1. **El compilador deberá reconocer todos los mnemónicos del set de instrucciones del MC68HC11 tanto en letras mayúsculas como minúsculas.**
2. **El compilador deberá reconocer la sintaxis correspondiente a cada uno de los seis modos de direccionamiento que soporta el CPcU de dicho microcontrolador.**
3. **El compilador deberá soportar archivos documentados con comentarios, los cuales tendrá que ignorar.**
4. **El compilador deberá reconocer las directivas de ensamblador “ORG”, “EQU”, FCB” y “END”.**
5. **El compilador será capaz de abrir un archivo de texto codificado en ANSI, con extensión “\*.asc”, que contenga el código fuente en lenguaje ensamblador.**
6. **El compilador hará un análisis del código fuente, línea por línea, para determinar el código objeto correspondiente.**
7. **El compilador generará un archivo de texto con extensión “\*.LST” que contenga el código fuente y el código objeto correspondiente empleando el formato del listado de Motorola.**
8. **El compilador será capaz de detectar errores de diferente índole y lo indicará en el listado agregando mensajes dependiendo del tipo de error.**

|  |  |  |
| --- | --- | --- |
| 001 | CONSTANTE INEXISTENTE |  |
| 002 | VARIABLE INEXISTENTE |  |
| 003 | ETIQUETA INEXISTENTE |  |
| 004 | MNEMÓNICO INEXISTENTE |  |
| 005 | INSTRUCCIÓN CARECE DE OPERANDO(S) |  |
| 006 | INSTRUCCIÓN NO LLEVA OPERANDO(S) |  |
| 007 | MAGNITUD DE OPERANDO ERRONEA |  |
| 008 | SALTO RELATIVO MUY LEJANO |  |
| 009 | INSTRUCCIÓN CARECE DE ALMENOS UN ESPACIO RELATIVO AL MARGEN |  |
| 010 | NO SE ENCUENTRA END |  |
| 011 | NO SE ENCUENTRA ORG |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. **El compilador será capaz de detectar advertencias de diferente índole y lo indicará en el listado agregando mensajes dependiendo del tipo de advertencia.**

|  |  |  |
| --- | --- | --- |
| 001 | **VARIABLE NOMBRADA IGUAL QUE UNA ETIQUETA** |  |
| 002 | **CONSTANTE NOMBRADA IGUAL QUE UNA ETIQUETA** |  |
|  |  |  |

1. **El compilador deberá generar un archivo de texto con extensión “\*.hex” que contenga el código objeto utilizando el siguiente formato:**

**<dirección> XX0 XX1 XX2 XX3 XX4 XX5 XX6 XX7 XX8 XX9 XXA XXB XXC XXD XXE XXF**

**…**

**Ejemplo:**

**<8000> 02 56 33 76 4B CC 26 38 59 AA 26 09 12 49 5D FF**

**<8010> 3E A4 29 56 67 18 9C 55 8F 22 06 00 0F 3C A0 71**

**...**

**Donde cada “xx” es un byte.**

1. **csa**
2. **veG**