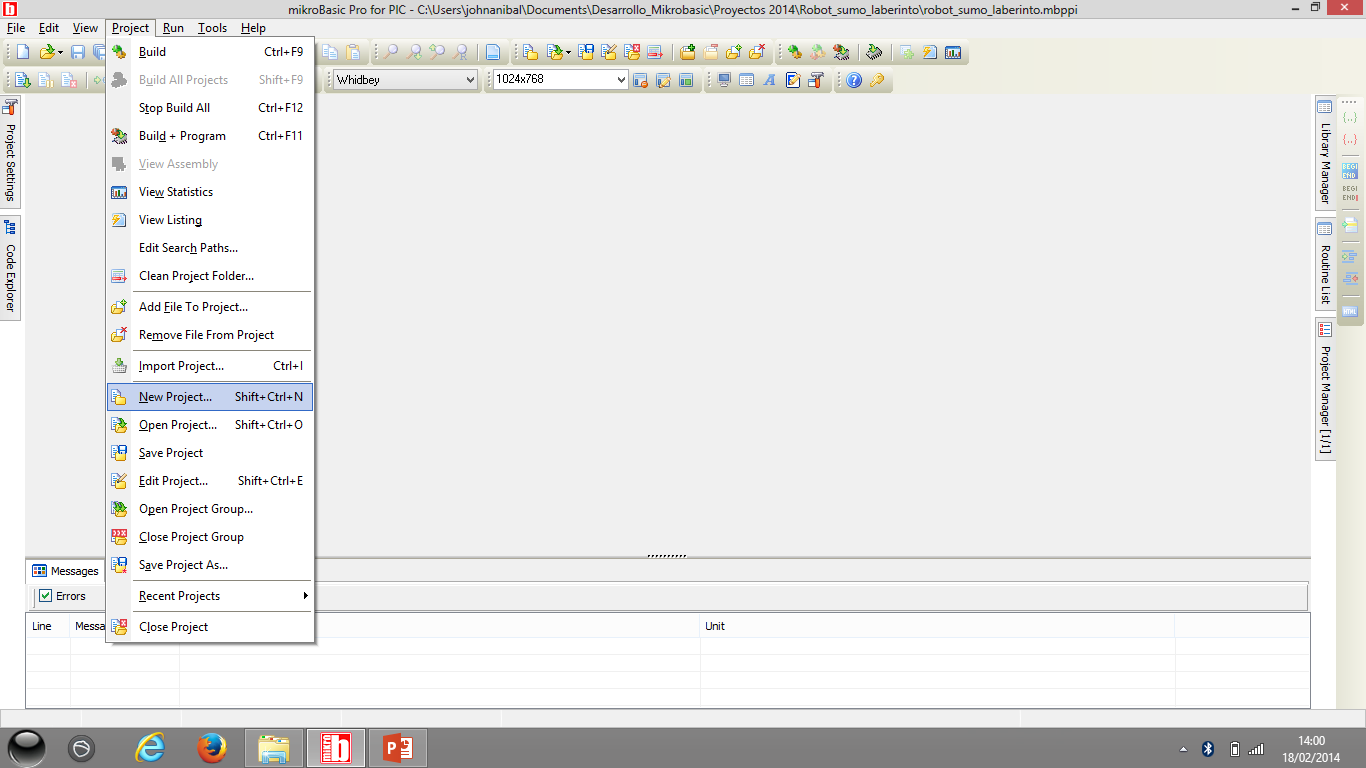
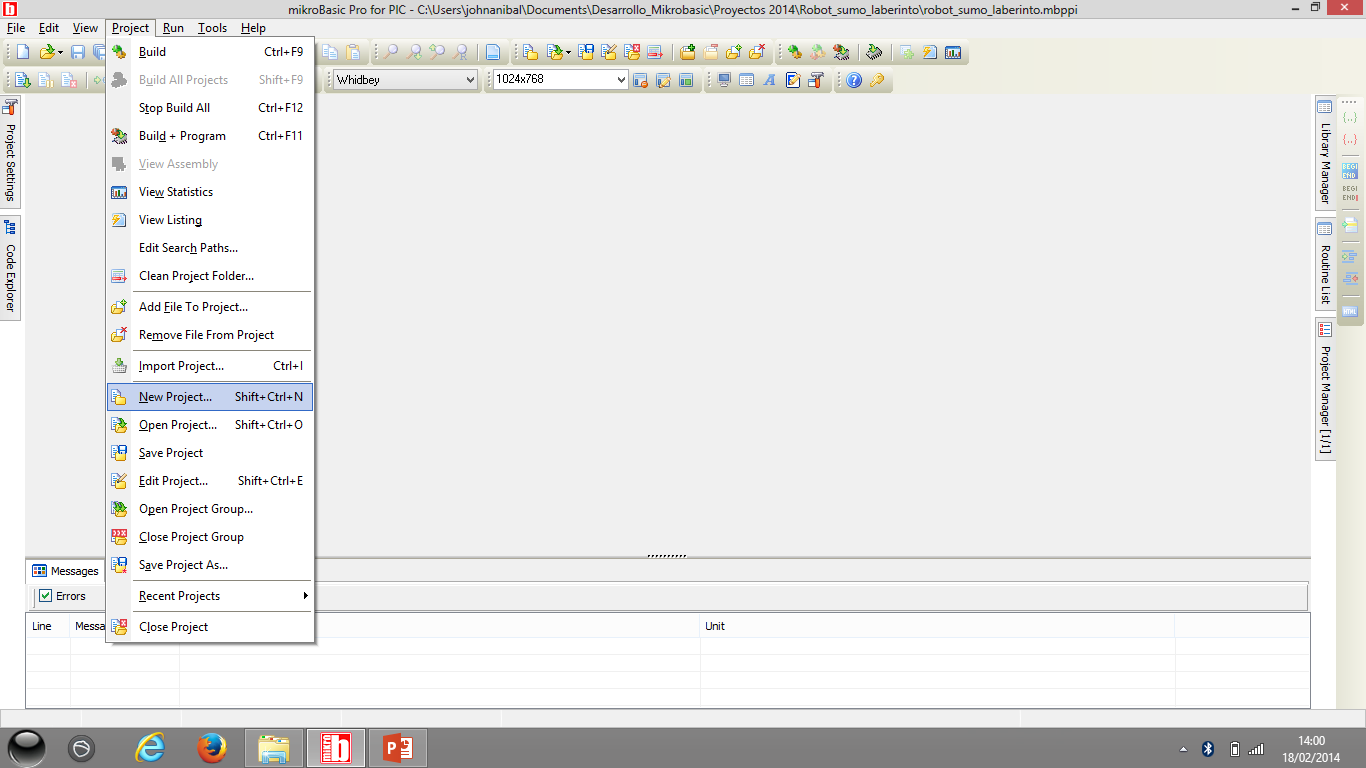
**ANEXOS**

1. **CREACION DE UN PROYECTO EN MIKROBASIC PRO FOR PIC**

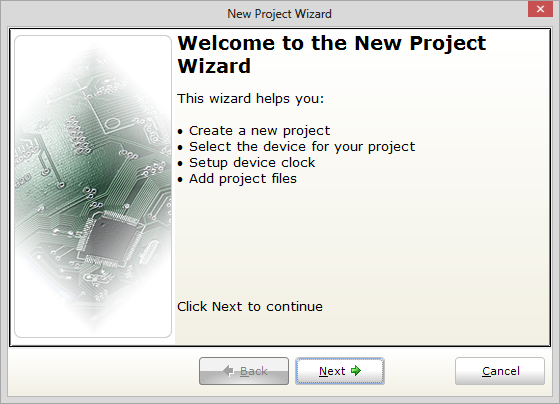
PASO # 1 EJECUTAR COMO ADMINISTRADOR MIKROBASIC PRO FOR PIC



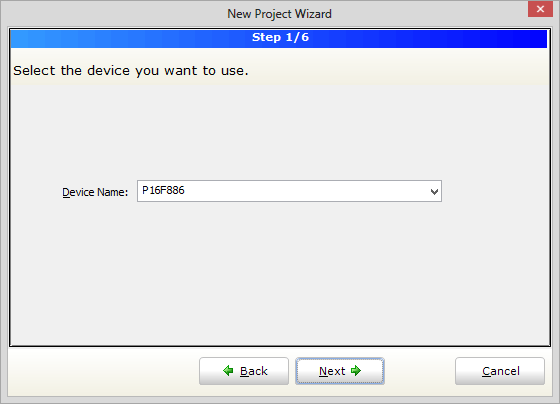
PASO # 2 PROJECT –NEW PROJECT



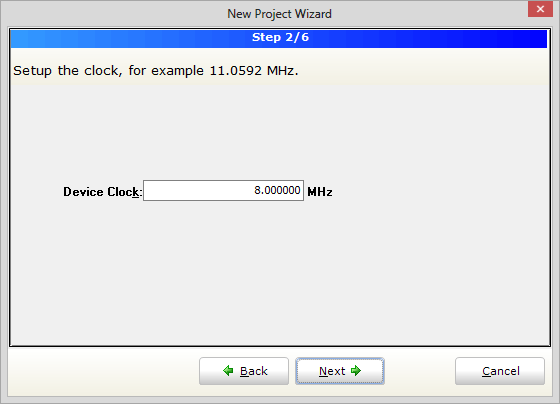
PASO # 3 CLICK EN NEXT



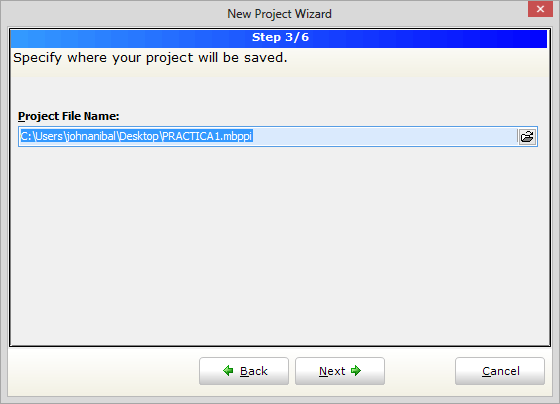
PASO # 4 DEVICE NAME: 16F886 -CLICK NEXT



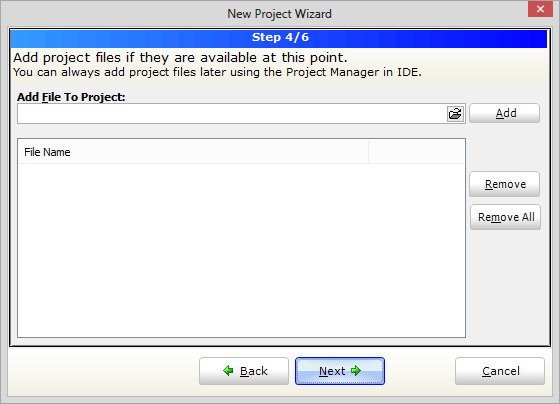
PASO # 5 DEVICE CLOCK: 8.000000 -CLICK NEXT



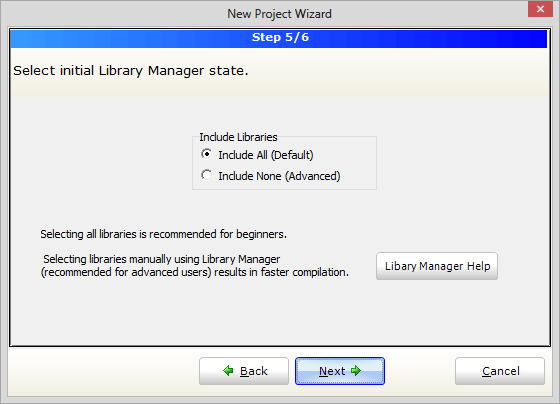
PASO # 6 GUARDAR EL ARCHIVO -CLICK NEXT



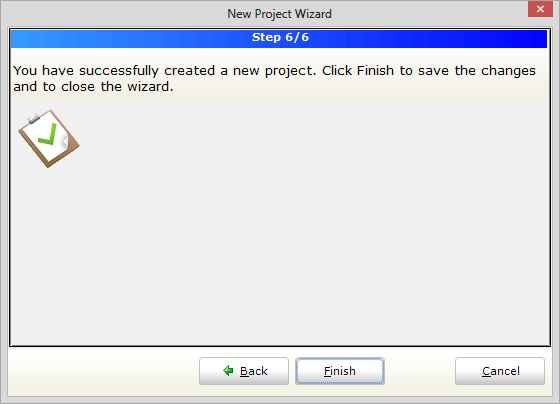
PASO # 7 CLICK NEXT



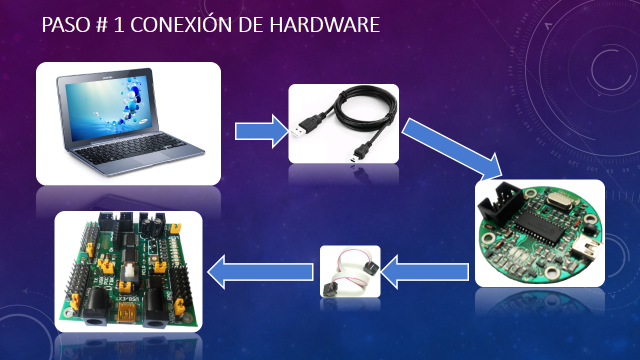
PASO # 8 CLICK NEXT



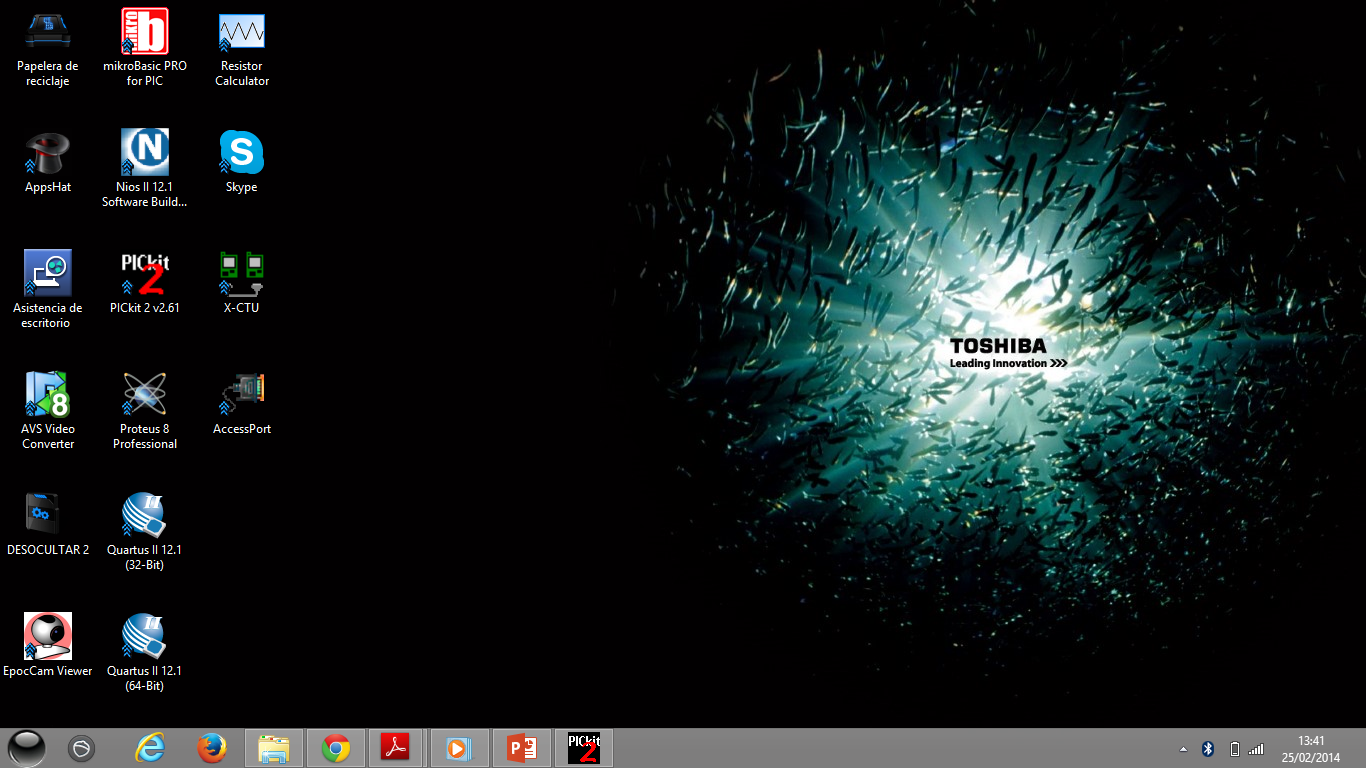
PASO # 9 CLICK NEXT



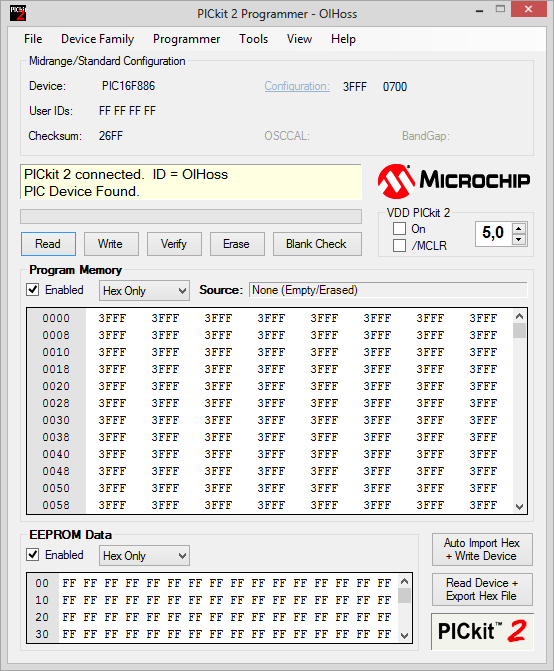
1. **PROGRAMACION DE UN MICROCONTROLADOR CON PICKIT2**

****

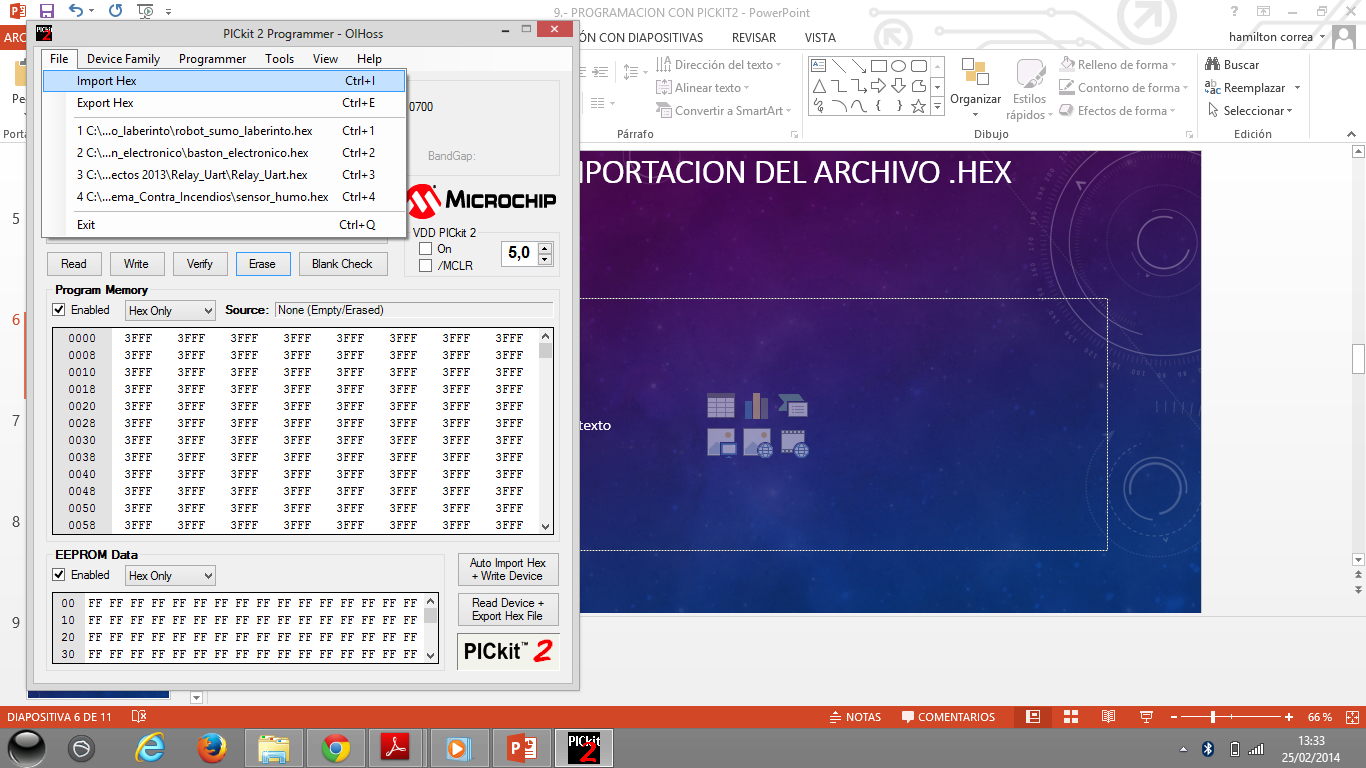
Ejecute el icono de PICKit

****

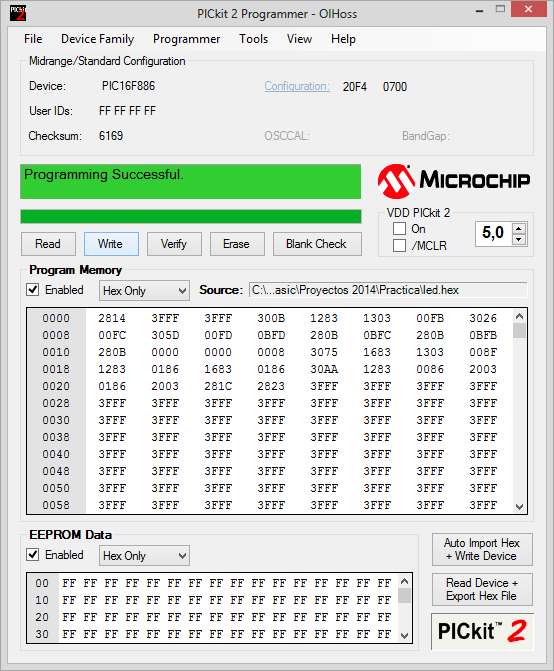
Hacer click en menu tool – check comunication

****

Observe que el dispositivo haya sido detectado y proceda a borrar la información con el botón de ERASE.

Importe el archivo .HEX 

Programe el dispositivo haciendo click en el botón de WRITE



**FUENTE BILBIOGRÁFICAS DE ANEXOS**

[**www.ideastechnology.com**](http://www.ideastechnology.com)

**DIAGRAMA PCB**

