|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Contador Hexadecimal con PIC16F84A** | | | **No.** | **2** |
| **Asignatura:** | **Arquitectura de computadoras** | **Carrera:** | **Ing. Sistemas Computacionales** | **Duración de la práctica (Hrs)** | **5** |

**I. Competencia(s) específica(s):**

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

**Aula NIII**

**III. Material empleado:**

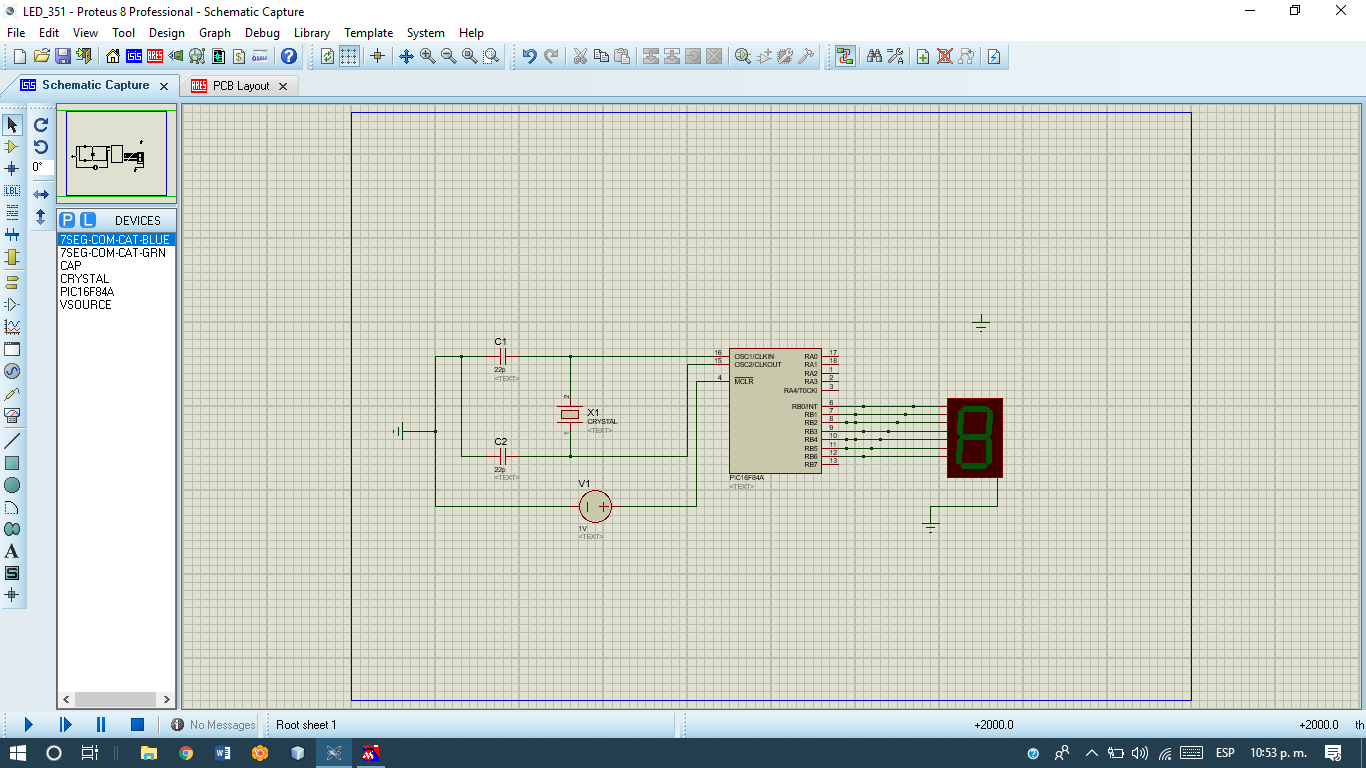
**Software MPLAP (programación en Ensamblador)**

**Software Proteus (para armar el circuito en la PC)**

**Programador de PIC y su software correspondiente**

**IV. Desarrollo de la práctica:**

Primero que nada, se instalaran en el equipo los softwares ya antes mencionados para realizar esta práctica, ya que se hallan instalado lo primero que se hizo fue unir el circuito en el software Proteus para después meterle el programa del contador:



En el circuito se utilizan los siguientes componentes:

* Un cristal oscilador de 4
* Dos cerámicos de 22 picofaradios
* Una fuente de voltaje de 5 v
* Un cátodo común de 7 segmentos
* Un PIC16F84A

El cátodo común de 7 segmentos se conectará al PIC iniciando por la salida RB0 hasta llegar a la salida RB6

En la entrada del PIC se conectará el uno de los lados del cristal oscilador y después uno de los cerámicos para hacer que tenga contacto con la fuente de energía.

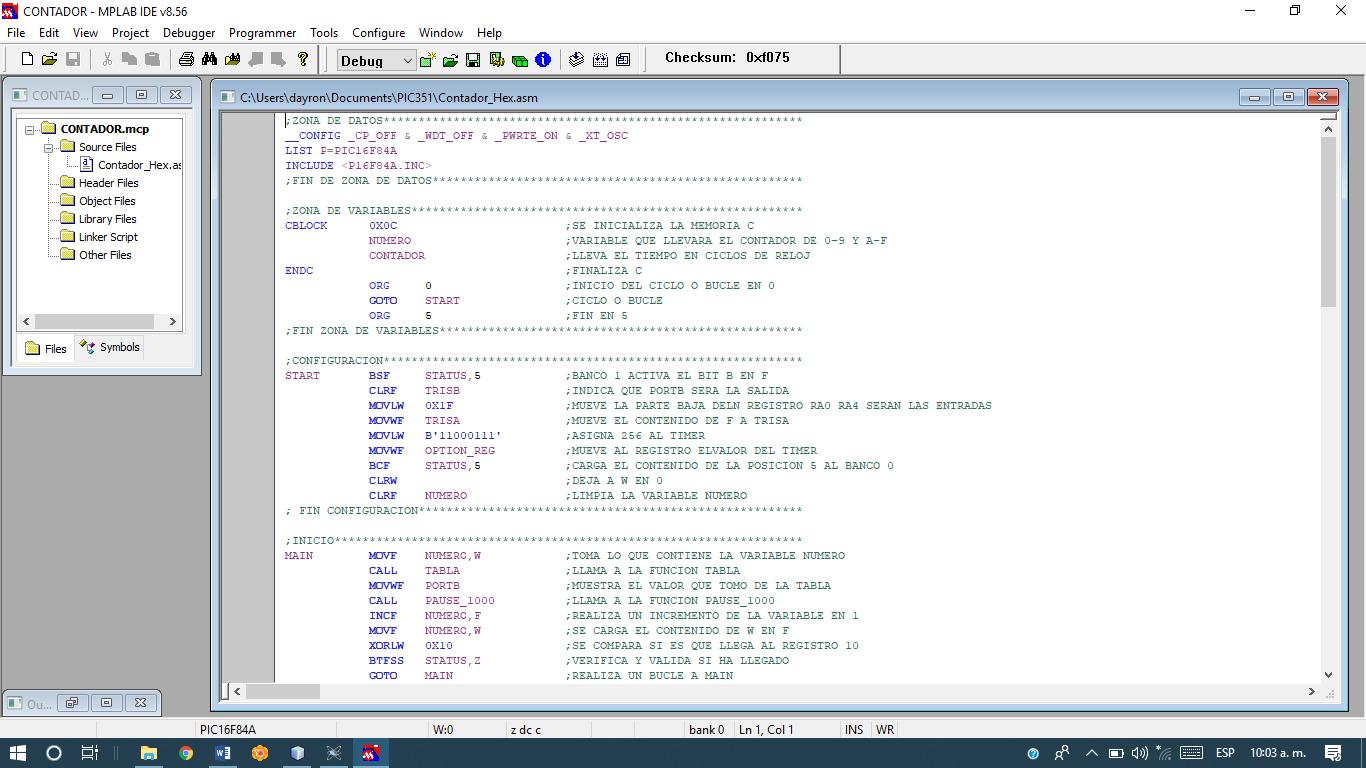
Y por la salida se hará de la misma manera y después se interceptará con la misma línea de voltaje a la que va el primer cerámico.

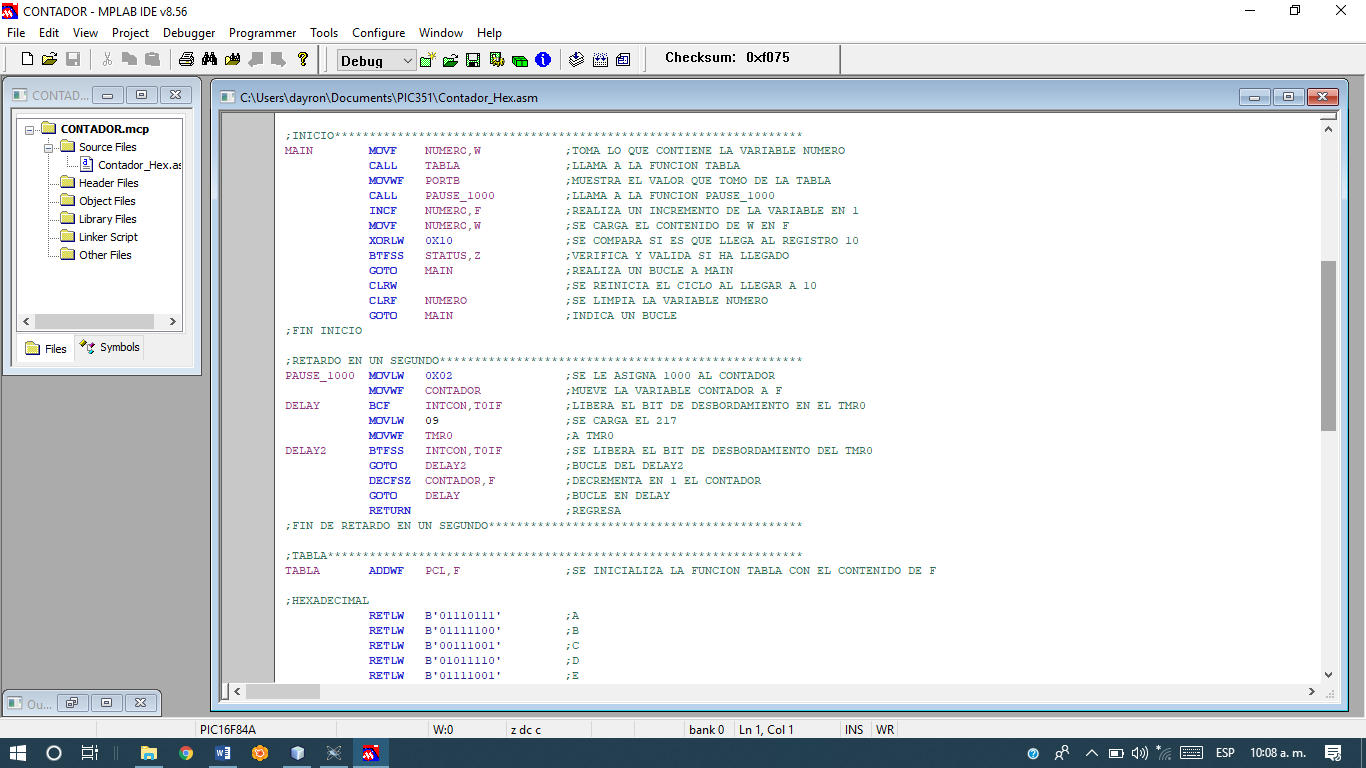
Y por último se conectará la fuente de energía al pin 4.

Ya que tenemos el circuito ensamblado tocara realizar el código en lenguaje ensamblador para poder meterlo al PIC.

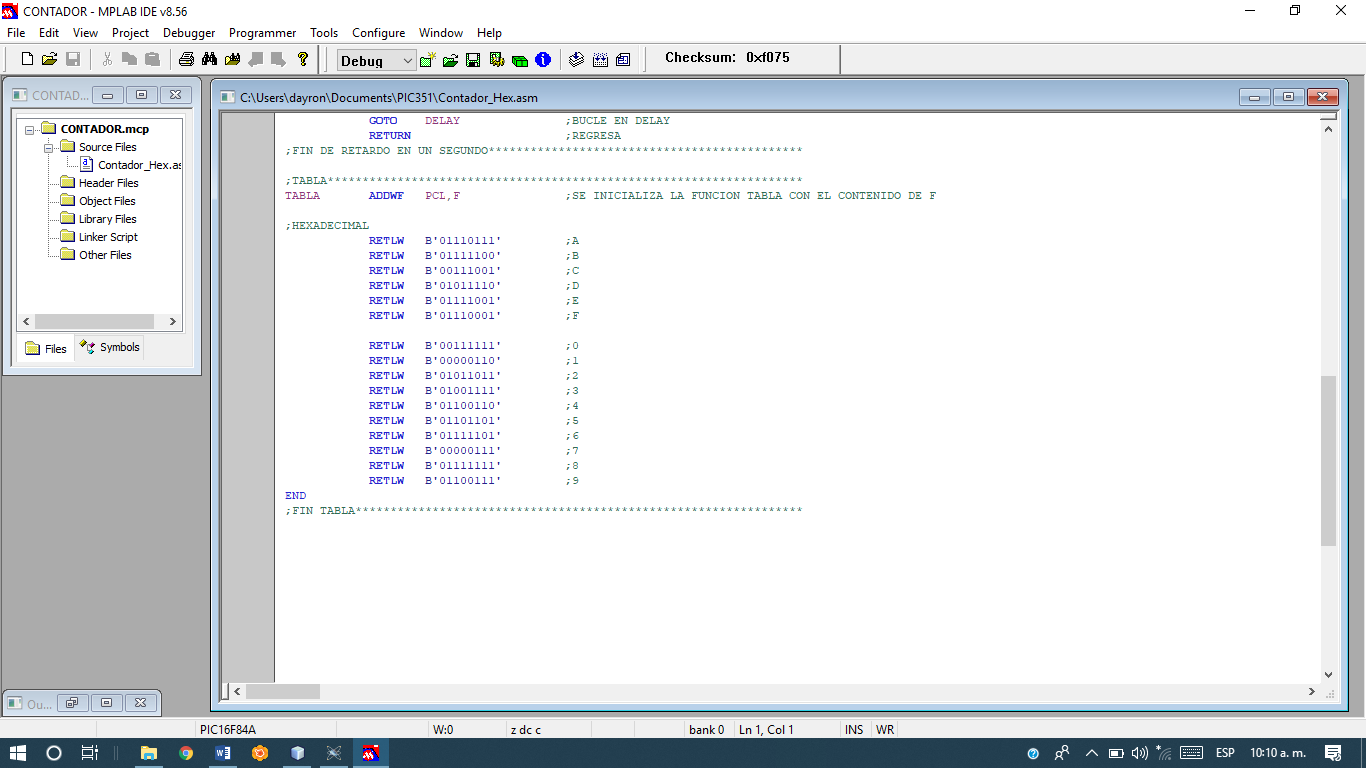
Para esto se utilizará el software MPLAP.

Se creará un nuevo proyecto y se empezara a hacer la codificación cada una de las las líneas de código viene explicada su función en la misma impresión de pantalla.



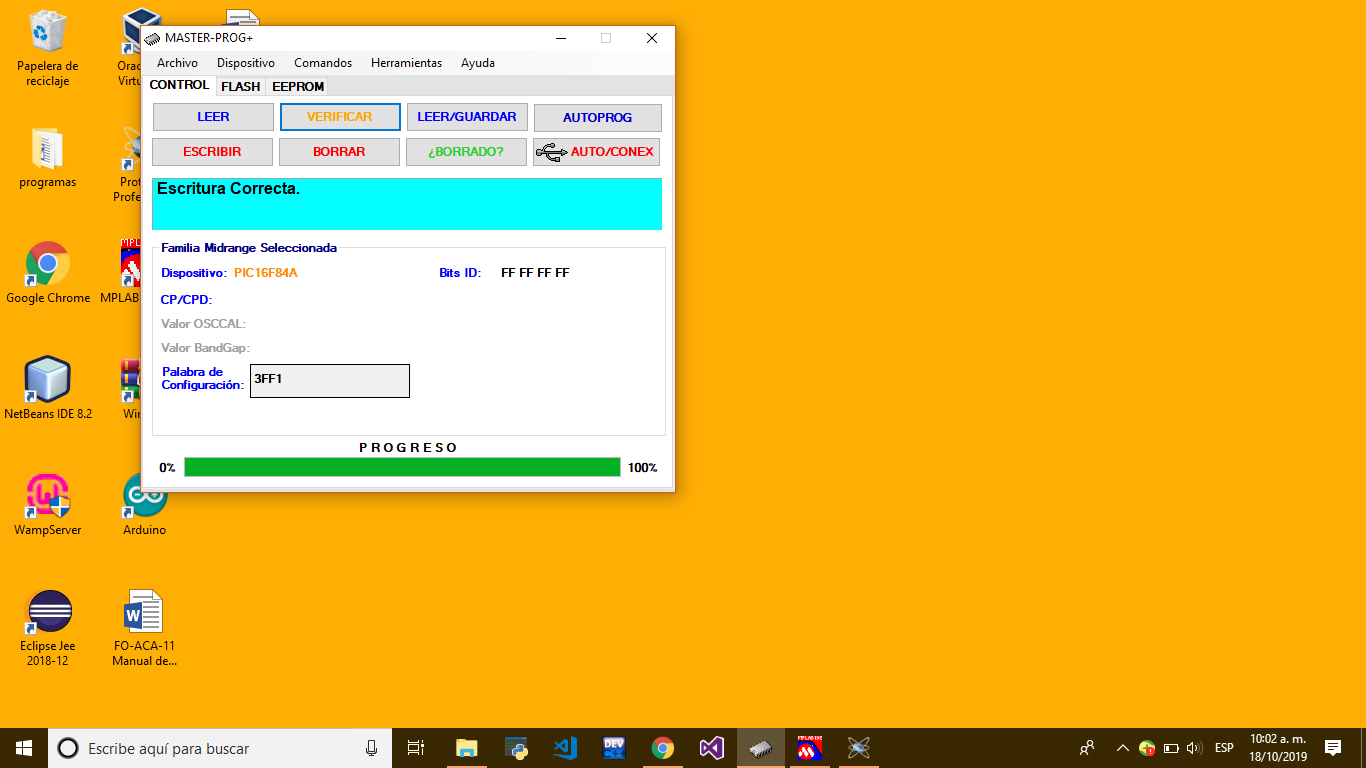


En esta última parte no es más que la declaración de las salidas de los números y letras en binario, pero no en conversión si no en posición del segmento que se quiera encender tomando en cuenta que el segmento 1 es el último número en binario leyendo de izquierda a derecha.

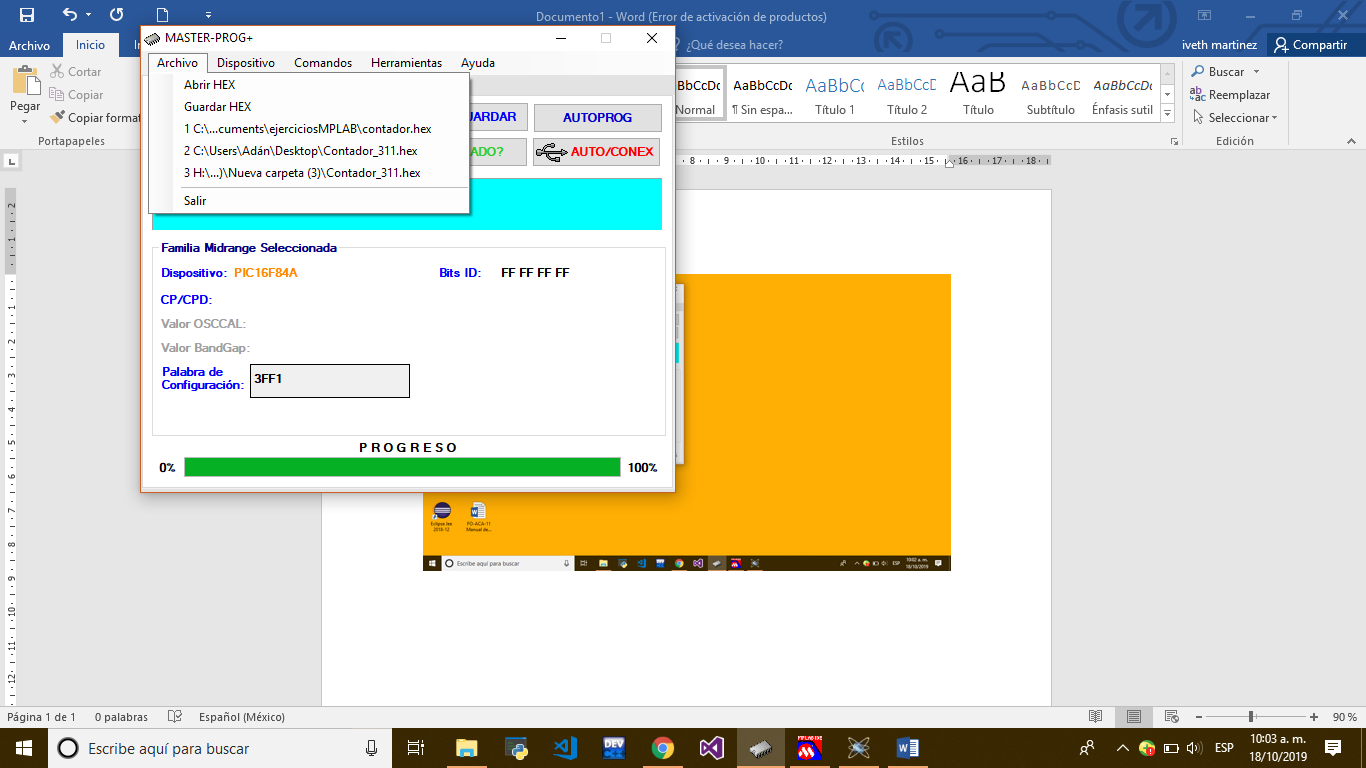
P

En la siguiente parte se mostrara como se cargó el programa ya antes mencionado del ensamblador con el programador de PIC

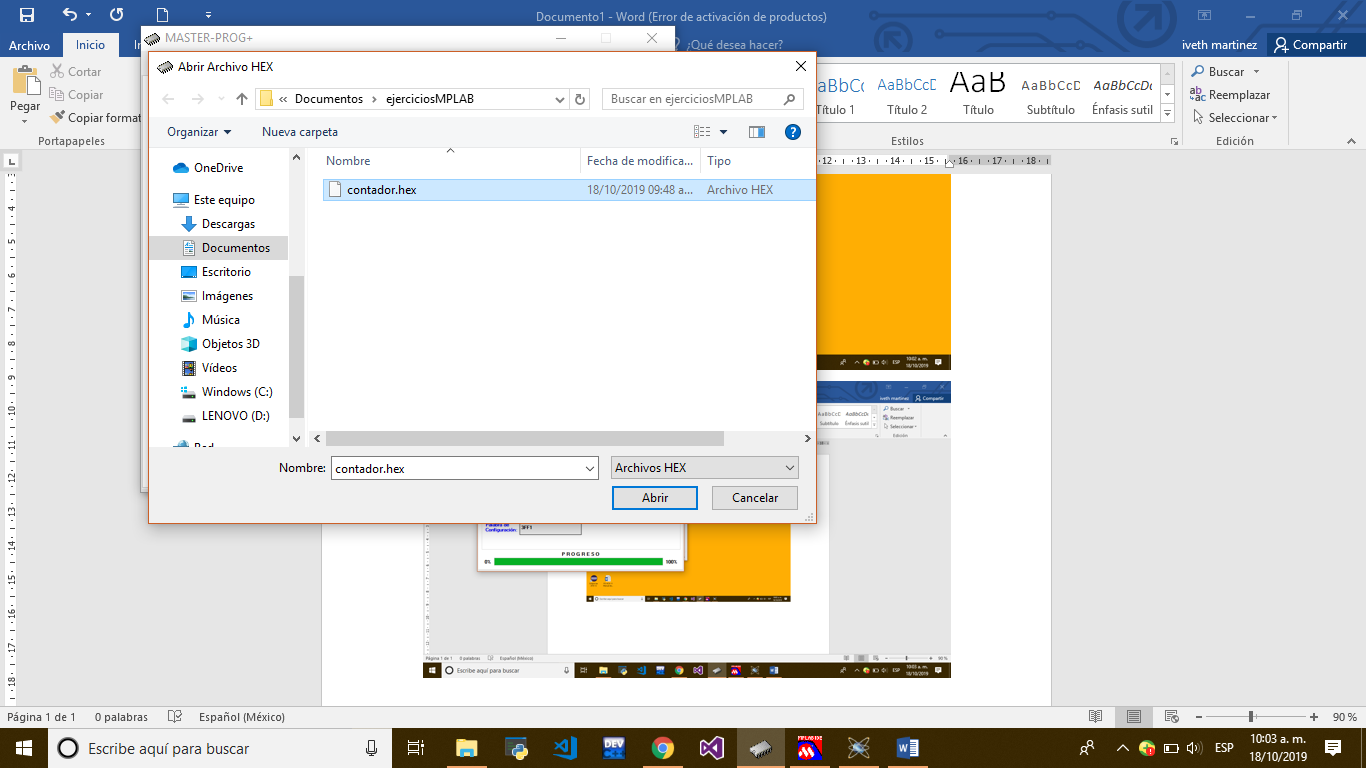
Primero se inserta el PIC en el programador y dar verificar para que detecte que tipo de PIC es:



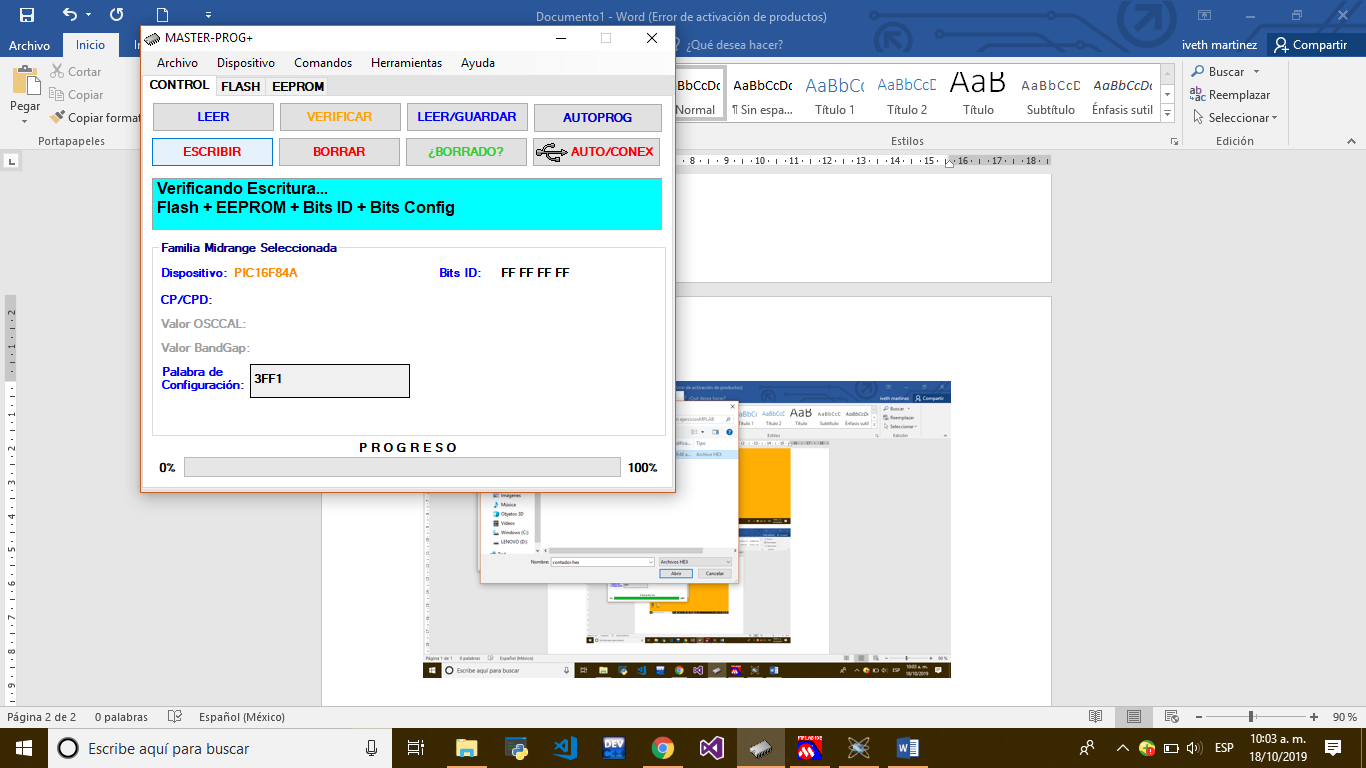
Ya que se detectó del PIC se procederá a cargar el archivo .HEXE que contiene el programa hecho en lenguaje ensamblador, se abre la pestaña archivo y abrir HEXE.



Se elige el archivo que contiene el código:



Se le da abrir y después se le pica en el botón de escribir y de esta forma se cargara el código en el PIC que se insertó.



Cuando el progreso llegue a 100% el código se habrá cargado en el PIC y se podrá proseguir con el siguiente.

**V. Conclusiones:**

**En cuanto el armado en Proteus es bastante no es tan fácil como parece cada componente tiene un papel demasiado importante para el funcionamiento del circuito.**

**El programado en el lenguaje ensamblador es bastante confuso si lo quieres entenderlo como cualquier otro lenguaje, pero es bastante bueno y te ayuda a ver la programación mas a fondo y como se mueve la memoria en tu programa.**