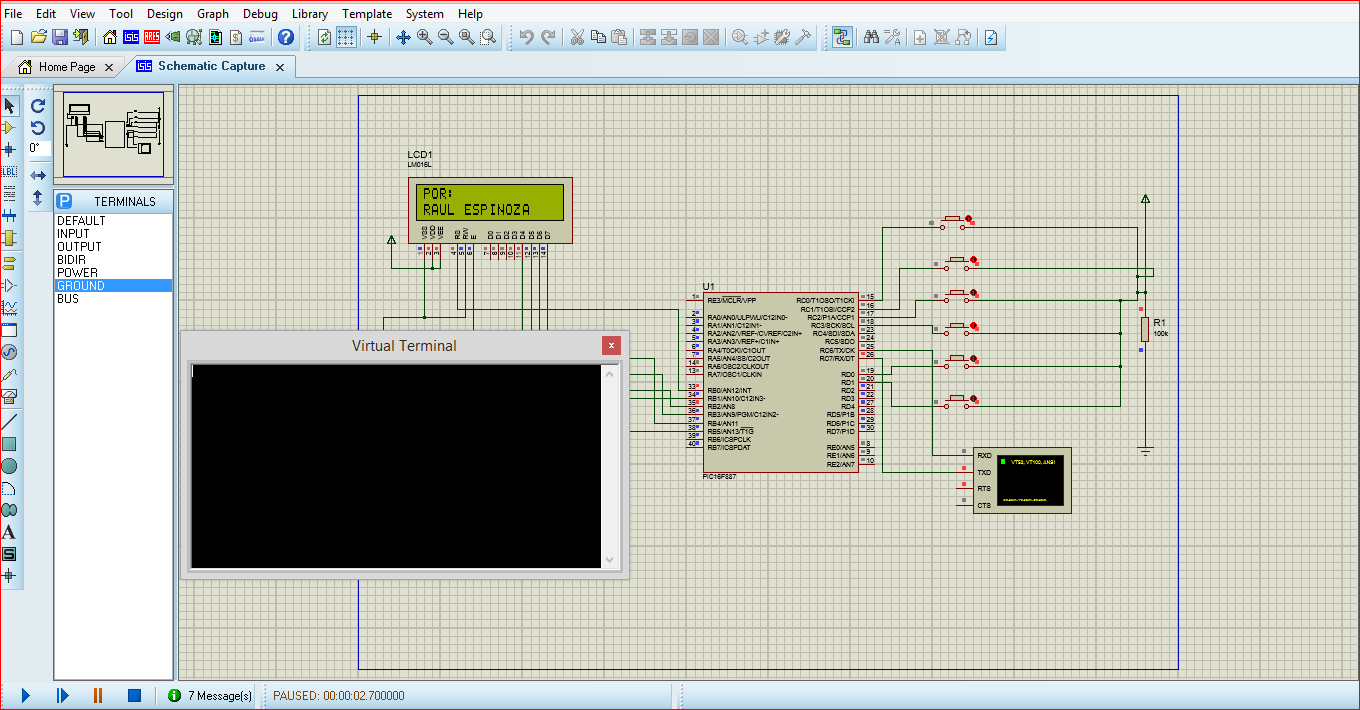
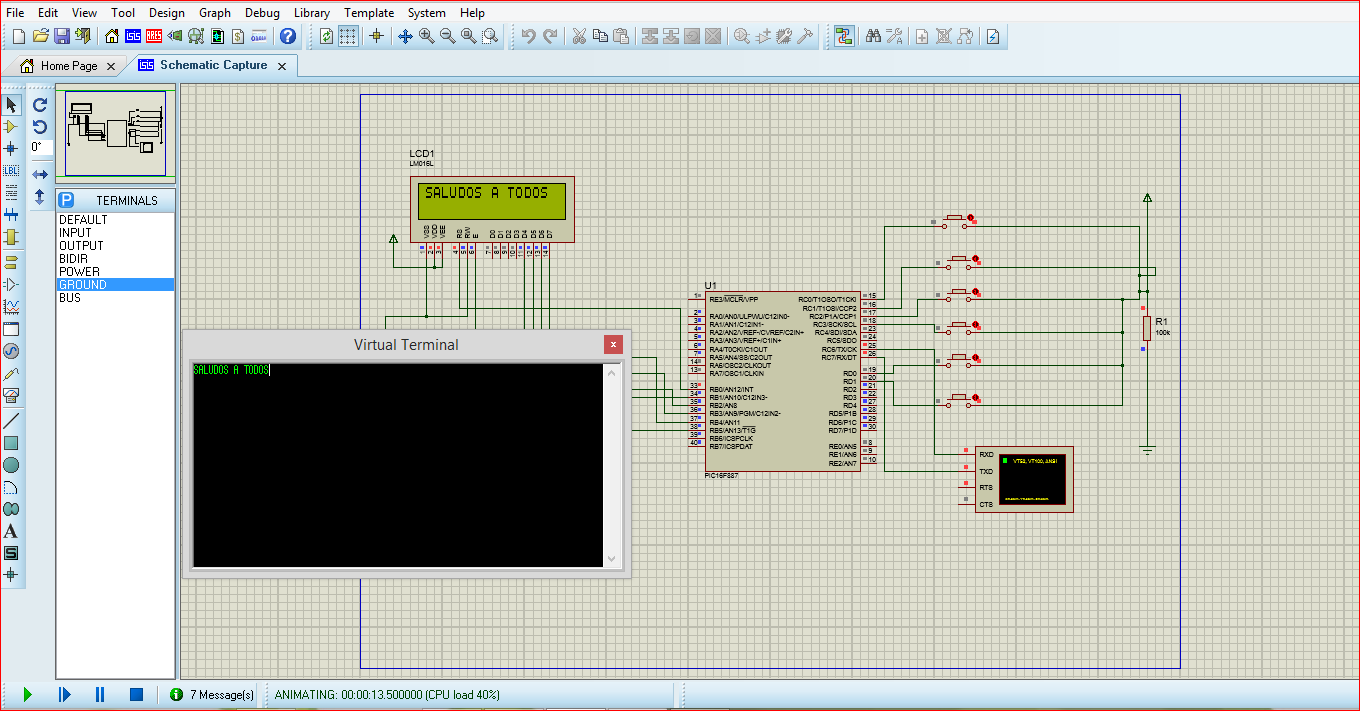
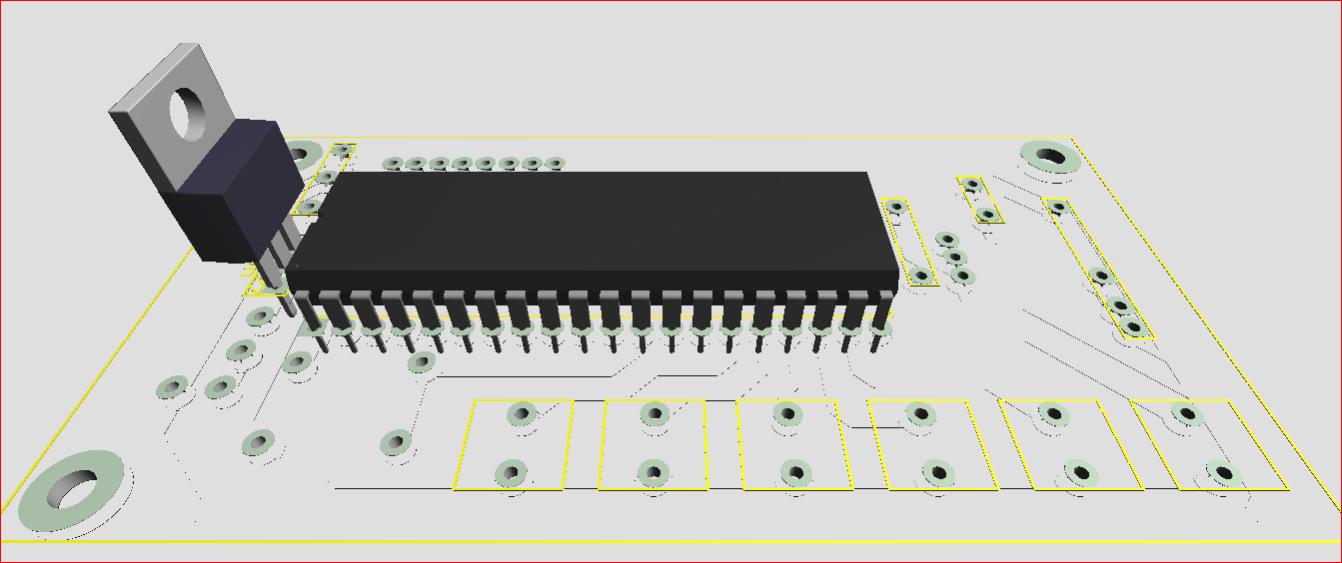
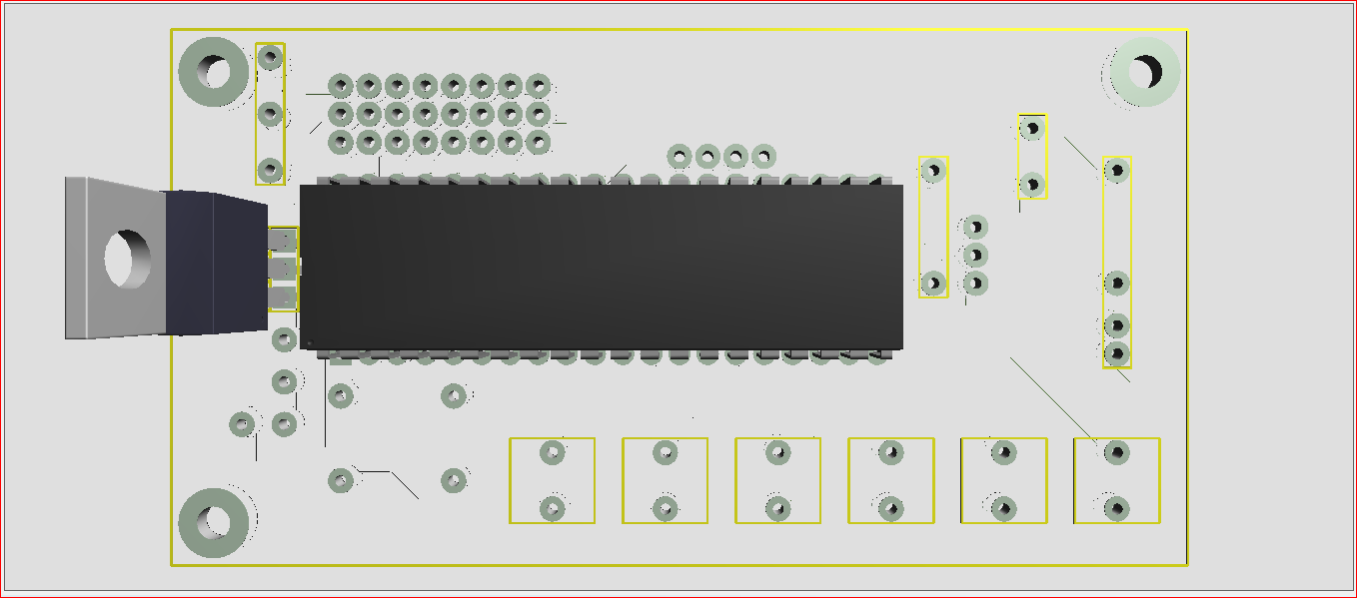
SIMULACION



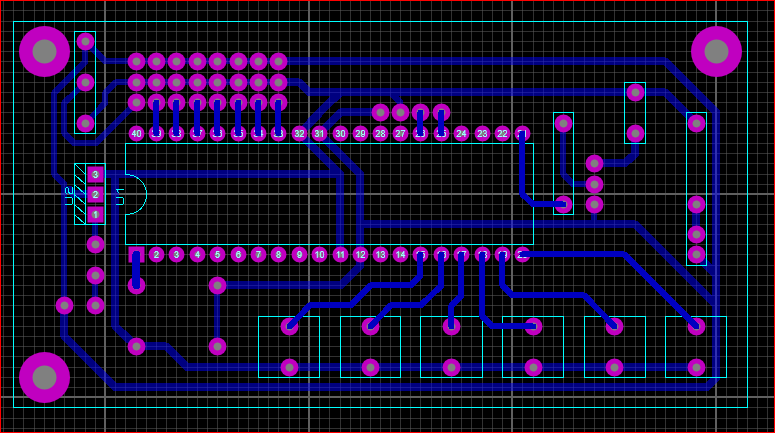


VISTA 3D





DISEÑO ARES



CODIGO

program Sistema\_comunicacion

symbol BOTON1 = PORTC.RC0

symbol BOTON2 = PORTC.RC1

symbol BOTON3 = PORTC.RC2

symbol BOTON4 = PORTC.RC3

symbol BOTON5 = PORTD.RD0

symbol BOTON6 = PORTD.RD1

' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CONECXIONES PARA EL MODULO LCD\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

dim LCD\_RS as sbit at RB0\_bit

LCD\_EN as sbit at RB1\_bit

LCD\_D4 as sbit at RB2\_bit

LCD\_D5 as sbit at RB3\_bit

LCD\_D6 as sbit at RB4\_bit

LCD\_D7 as sbit at RB5\_bit

LCD\_RS\_Direction as sbit at TRISB0\_bit

LCD\_EN\_Direction as sbit at TRISB1\_bit

LCD\_D4\_Direction as sbit at TRISB2\_bit

LCD\_D5\_Direction as sbit at TRISB3\_bit

LCD\_D6\_Direction as sbit at TRISB4\_bit

LCD\_D7\_Direction as sbit at TRISB5\_bit

' \*\*\*\*\*\*\*\*\*\*VARIABLES Y CADENAS DE CARACTERES A UTILIZAR\*\*\*\*\*\*\*\*\*\*

dim txt2 as char[33]

dim counter as byte

' Declarations section

main:

' Registro Oscilador de control

OSCCON = 0X75 ' Oscilador interno de 8mhz

' Registro PUERTO A

TRISA = 0X00 ' PORTA.0 Entrada

PORTA = 0X00

' Registro PUERTO B

TRISB = 0X00 ' PORTB salidas

PORTB = 0X00

' Registro PUERTO C

TRISC = 0XFF ' PORTC salidas

PORTC = 0X00

' Registro PUERTO D

TRISD = 0X03 ' PORTE como salidas digital

PORTD = 0X00

' Seleccion de registro analogico. 1 analogico, 0 digitales

ANSEL = 0X00 ' AN<7:0>

ANSELH = 0X00 ' AN<13:8>

Lcd\_Init() ' Inicializacion Lcd

Lcd\_Cmd(\_LCD\_CLEAR) ' encera display

Lcd\_Cmd(\_LCD\_CURSOR\_OFF) ' Cursor off

Lcd\_Out(1,1,"SISTEMA XBEE ") ' escribe la cadena en la primera fila

Lcd\_Out(2,1,"COMUNICACION ") ' escribe la cadena en la segunda fila

delay\_ms(2000)

Lcd\_Cmd(\_LCD\_CLEAR) ' encera display

Lcd\_Out(1,1,"POR:") ' escribe la cadena en la primera fila

Lcd\_Out(2,1,"RAUL ESPINOZA") ' escribe la cadena en la segunda fila

delay\_ms(2000)

Lcd\_Cmd(\_LCD\_CLEAR) ' encera display

delay\_ms(250)

UART1\_Init(9600)

while(1)

if (UART1\_Data\_Ready() <> 0) then ' If data is received,

UART1\_Read\_Text(txt2,"-",32) ' lee la cadena hasta que encuentre la letra K mayuscula

UART1\_Write\_Text(txt2) ' envia un eco de la cadena

Lcd\_Cmd(\_LCD\_CLEAR) ' encera display

for counter = 0 to 15

Lcd\_Chr(1,counter+1,txt2[counter])

next counter

for counter = 0 to 15

Lcd\_Chr(2,counter+1,txt2[counter+16])

next counter

txt2 =" "

end if

if BOTON1=1 then

while BOTON1=1 wend

UART1\_Write\_Text("OK")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

if BOTON2=1 then

while BOTON2=1 wend

UART1\_Write\_Text("RECIBIDO")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

if BOTON3=1 then

while BOTON3=1 wend

UART1\_Write\_Text("ESTOY EN CAMINO")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

if BOTON4=1 then

while BOTON4=1 wend

UART1\_Write\_Text("ESTOY OCUPADO ")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

if BOTON5=1 then

while BOTON5=1 wend

UART1\_Write\_Text("ESTOY LIBRE")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

if BOTON6=1 then

while BOTON6=1 wend

UART1\_Write\_Text("NECESITO AYUDA")

UART1\_Write(0X0D)

UART1\_Write(0X0A)

end if

wend

end.