

GCC MC68HC11 Compiler

HOW TO

ver 0.5

Create a C program for the 68HC11 using a text editor (use notepad or notepad++, available at <http://notepad-plus-plus.org/>) to create the code file. Name it as a ".c" file.

Goto the file submission page (<http://tech-uofm.info/upload/upload1.php>) and submit the file.

When you submit the file one of two things will happen:

- If there is an error, the browser will show you the errors returned by the compiler similar to the capture below

```
Success!
The file (test5 - Copy.c) has been resubmitted on September 10, 2012, 9:07 pm as lab_1a_c_fred.c.

-----

Compiling /home/uofm/public_html/upload/files/TECH4234/lab_1a_c_fred.c

-----

ERROR: /home/uofm/public_html/upload/files/TECH4234/lab_1a_c_fred.c: In function 'main':
ERROR: /home/uofm/public_html/upload/files/TECH4234/lab_1a_c_fred.c:25: error: syntax error before "disp8"
ERROR:
```

Here the compiler is showing that there is an error in function main. The error is on line 25 and it is a *syntax error before "disp8"*. You will have to debug your code, edit your .c file and resend it to correct the errors.

- If the compile is successful you show see something similar to the following

```
Success!
The file (test5.c) has been resubmitted on September 10, 2012, 9:09 pm as lab_1a_c_fred.c.

-----

Compiling /home/uofm/public_html/upload/files/TECH4234/lab_1a_c_fred.c

-----

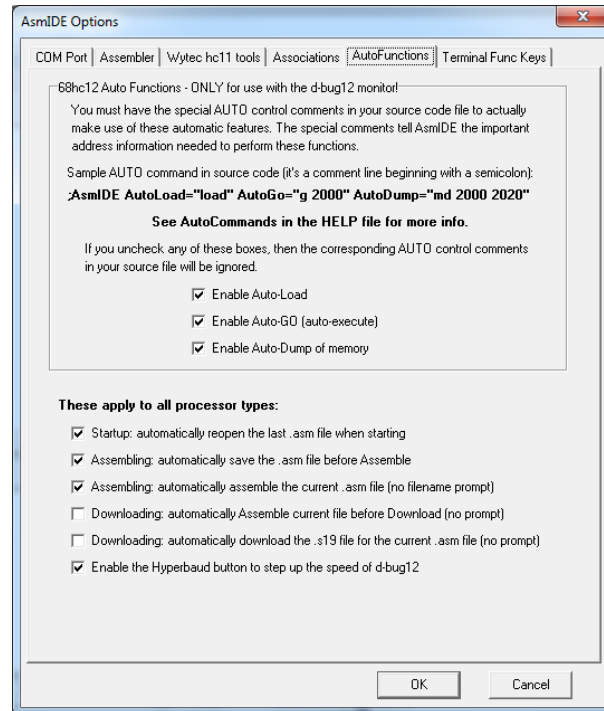
Compile Successful!
Download .S19 File
Download .LST File
```

From here you can do a right click on both the .s19 file and the .lst file to get them to your computer (you only need the .s19 file, the .lst (or List File) is for reference when debugging code.

To send the .s19 file to the Fox11 board:

- Open AsmIDE as you would start an ASM program for the FOX11 board
- On the menu bar goto VIEW | OPTIONS and click on the AutoFunctions Tab.
- Uncheck:
 - Downloading: automatically assemble the current.asm file (no filename prompt)
 - Downloading: automatically download the .s19 file for the current .asm file (no prompt)

So the screen looks like:



These changes make it so it will ask the user to select a .s19 file instead of assuming the .s19 file for the .asm file opened in the editor window.

- Click on the terminal window and type 'load t' like you would to get the Fox11 board ready to receive code.
- On the menu, do a BUILD | Download. At this point it should ask you to open a .s19 file (select the one you downloaded from the file submission system)
- In the terminal window you should see the '*****' indicating a download in progress.
- Once the '>' prompt is available, enter the command 'g 8000' and the program should execute.

Available C Libraries:

buffalo.h – allows for the use of the BUFFALO routines similar to the buffalo function calls in asm.

```
extern void outstr(const char *);           // Call OUTA to display a string
extern int input(void);                    // Call INPUT
extern int inchar(void);                   // Call INCHAR
extern void abort(void);                   // Generate an SWI
extern void displ6(unsigned word);         // Display 16-bit hex number
extern void disp8(unsigned char byte);     // Display 8-bit hex number
extern unsigned char atox8(const char[2]); // Convert two characters to 8-bit
hex
extern unsigned atox16(const char[4]);     // Convert four characters to
// 16-bit hex

extern void interruptsON(void);            // Enable interrupts using CLI
extern void interruptsOFF(void);          // Disable interrupts using SEI
```

fox11_lcd.h – allows for the use of the LCD routines available on FOX11 boards

```
extern void lcd_init(void);
extern void lcd_line1(const char *);
extern void lcd_line2(const char *);
```

hc11a8.h – defines all ports (and the bits there-in) for the 68HC11 hardware control registers. For EXAMPLE:

```
#define ADCTL (*(volatile REGISTER*)(REG_BASE + 0x30))
```

The ADCTL port is defined as REG_BASE (\$1000) + \$30 or \$1030. So we can set this register up by using a line like:

```
ADCTL = 0x27;
```

Will set ADCTL to \$27.

interrupts.h – defines the interrupt JMP table used by BUFFALO. [will be added later]

Other Notes:

To use the FOX11 LED lights you will need to set up a pointer to the LED's address of \$1404. Here is the definition and a line setting the leds:

```
unsigned char volatile * leds_ptr = (unsigned char *) 0x1404;
*leds_ptr = 0xAA;
```

Will turn on the LED's in the pattern of ON OFF ON OFF ON OFF ON OFF

To read the FOX11 dip switch you must set up the pointer to the dip switch address of \$1403 and then you can read the switch status into any variable:

```
unsigned char volatile * switches_ptr = (unsigned char *) 0x1403;  
int x;  
  
x= *switches_ptr;
```