

Practica 1: Modelo de Programación del Microcontrolador MC68HC12

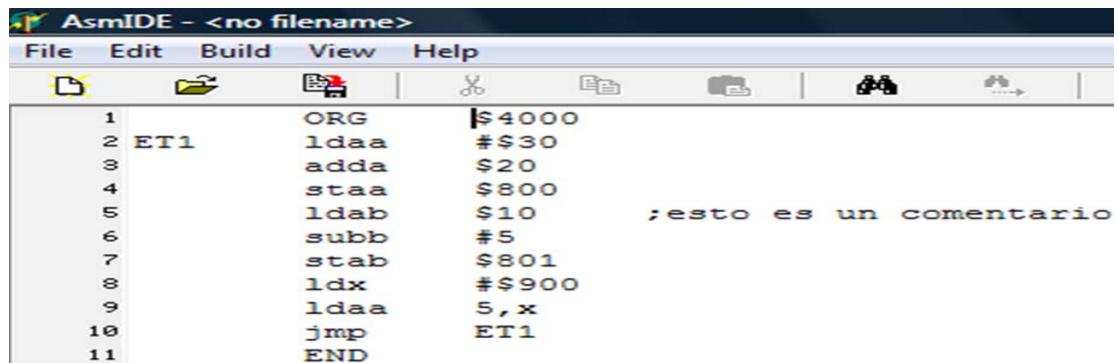
1.2 Modelo de programación de la arquitectura de computadora.

OBJETIVO DEL TEMA. Identificar el modelo de programación de la arquitectura, analizando la arquitectura de la máquina

Descargar el archivo HC12.zip del grupo electrónico de yahoo, de la siguiente liga:
http://mx.groups.yahoo.com/group/t_prog_sis/files/

Este archivo se descomprime y a continuación ejecutamos el archivo ASMIDE.EXE que es un entorno de programación para el microcontrolador HC12.


Capturar las instrucciones que se muestran respetando el formato.



```
1      ORG      $4000
2  ET1  ldaa     #$30
3      adda     $20
4      staa     $800
5      ldab     $10      ;esto es un comentario
6      subb     #5
7      stab     $801
8      ldx      #$900
9      ldaa     5,x
10     jmp      ET1
11     END
```

Se debe guardar utilizando como nombre el numero de codigo de alumno y con extensión “.asm”.

Al momento de guardar deberán cambiar los colores de la fuente dependiendo del campo al que corresponde.

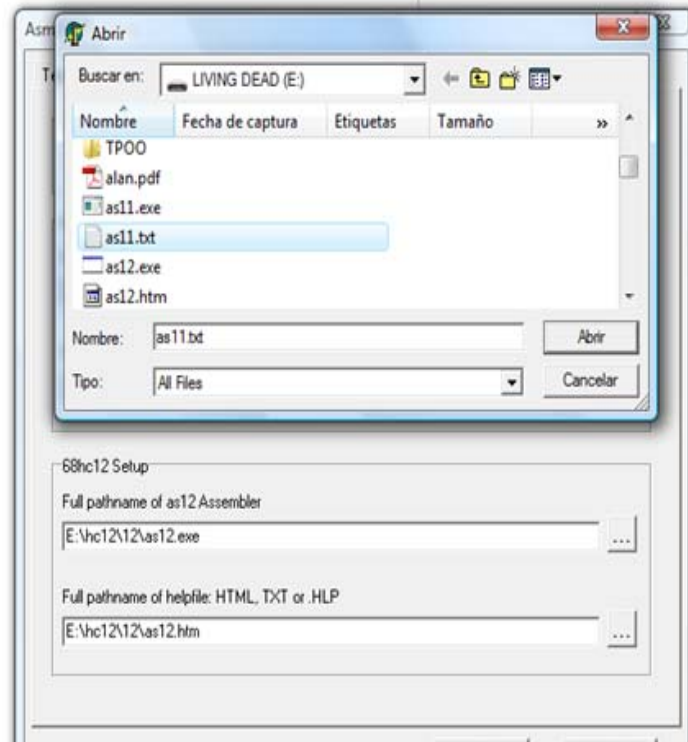


```
1      ORG      $4000
2  ET1  ldaa     #$30
3      adda     $20
4      staa     $800
5      ldab     $10      ;esto es un comentario
6      subb     #5
7      stab     $801
8      ldx      #$900
9      ldaa     5,x
10     jmp      ET1
11     END
```

Una vez guardado se ensambla se da clic en los engranes para depurar

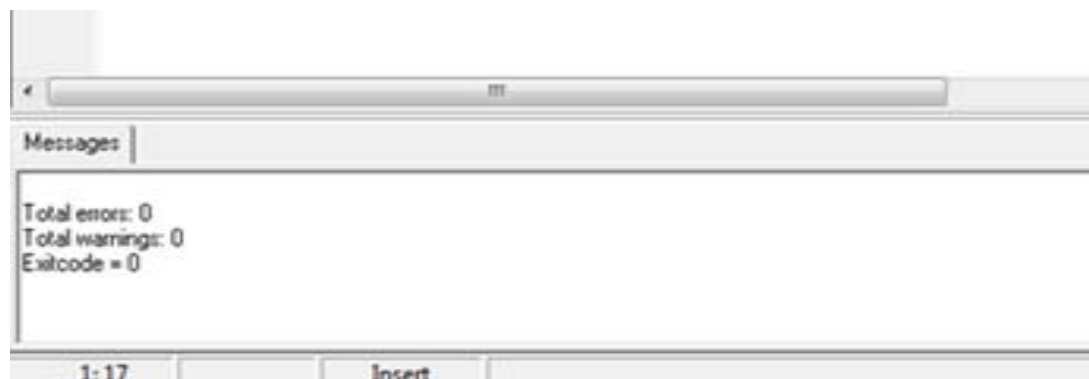
Es posible que se requiera configurar el entorno, para ello haremos lo siguiente:
Abrir el Menú VIEW/OPTIONS y la pestaña ASAMBLE

```
ORG    $4000
ET1    ldaa    #$30
      adda    $20
      staa    $800
      ldab    $10    ;esto es un comentario
      subb    #5
      stab    $801
      ldx     #$900
      ldaa    5,x
      jmp     ET1
      END
```



Verificar que se encuentre activa la opción 68HC12 , asegurar que las rutas de los archivos de setup sean correctas.

En la parte baja (area de Messages) aparecerán el numero total de errores, warnings y el código de salida, cuando la sintaxis y las instrucciones sea correcta deberá marcar 0.



Se generan 2 archivos con extensión LST y otro con extensión S19.

El archivo LST contiene los códigos maquina de las instrucciones generadas.

```
303234509P1.lst

as12 (v1.2d) Absolute assembler for the Motorola CPU12 (freeware)
Executed: Wed Sep 02 07:40:55 2009

4000                                ORG    $4000
4000 86 30                        ET1    ldaa  #$30
4002 9b 20                        adda   $20
4004 7a 08 00                      staa   $800
4007 d6 10                        ldab   $10    ;esto es un comentario
4009 c0 05                        subb   #5
400b 7b 08 01                      stab   $801
400e ce 09 00                      ldx    #$900
4011 a6 05                        ldaa   5,x
4013 06 40 00                      ET1

Total errors: 0
Total warnings: 0
```

Contador de localidades (CONLOC)

Código de Operación (cop)

Operando (op)

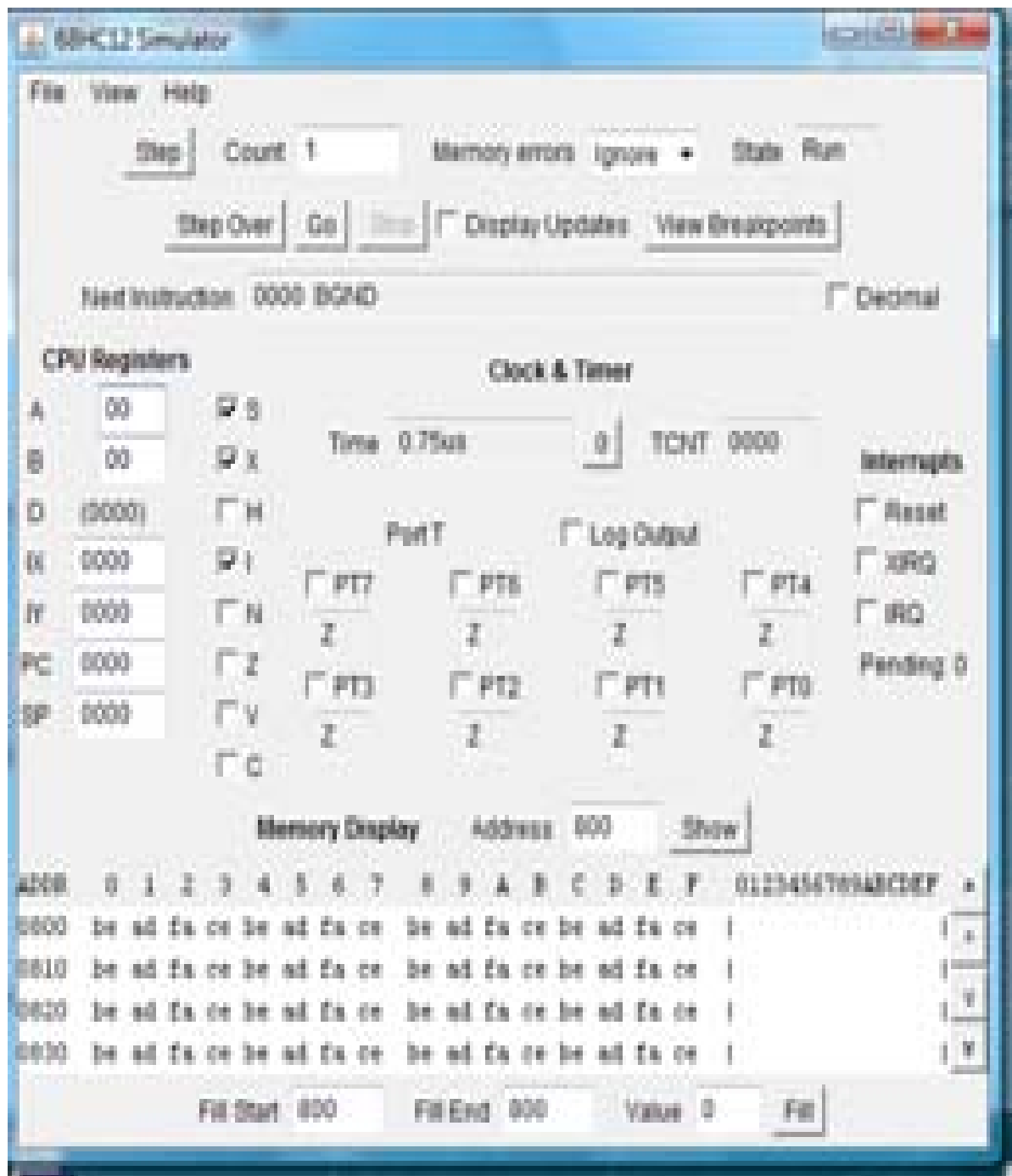
Los códigos maquina y el contador de localidades se encuentran en base hexadecimal.

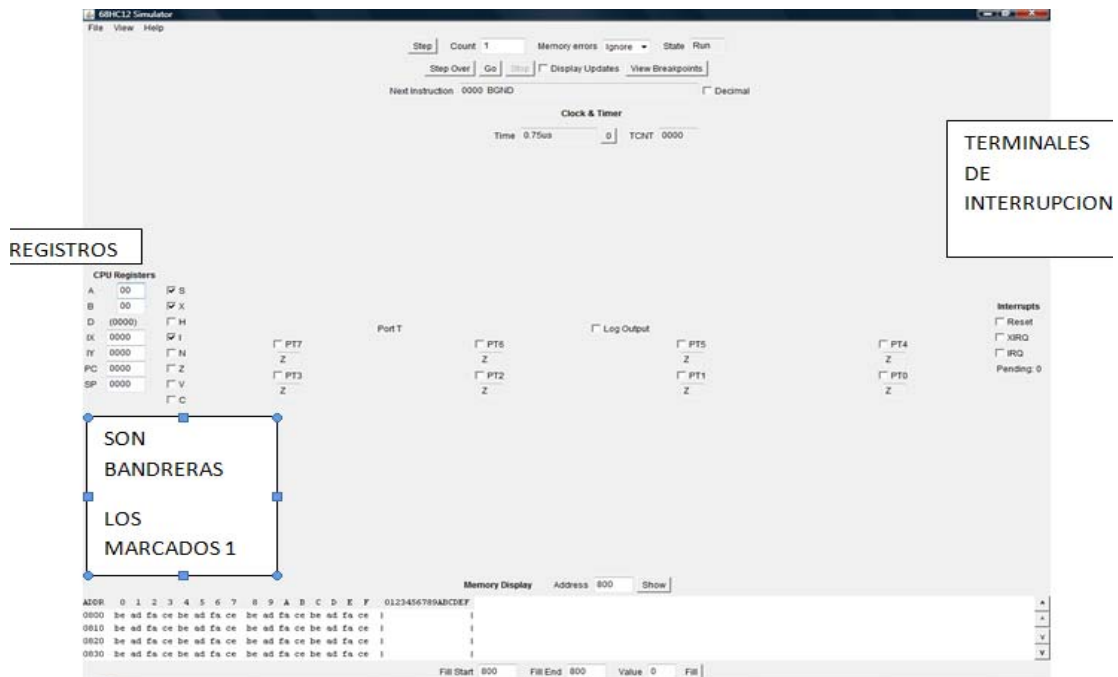
El archivo con extensión S19 es el archivo objeto que se utilizara en el simulador y para programar el microcontrolador. Este archivo se encuentra en un formato especial para los archivos objeto denominado SRECORD propietario de Motorola.

```
S01D000042696E6172792043726561746F723A20453A5C617331322E657816
S01D000065200A436F6D6D616E64204C696E653A200A46696C653A2033304B
S01100003332333435303950312E41534D0AEA
S113400086309B207A0800D610C0057B0801CE09B3
S109401000A605064000B5
S9030000FC
```

Después de esto el siguiente paso es ejecutar el simulador en donde podemos observar los elementos internos del microcontrolador.

El simulador es un programa elaborado en java, lo ejecutamos con la siguiente instrucción.
 Runsim.bat.
 Es necesario tener instalado la maquina virtual de java.





Para simular el programa que acabamos de escribir es necesario utilizar el menu file/load y abrir el programa con extensión s19

