



Nombre de la práctica	Proto board			No.	1
Asignatura:	Arquitectura de la computadora	Carrera:	Ingeniería en Sistemas Computacionales	Duración de la práctica (Hrs)	

Alumno: Ismael Alcántara Bueno

312

I. Competencia(s) específica(s):

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro): Aula

III. Material empleado:

- 1 proto Board
- 2 capacitores de 22 picofaradios
- 1 cristal oscilador de 4MHz
- 1 catodo común de 7 segmentos
- 1 PIC16F84A
- Cable utp

Programas:

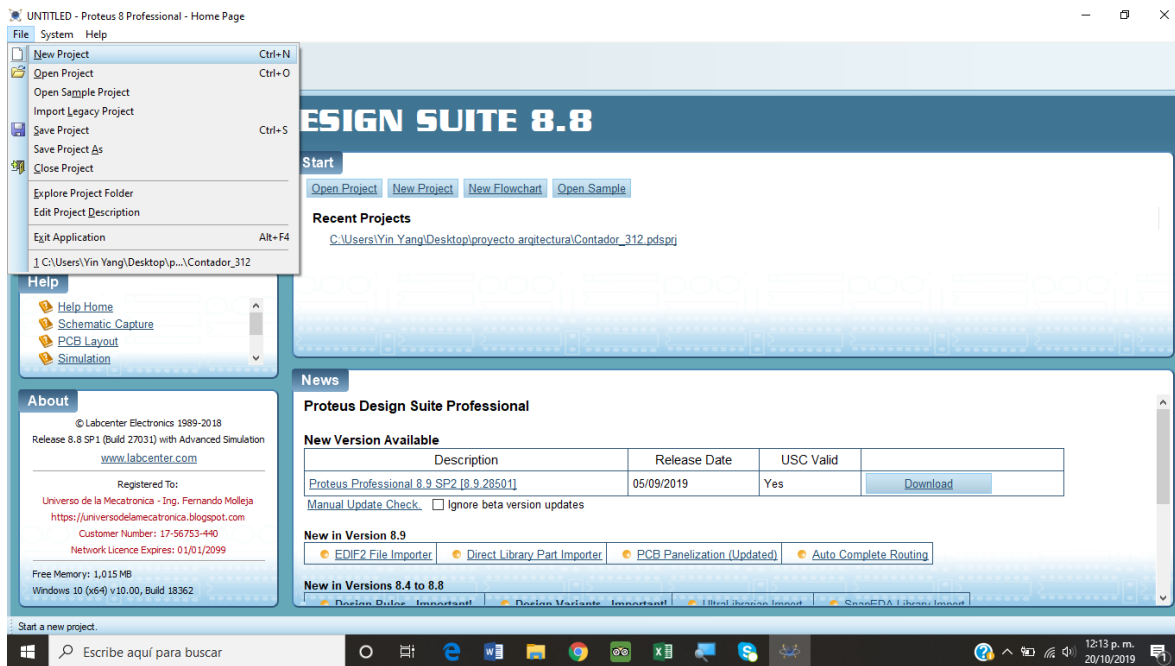
- Proteus 8
- MPLAB

IV. Desarrollo de la práctica:

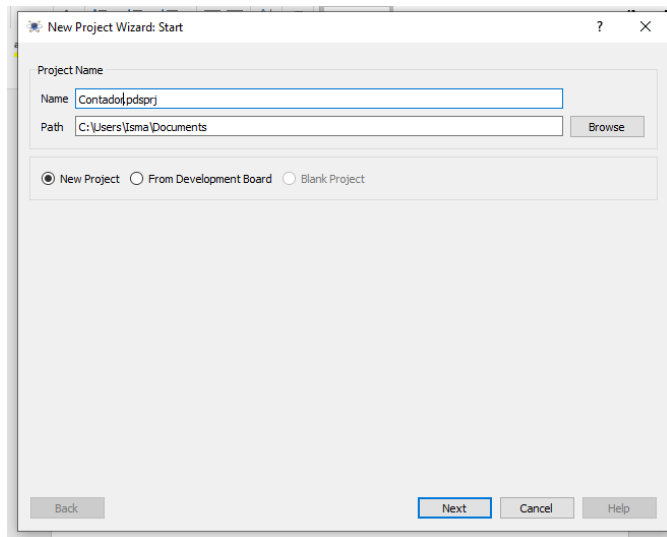
V. Conclusiones:

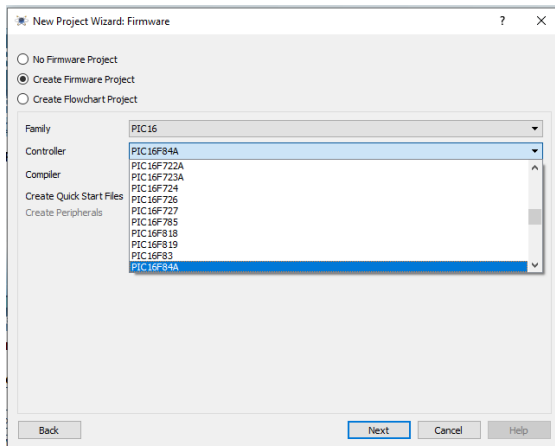
Contador, cátodo común de 7 segmentos

1.- Se abre el proteus y se crea un nuevo proyecto.

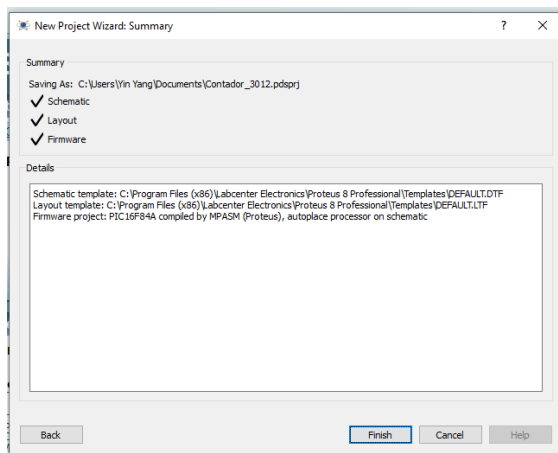


2.- Aquí se le da nombre al proyecto dejando la extensión del mismo.

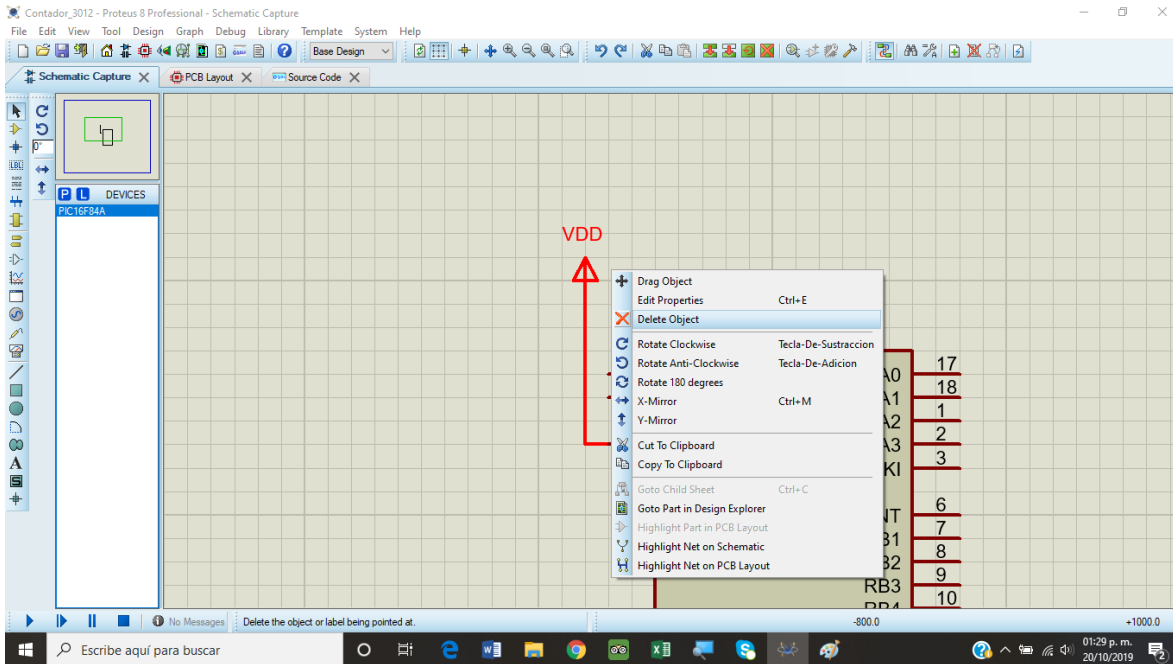




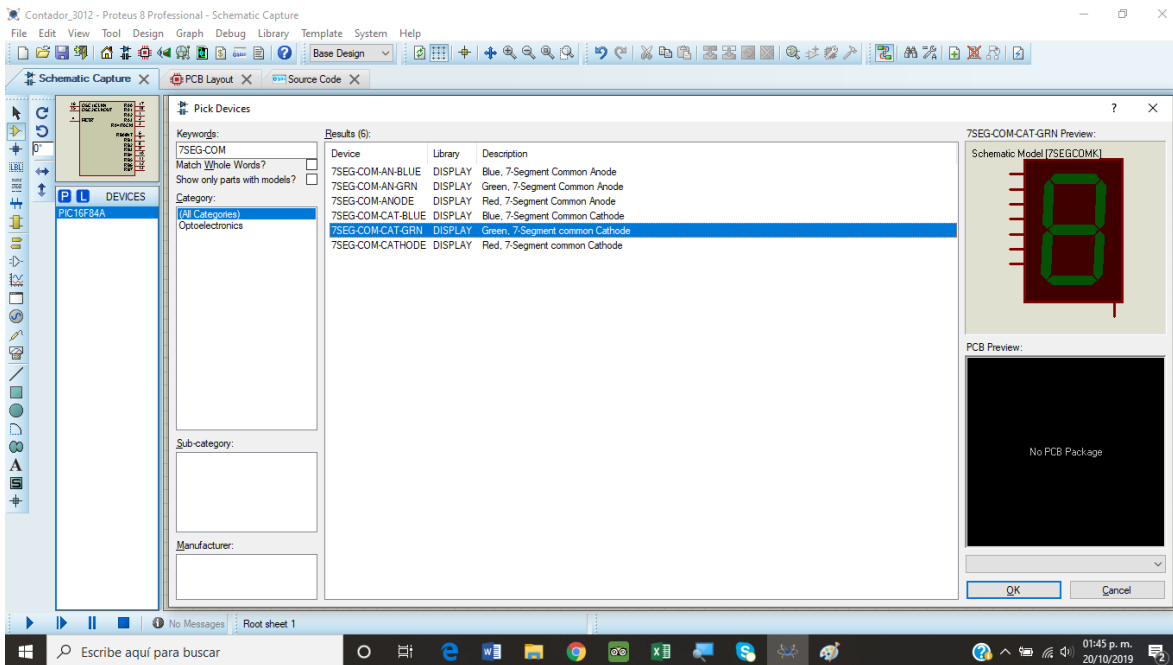
6.- Se da click en finish.

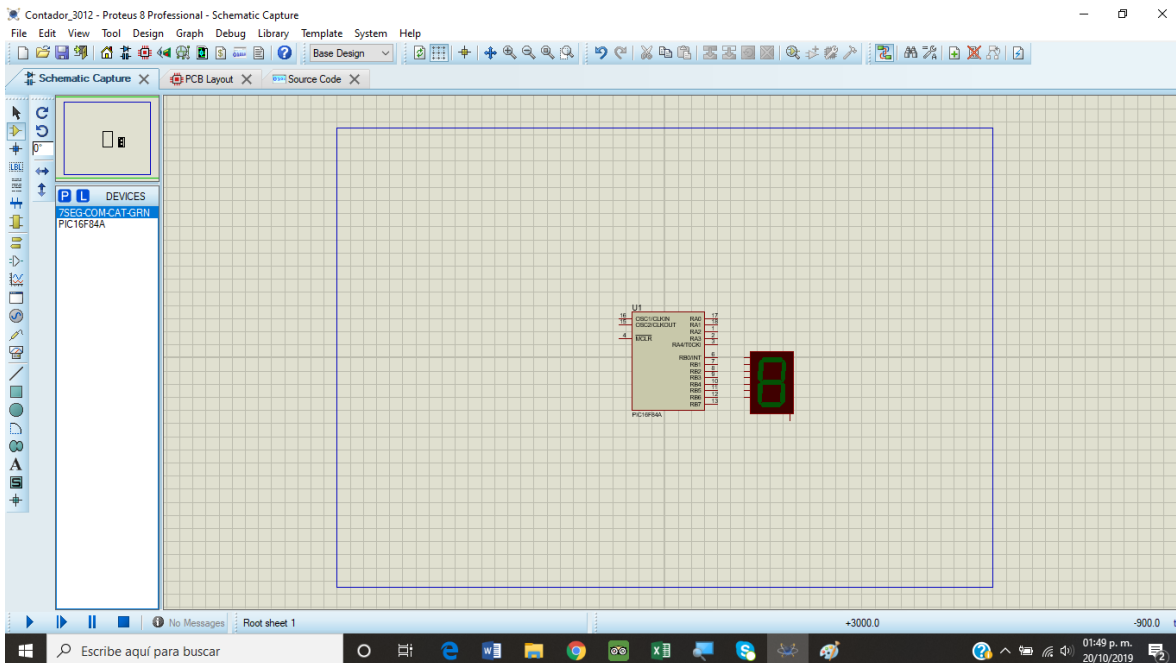
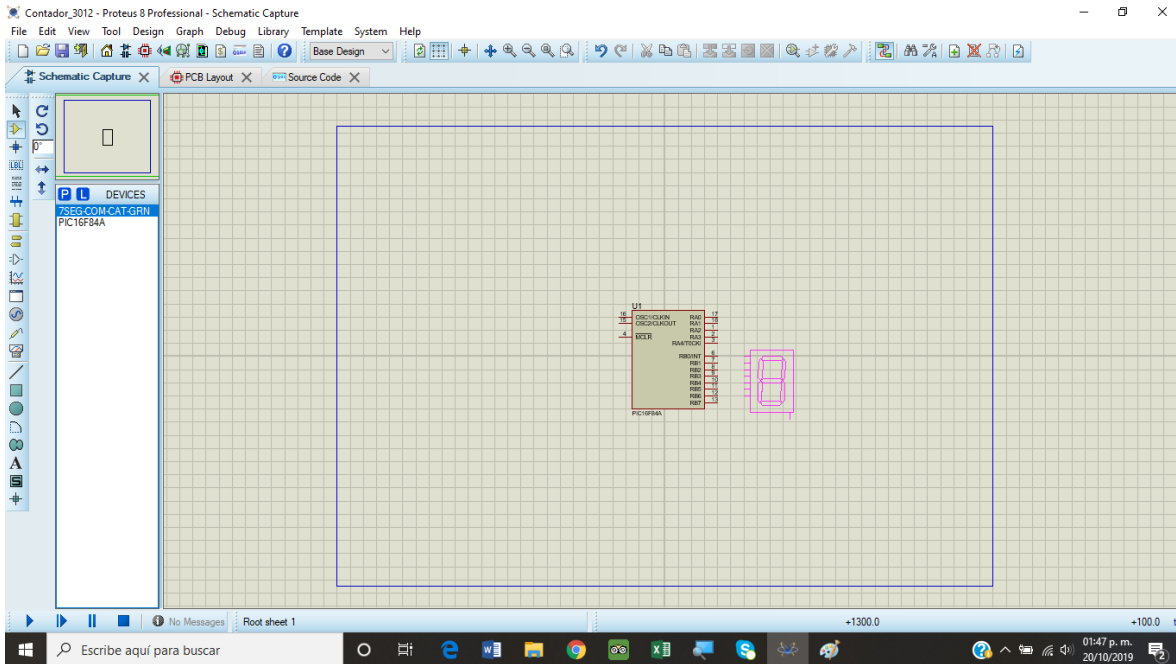


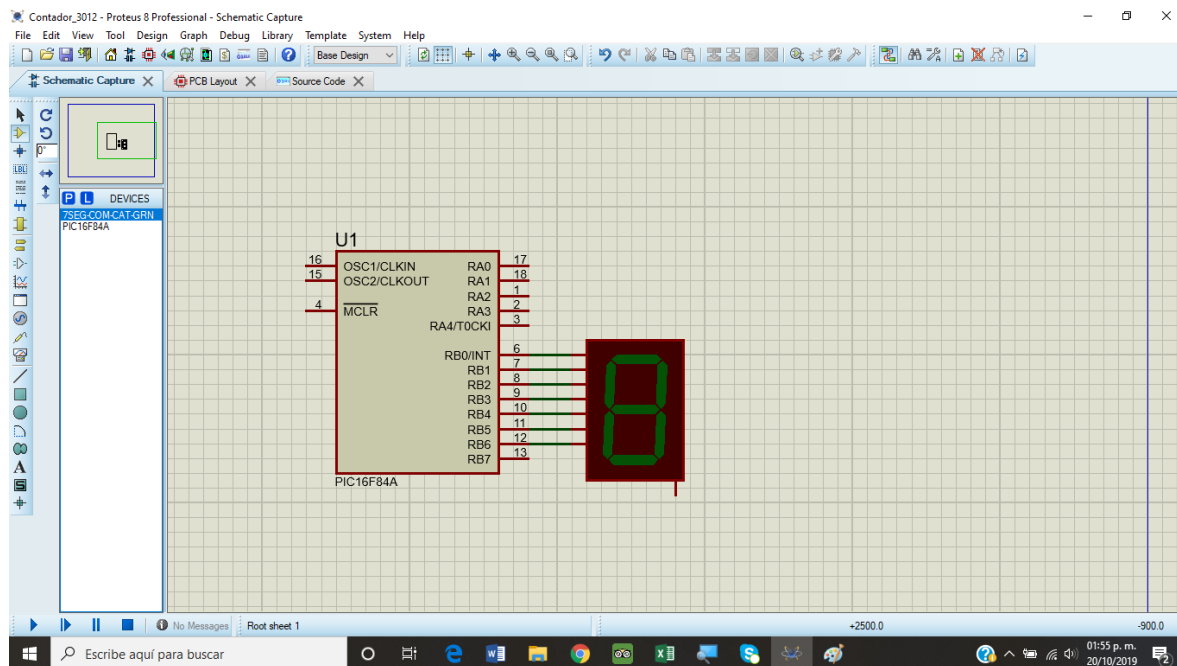
7.- Se elimina la flecha que sale del pin 4 dando click derecho, Delet Objec



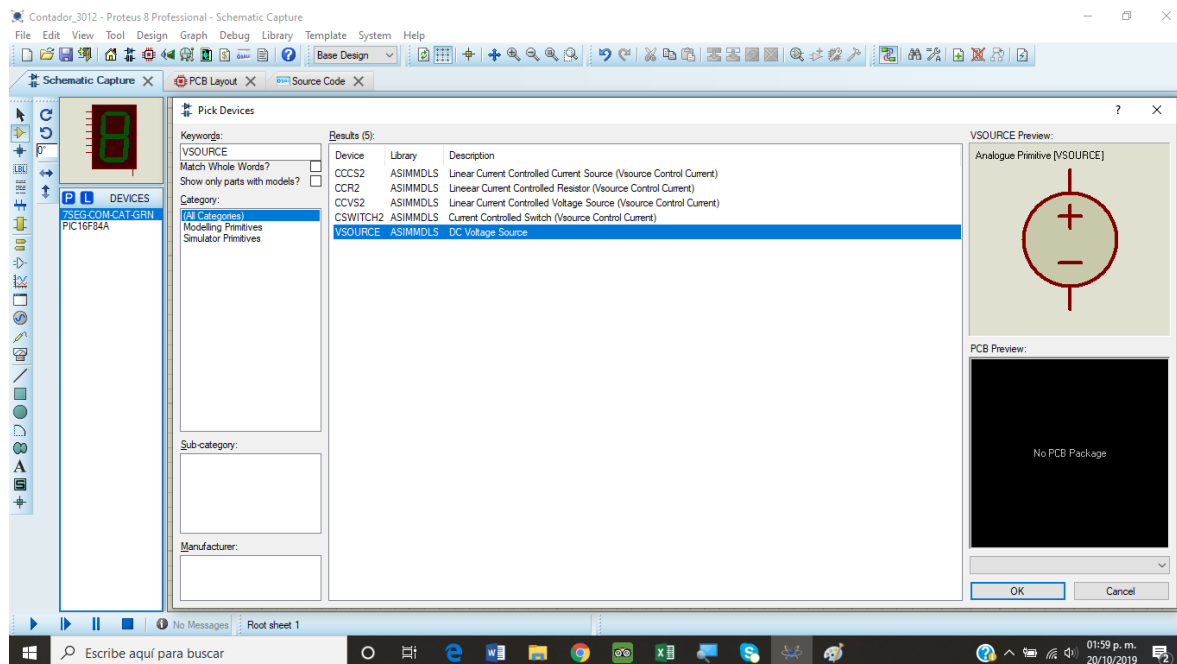
8.- Primero se cargara el Catodo comun de 7 segmentos.

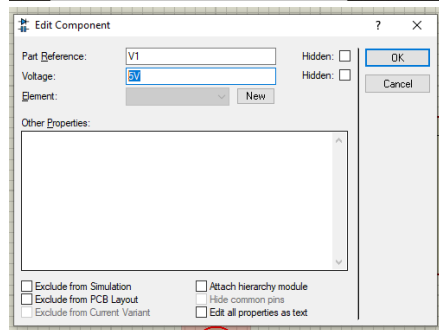
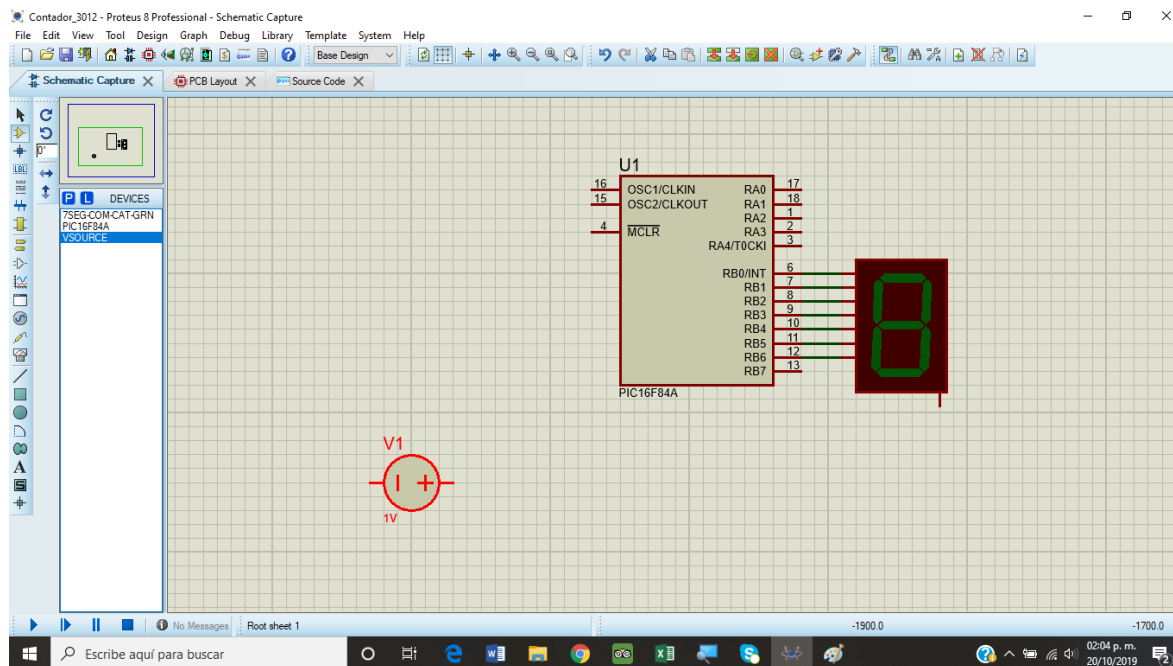






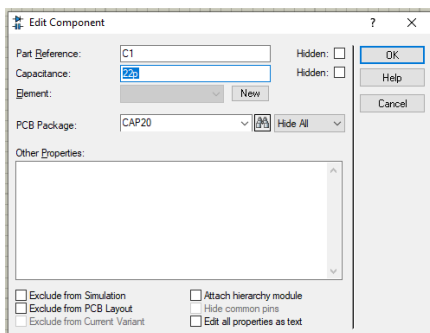
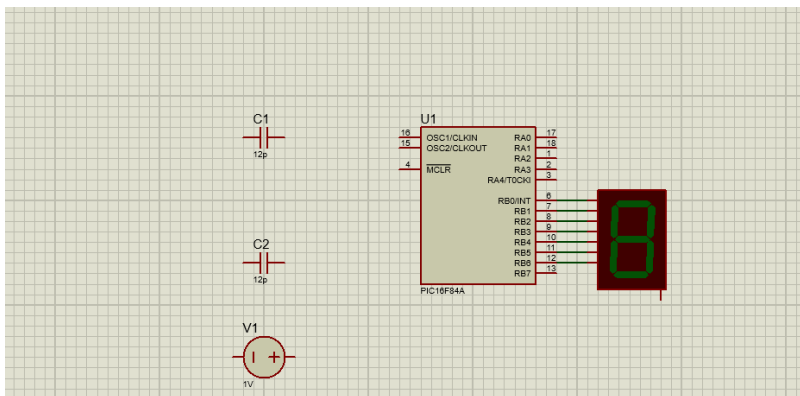
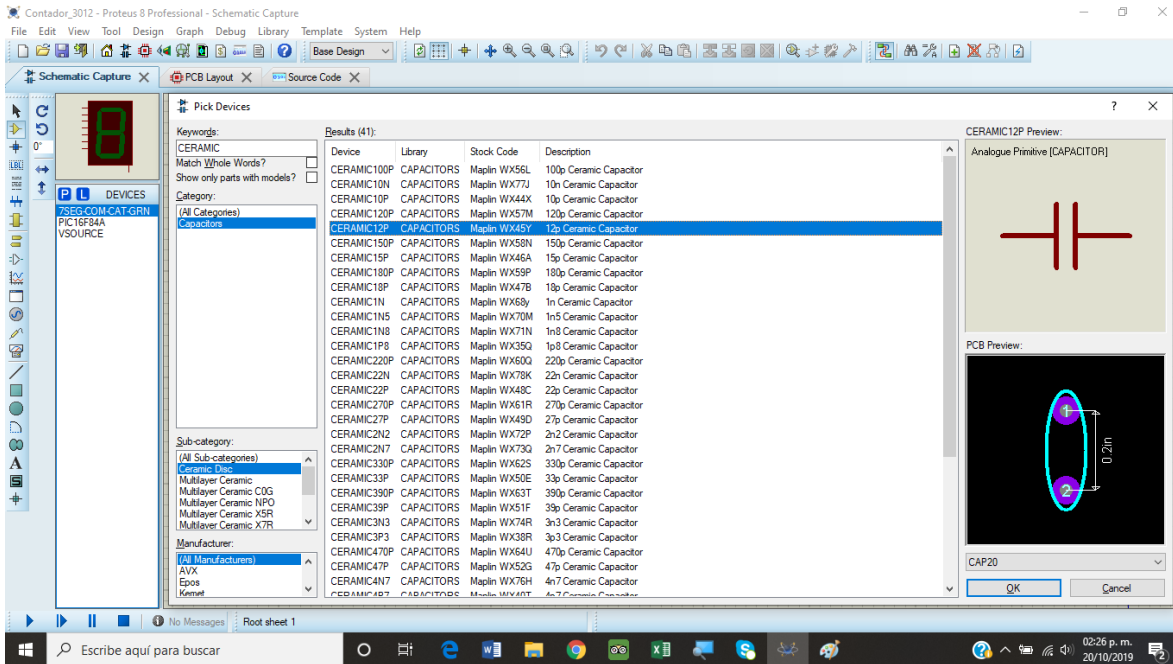
9.- Se cargara la pila de 5v.





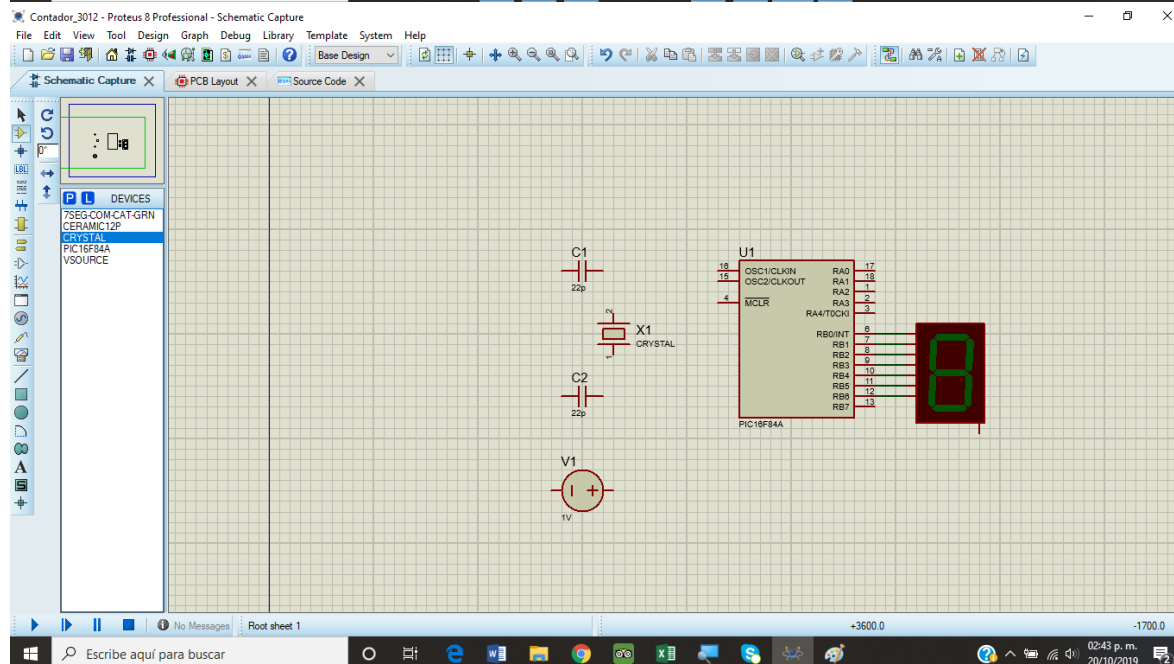
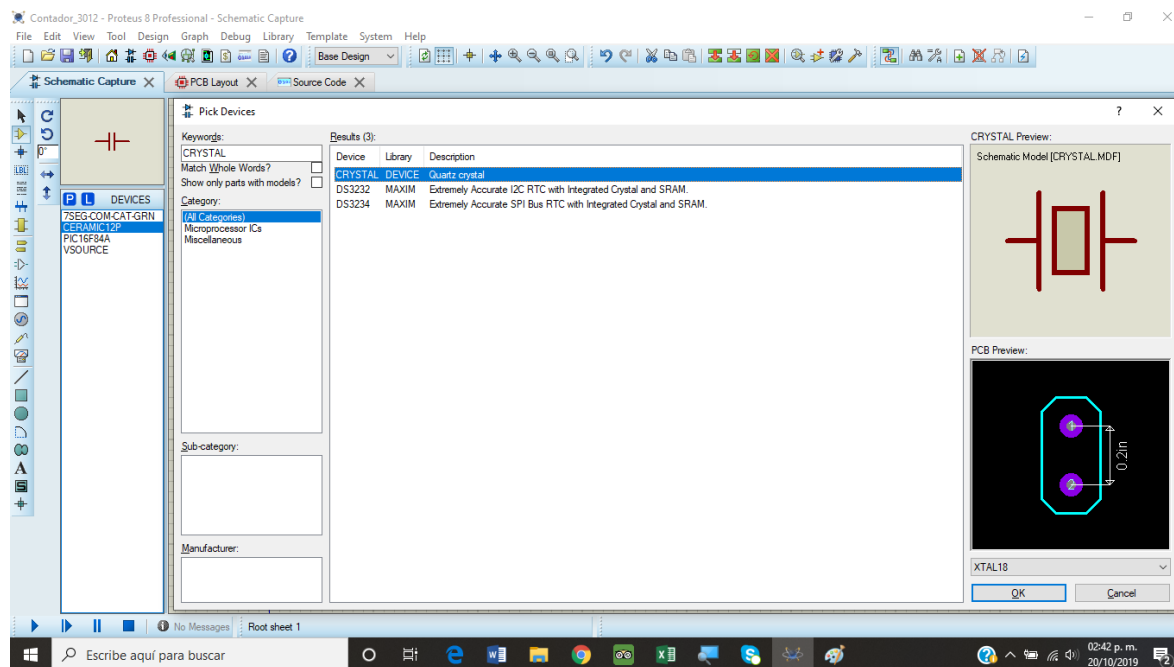
se cambia el voltaje de la pila

10.- Se cargan los capacitores de 22p



se cambia la capacidad por 22p en ambos capacitores.

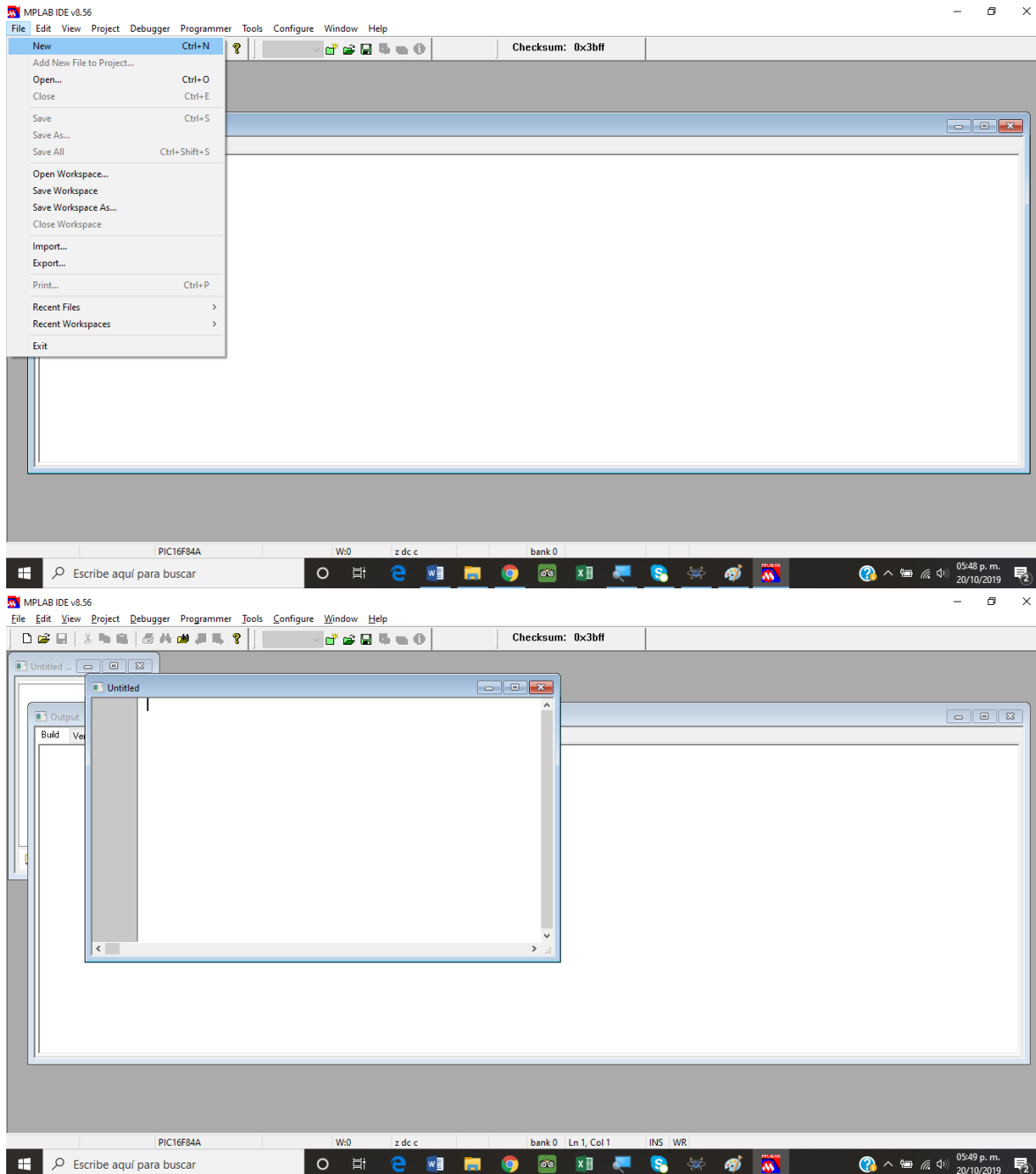
11.- Se cargara un Cristal osilador de 5MHZ,



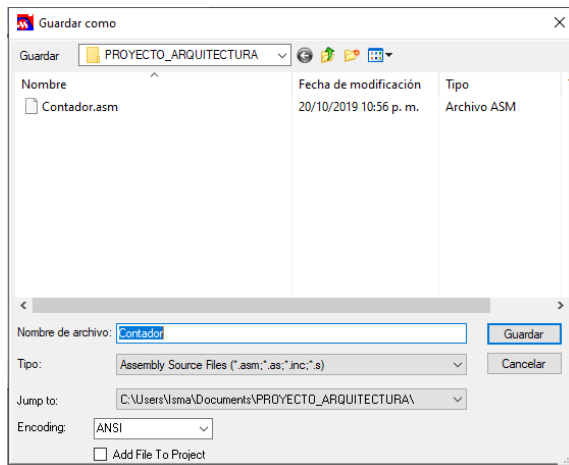
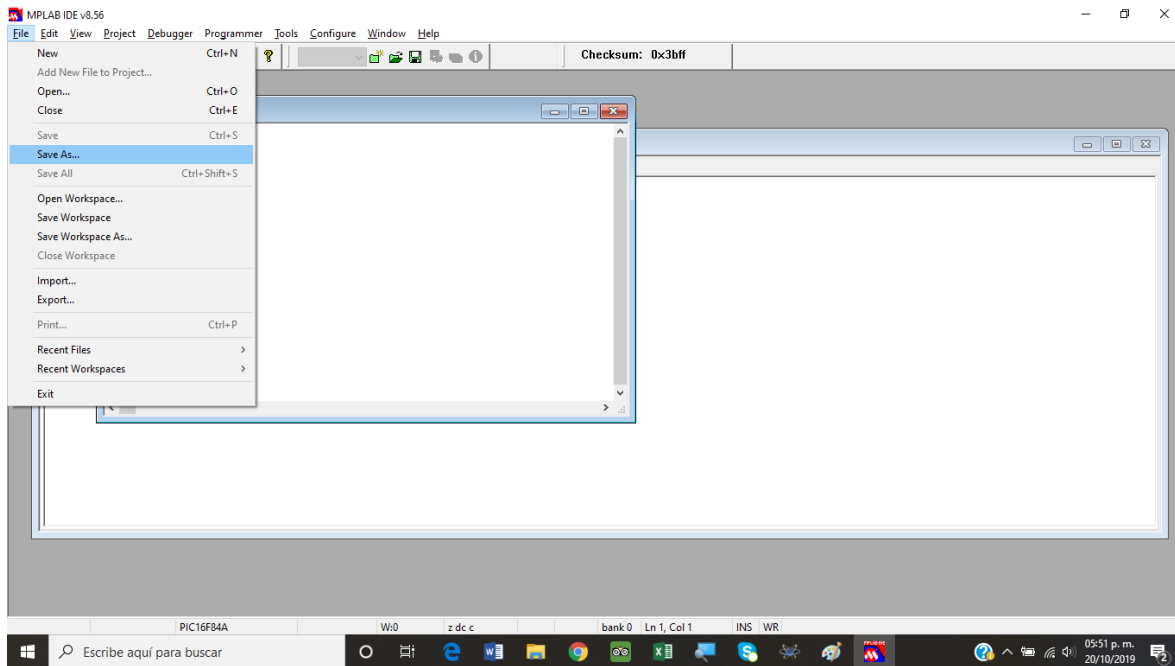


Se abre el programa MPLAB para crear el código.

13.- se inicia MPLAB y se crea un nuevo archivo.

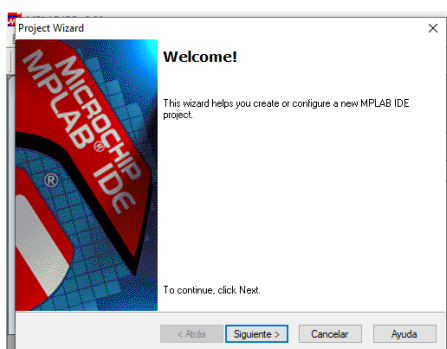
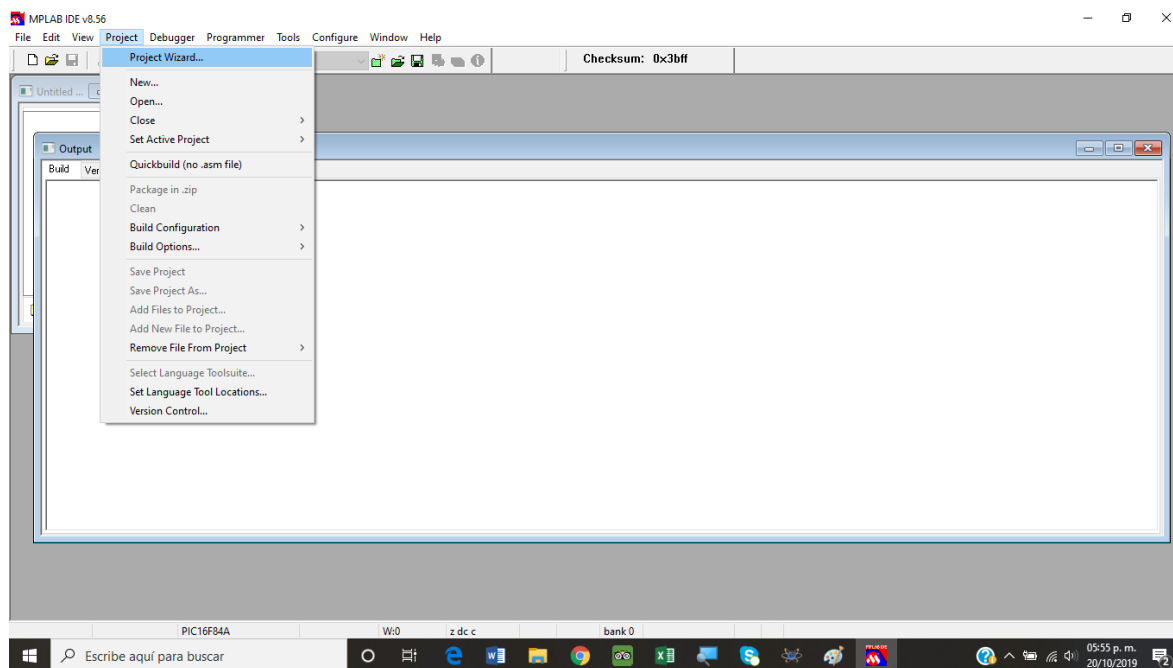


14.- Este se guardara con la extension “.asm”

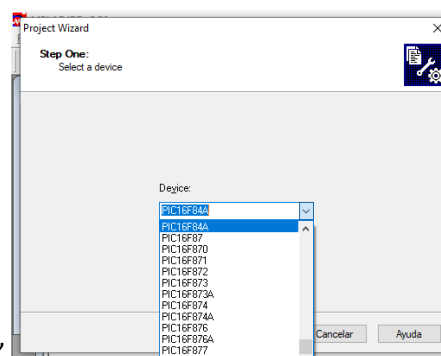


Se le nombra y guarda

15.- Se crea un nuevo proyecto.

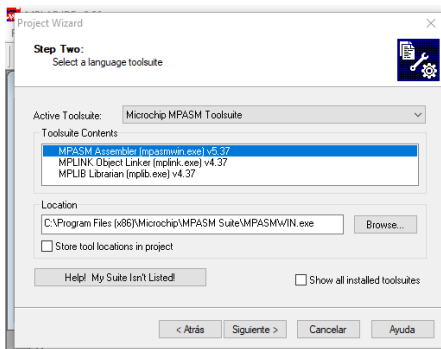


16.- Next

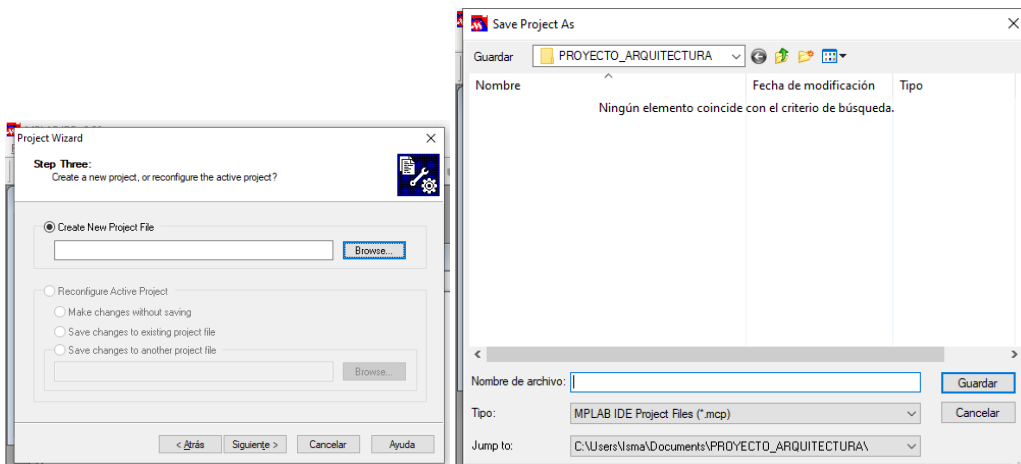


17.- Se elige el modelo del Pic que se usara "PIC16F84A"

18.- Next.

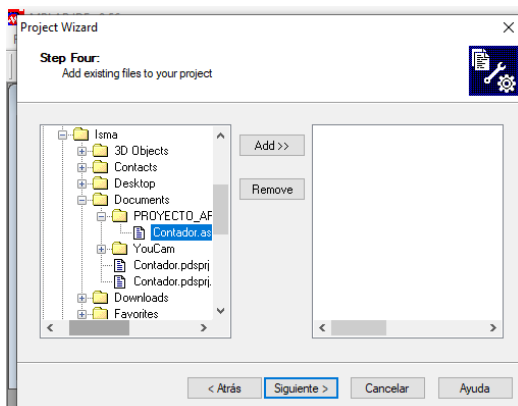


19.- Se da click en Bowser para guardar el proyecto en una carpeta y darle un nombre..

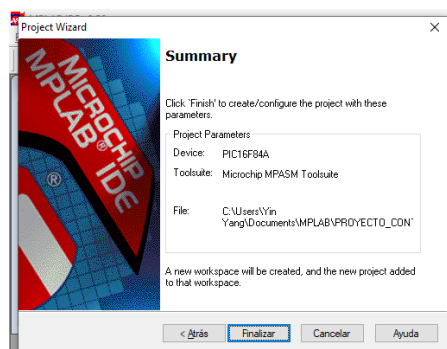


Click en siguiente.

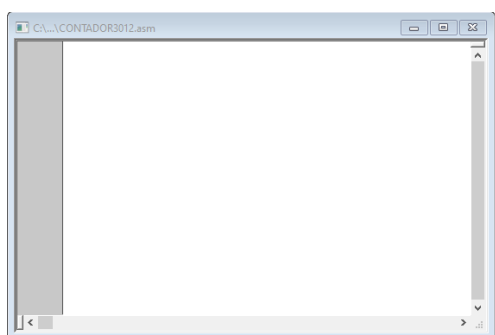
20.- Se busca el archivo que se guardo.



21.- finalizar.



En esta ventana es donde se pondra el codigo ensamblador.



22.- Se genera el codigo que hara funcional al PIC



```
MPLAB IDE v8.56 - [C:\Users\sma\Documents\PROYECTO_ARQUITECTURA\Contador.asm]
File Edit View Project Debugger Programmer Tools Configure Window Help

; ZONA DE DATOS *****
_CONFIG _CP_OFF & _WDT_OFF & _PWRTE_ON & _XT_OSC
LIST P=PIC16F84A
INCLUDE <P16F84A.INC>

; ZONA DE VARIABLE*****
CBLOCK 0x0C      ; SE INICIALIZA LA MEMORIA
NUMERO          ; VARIABLE QUE LLEVARA EL CONTADOR DE 0-9 Y A-F
CONTADOR        ; LLEVA EL TIEMPO EN CICLOS DE RELOJ
ENDC

ORG 0           ; INICIO EN 0
GOTO START     ; CICLO O BUCLE
ORG 5           ; FIN EN 5

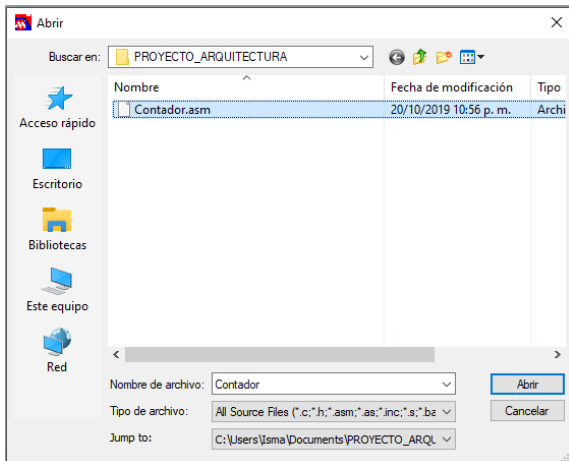
; CONFIGURACION*****
START          BSF STATUS,5      ; BANCO 1 ACTIVA EL BIT 5 DE F
               CLRW TRISE       ; PORTE ES LA SALIDA
               MOVWL 0x1F        ; MOVERA A LA PARTE BAJA DE REGISTRO DE MEMORIA RA0 RA4 SON LAS ENTRADAS
               MOVWF TRIISA      ; MOVER A F A TRIISA
               MOVWL B'11000111' ; ASIGNA 256 AL TIMER
               MOVWF OPTION_REG
               BSF STATUS,5      ; CARGAR EL CONTENIDO DE LA POSICION 5 AL BANCO 0
               CLRW             ; DEJA A W=0
               CLRWF NUMERO      ; LIMPIA LA VARIABLE NUMERO

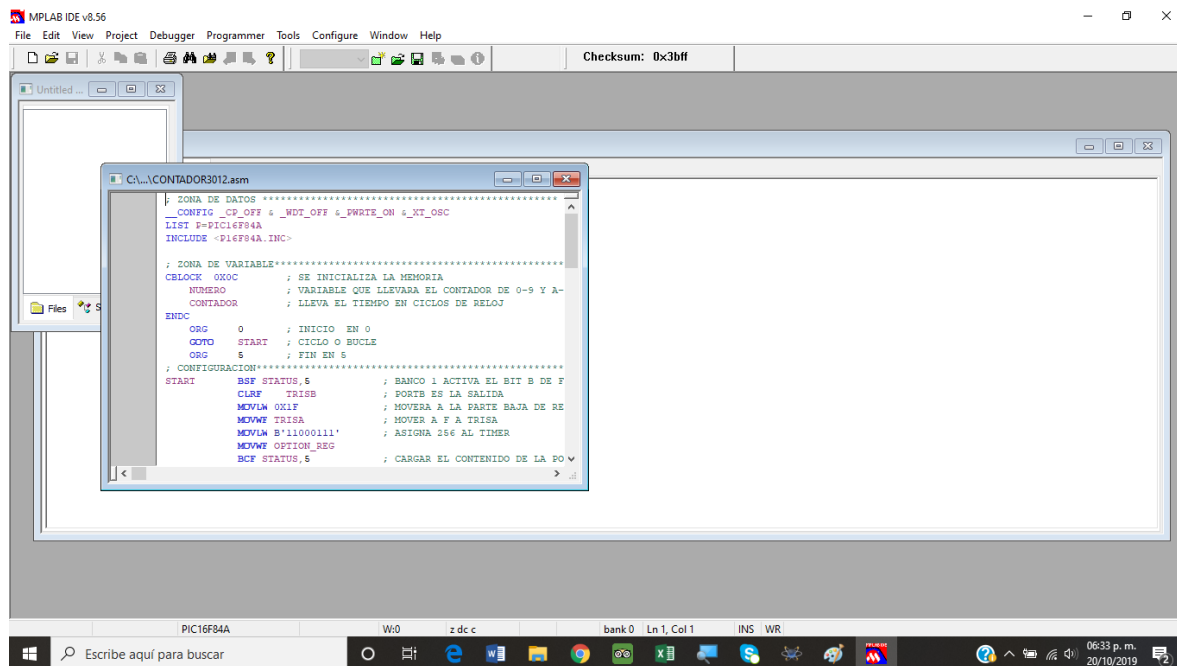
; Inicio*****
MAIN          MOVWF NUMERO,W     ; TOMA EL CONTENIDO DE W Y LO CARGA EN F
               CALL TABLA        ; LLAMA LA FUNCION TABLA
               MOVWF PORTB       ; MUESTRA EL VALOR QUE TOMO DE LA TABLA
               CALL PAUSE_1000   ; LLAMA A LA FUNCION PAUSE
               INCF NUMERO,F      ; REALIZA UN INCREMENTO DE LA VARIABLE EN 1
               MOVWF NUMERO,W    ; SE CARGA LA VARIABLE EN W
               XORLW 0x10        ; SE COMPARA SI ES QUE LLEGA AL REGISTR 10
               BTFSS STATUS,Z    ; VERIFICA SI HE KKEGADO
               GOTO MAIN        ; SE REGRESA AL MAIN
               CLRW             ; AL LLEGAR A 10 SE RESETEA A F
               CLRWF NUMERO      ; SE LIMPIA A NUMERO
               GOTO MAIN        ; SE REGRESA A MAIN

; FIN DE INICIO*****
; PAUSE_1000 SE REALIZA UN RETARDO DE 1 SEGUNDO*****
PAUSE_1000    MOVWL 0x02        ; SE LE ASIGNA 1000 AL CONTADOR
               MOVWF CONTADOR    ; SE MUEVE EL VALOR DEL CONTA DE A A F
               DELAY WZF INTCON TMR0 ; SE LLEVA EL BIT DE DESARROLAMIENTO TMR0

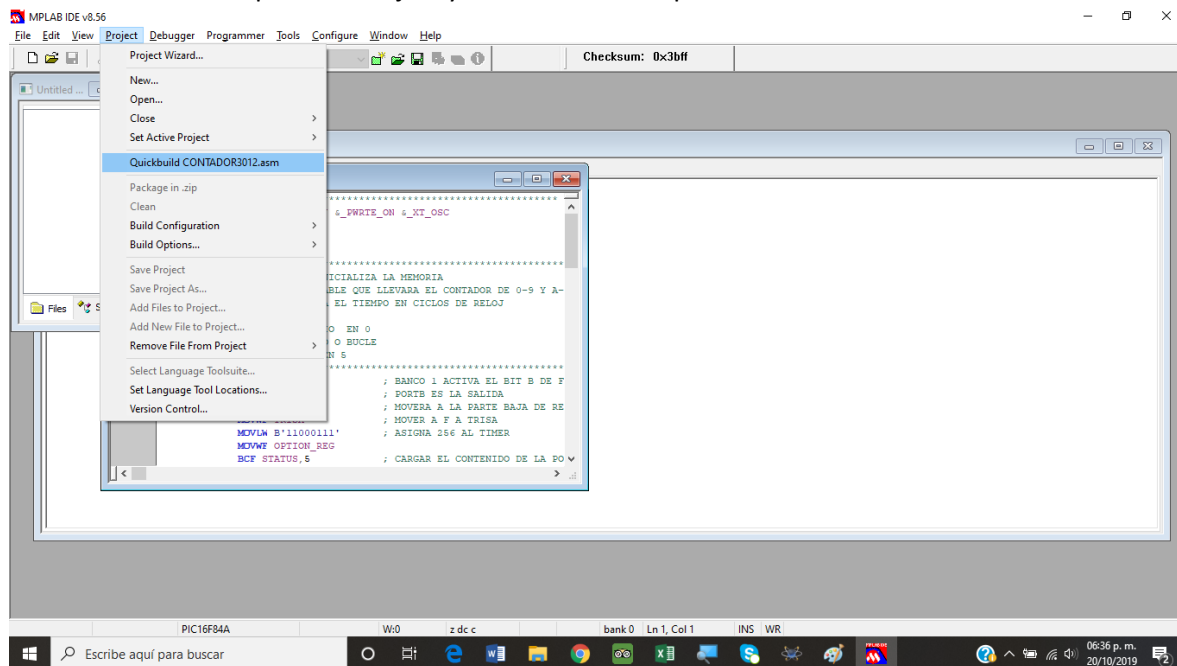
PIC16F84A      W:0      z d c      bank 0 Ln 68, Col 4      INS WR
```

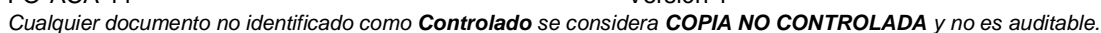
23.- Para compilarlo solo se debera abrir el archivo donde se escribio el codigo.

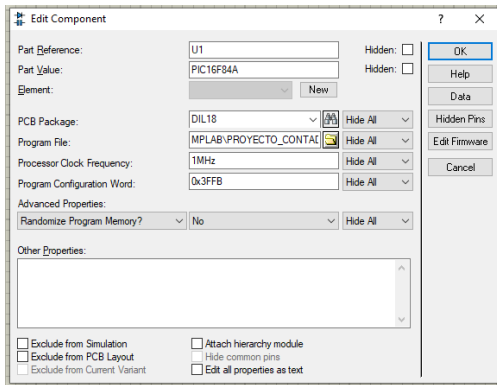




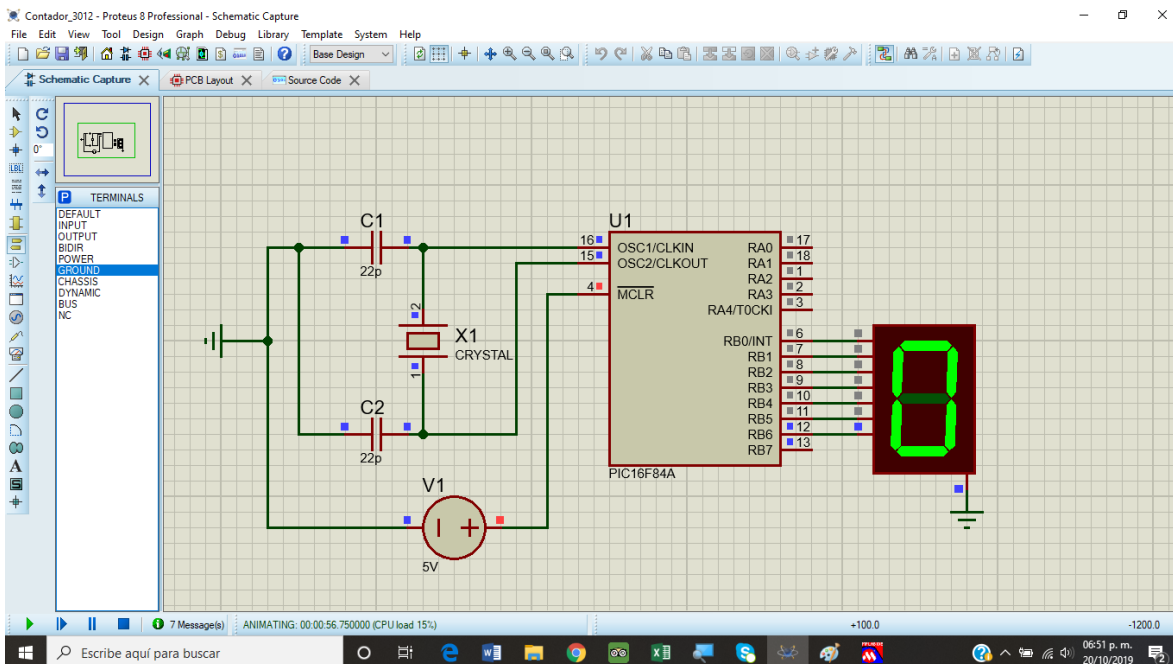
24.- Se da click en la pestaña Project y se selecciona la opción “Quickbuild” con el nombre del archivo.







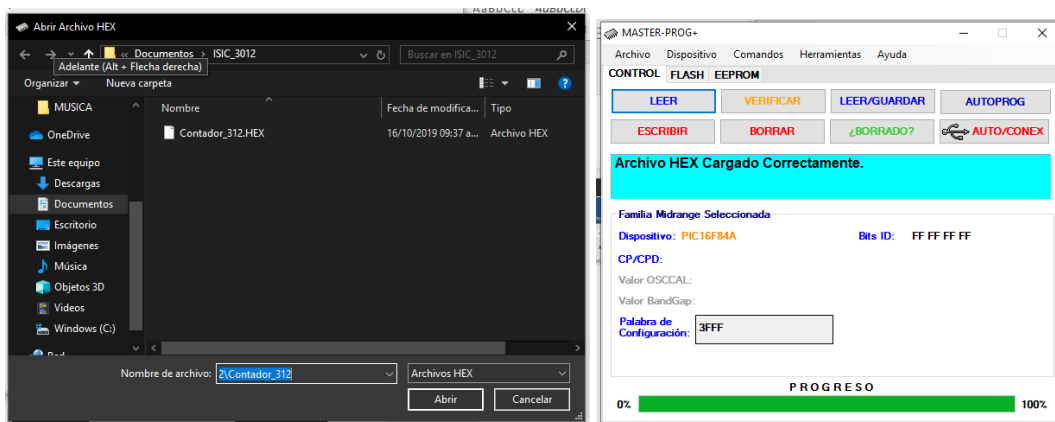
26.- Se ejecuta el programa dando clic en play.



1.- Se abre el programa y se espera a que se reconozca el Pic.



2.- Se abre el archivo .HEX



3.- Una vez abierto se Da click en la opcion Escribir..



5.- Se comprueba si se programo bien dando click en Verificar.



Los diagramas de abajo se usaran como guía para el ensamblaje.

Cátodo Común

