



Universidad Nacional Autónoma de México

Facultad de Ingeniería

Fundamentos de Sistemas Embebidos

Documentación de controladora de ventiladores con
iluminación integrada

Grupo: 01

Integrantes: Fuentes Díaz Alan Abner
Mancilla Checa Luis Enrique

Investigación del costo de los componentes

A continuación mostraremos una tabla la cuales se compararon precio de páginas de internet así como algunos precios que previamente hemos visto, puesto que comprábamos material antes de la pandemia.

Componentes	Cantidad	Precio
PIC16F887	1	\$105
Potenciómetro 1K Ohm	1	\$10
555	1	\$10
Capacitor 10uF	1	\$3
Motor DC	2	\$33
Transistor BC548B	12	\$5
Diodo 1N4001A	2	\$3
Led RGB	8	\$7
Resistencia 330 ohm	1	\$0.50
Resistencia 100 ohm	1	\$0.50

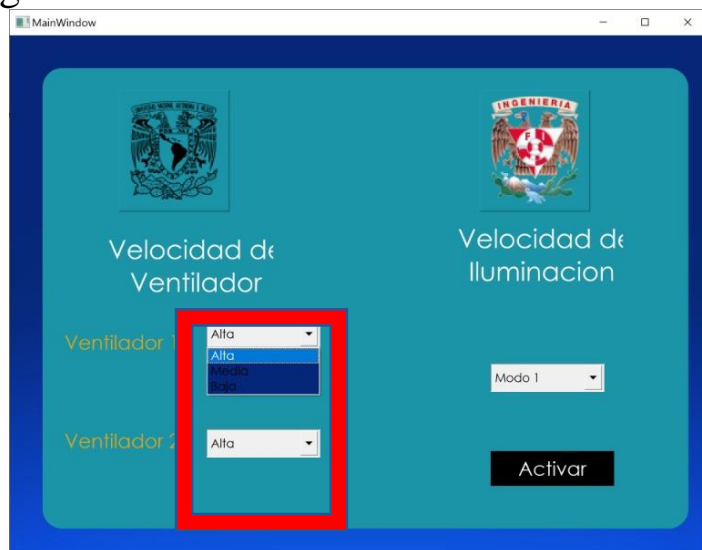
Para nuestro proyecto se decidió utilizar el PIC16f887 ya que lo hemos utilizado antes y a nuestro parecer es eficaz y fácil de conseguir, adicionando que hay muchos diagramas de cómo está formado así como la documentación, sin embargo este pic solo cuenta con 2 PWM, un detalle que se nos quedara corto esos 2 PWM porque se utilizaran para nuestros ventiladores y los led que ocupamos. Intentamos corregir esto con un 555 para generar otro PWM siendo controlado por un potenciómetro.

Como sabemos con el potenciómetro vamos a regular el PWM dado que el ancho de pulso estará conectado a los led.

Lista de comandos

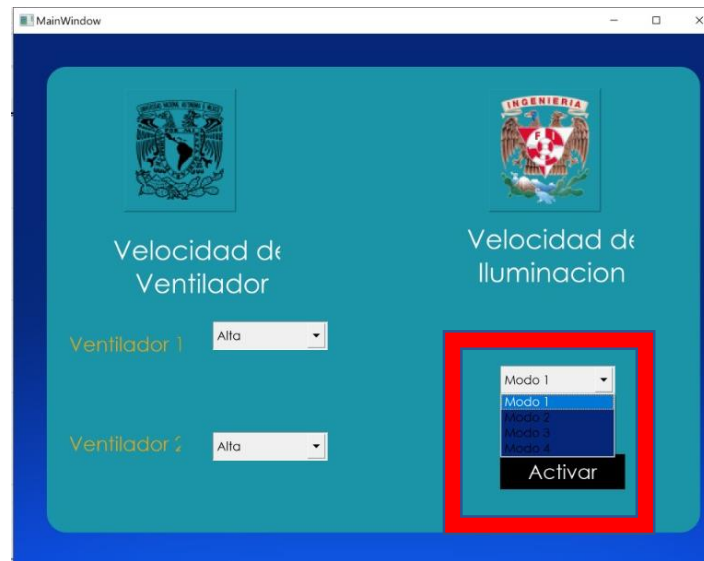
Usuario

Del lado de Velocidad de Ventilador, pusimos una parte donde se puede seleccionar Alta, Media y Baja, esto refiriéndonos a la velocidad de giro del ventilador

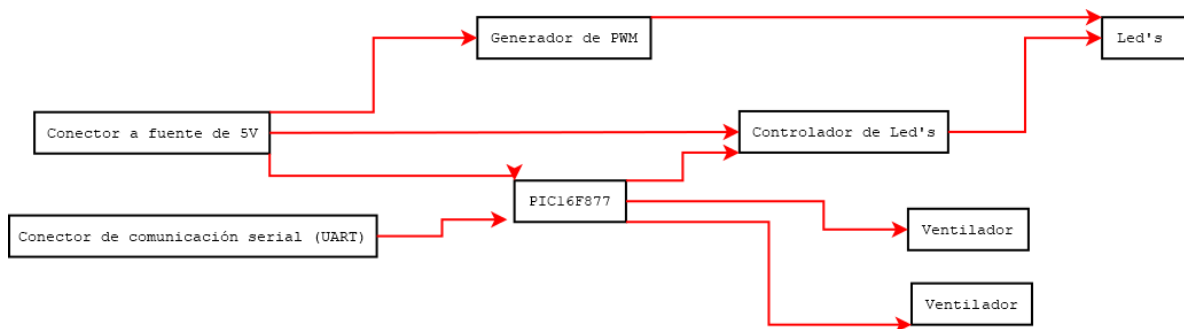


Adicional a ello para la iluminación tendremos 4 modos para que estas puedan ser controladas, a continuación mostramos una tabla para ver el significado de: Modo 1, Modo 2, Modo 3 y Modo 4

Nombre	Led prendidos	Secuencia de color	Colores por las que pasa
Modo 1	4	Completa	8
Modo 2	8	Completa	8
Modo 3	4	Sencilla	4
Modo 4	8	Sencilla	4



Arquitectura



Funcionamiento de ventilador y LED

Controlador de ventiladores		
Variable	Ventilador 1	Ventilador 2
‘a’	100%	100%
‘b’	100%	70%
‘c’	100%	40%
‘d’	70%	100%
‘e’	70%	70%
‘f’	70%	40%
‘g’	40%	100%
‘h’	40%	70%
‘i’	40%	40%

Controlador de iluminación			
Variable	Ventiladores y leds	Secuencia	Numero de colores
‘j’	El sistema de iluminación de los dos ventiladores y preden dos leds de cada ventilador.	Compleja	8 colores
‘k’	El sistema de iluminación de los dos ventiladores y preden cuatro leds de cada ventilador.	Compleja	8 colores
‘l’	El sistema de iluminación de los dos ventiladores y preden dos leds de cada ventilador.	Sencilla	4 colores
‘m’	El sistema de iluminación de los dos ventiladores y preden cuatro leds de cada ventilador.	Sencilla	4 colores

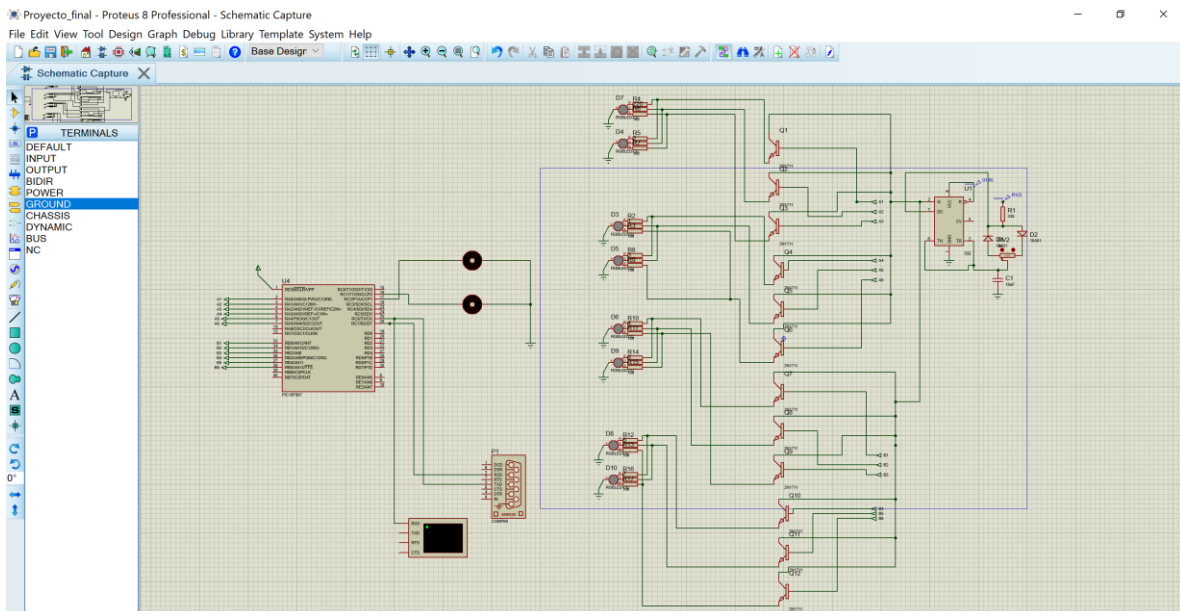
Estimación de costo por equipo

Componentes	Cantidad	Precio
PIC16F887	1	\$105
Potenciómetro 1K Ohm	1	\$10
555	1	\$10
Capacitor 10uF	1	\$3
Motor DC	2	\$66
Transistor BC548B	12	\$60
Diodo 1N4001A	2	\$6
Led RGB	8	\$56
Resistencia 330 ohm	1	\$0.50
Resistencia 100 ohm	1	\$0.50
	TOTAL	\$317

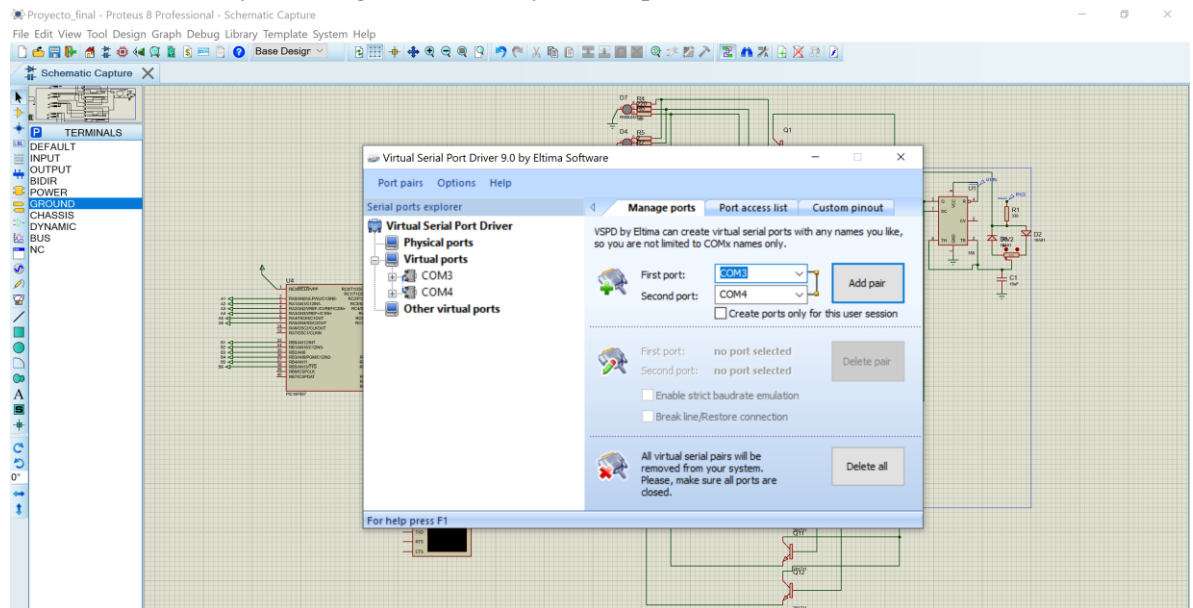
Cabe mencionar que es una estimación, y que no está incluido la mano de obra, la placa PBC que utilizaremos las mandaremos hacer, y los materiales serán aparte para el ensamblaje, como mencionamos pensamos que es solo un estimado por lo que puede aumentar o disminuir, así como también se puede hacer más complejo como se requiera

Manual de usuario

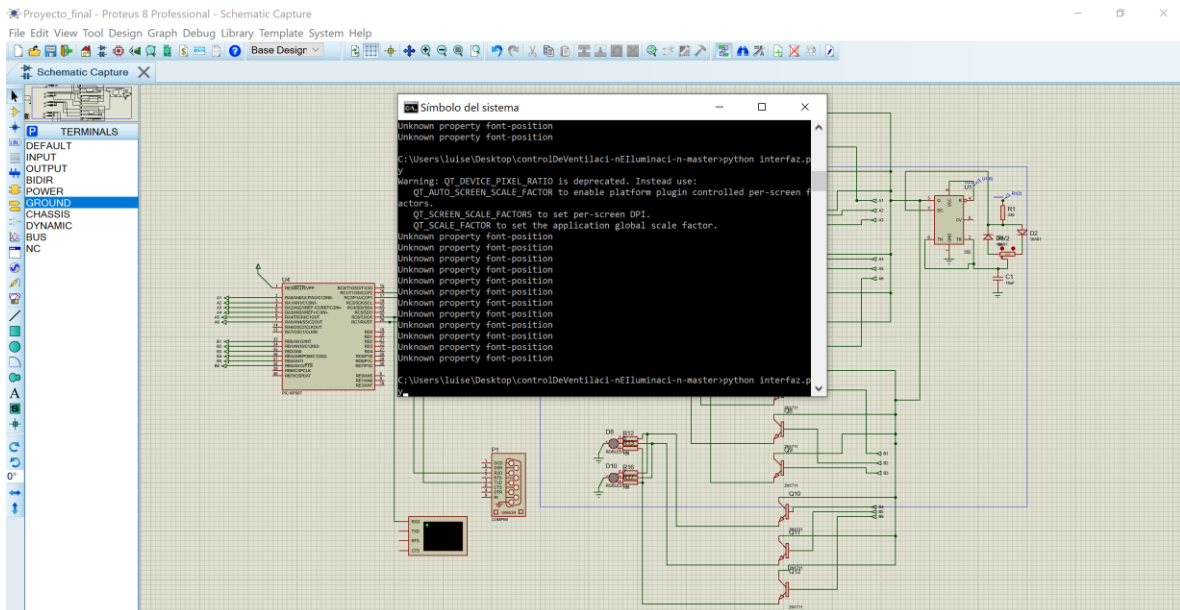
1. Para simular el proyecto se tiene que abrir el archivo “Proyecto_final.pdsprj” y se abrirá la simulación en Proteus:



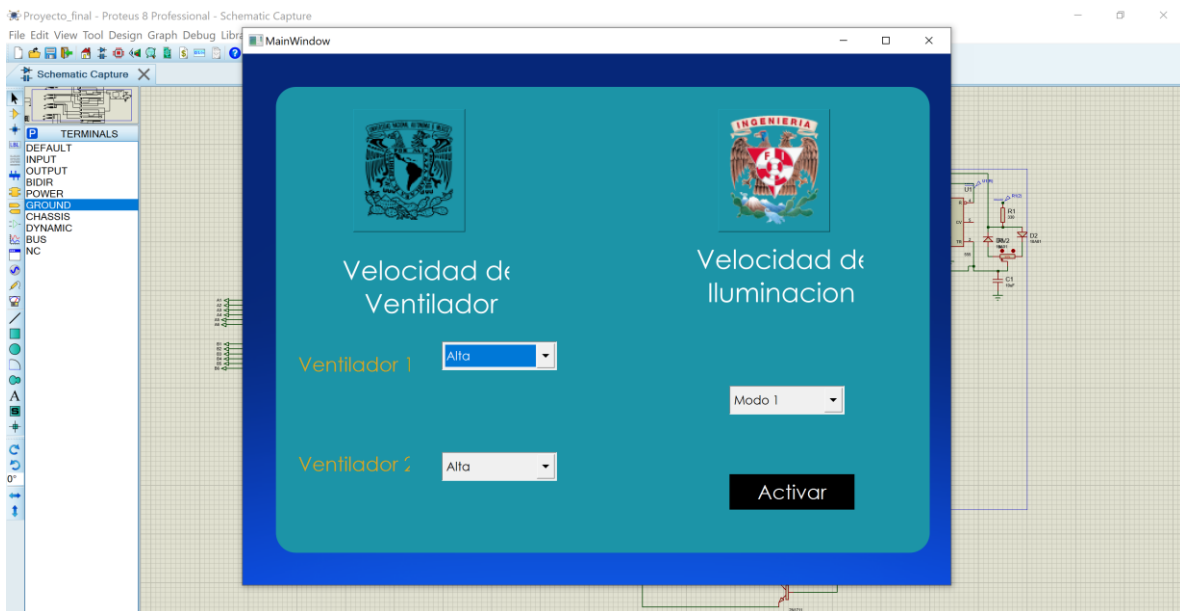
2. Se prosigue a realizar la virtualización de puertos COM, esto mediante el programa “Virtual Serial Port Driver” y se configura el COM3 y COM4 para la comunicación:



3. El siguiente paso es ejecutar el programa .py llamado interfaz.py desde el símbolo del sistema:



4. Finalmente observamos la interfaz de usuario funcionando con la comunicacion hacia nuestra simulacion en proteus:



Programas utilizados:

- Para realizar la simulación se utilizó del programa Proteus 8:



Proteus 8 Professional

Aplicación

- Para realizar la virtualización de los puertos para enlazar la comunicación se utilizo el programa “Virtual Serial Port Driver”:



Configure Virtual Serial Port Driver

Aplicación

- Para realizar la interfaz de usuario utilizamos el programa “Qt Creator”:

