

How to Compile with a Mixture of Assembly and C Files Using ImageCraft's ICC11

Application Note AN0001

Author : Lin Zhao

Date : August 4, 2003

Abstract: This application note shows how to compile with a mixture of assembly and C files using ImageCraft's ICC11. Although the programs in this application note were tested on the EVBPlus2 and FOX11 evaluation boards from Wyntec Company, they will work on most 68HC11 boards that boot from Buffalo monitor.

There are mainly two approaches to writing assembly language with ICC11. One of the method is to insert a single assembly instruction directly into a C function using *asm("string");*. Another method is to insert an entire assembly file into a C program using *asm(".include 'filename'");*. In this application note, we will concentrate on how to insert an entire assembly file into a C file.

In the example C file as shown in Listing 1, an assembly file, "ex.s", was embedded using *asm(".include 'ex2.s'");*. In the main program, a message "Calling Assembly Subroutine..." was printed in the terminal window before calling a function defined in the assembly file. Then, the "start()" function that was defined in the assembly file was called to start a counter, which will be explained in "ex.s" assembly file. Notice that the assembly function "start()" does not need a prototype in the C program.

Listing 1: The example C program with an assembly file embedded.

```
/*
 * C File Name      : assemcall.c
 * Assembly File Name: ex.s
 * Author          : Lin zhao
 * Date            : 8/4/2003
 * Description      : A sample file for C file to embed an assembly language file.
 */

#include <hc11.h>

asm(".include 'ex2.s'"); // insert assembly here

main()
{
    setbaud(BAUD9600);
    while(1) {
        puts("Calling Assembly subroutine...\n");
        start(); // call to assembly subroutine
    }
}
```

```

    }
}
/* Activate this ONLY if you are using BUFFALO. Using the
 * ROM routine will cause the character to be output
 * regardless of the port used.
 */
#if 1 // change to 1 to activate
int putchar(char c)
{
    asm("tba\n"); // BUFFALO expects the char in A
    asm("jsr 0xFFAF");
}
#endif

```

The assembly program, “ex.s”, that was embedded in the above C program is shown in Listing 2. This assembly program decreases a counter loaded with 0x0F and displays the content of the counter on the LEDs on Port B. Once the content of the counter reaches zero, the assembly program returns to the C program.

The *.area text* is the standard place for programs (in ROM). Assembly level function “**start()**” was defined beginning with an underscore, “_”. Notice that in the assembly file the name has an underscore, but the same name in the C file does not.

Listing 2: Assembly file “ex.s” embedded in the example C file

```

.area text

portb = 0x04
REGBLK = 0x1000
SPEED = 0x8000

_start:
    ldx    #REGBLK
    ldaa  #0x0f
back:
    staa  portb,x
    jsr   delay
    deca
    bne  back

delay:
    ldy   #SPEED
dly:
    dey
    bne  dly
    rts

```

Note:

The assembly program “ex.s” and the C program “assemcall.c” must be put in the same folder before you create your project and have these files compiled.

In order to run the S19 file on the EVBPlus2 evaluation board in Buffalo mode, you only need to add the C file to your project. DO NOT add assembly file and “vectors.c” to your project.

Also, if you use a FOX11 board, you have to redefine PORTB in the header file “hc11.h”. Replace “#define PORTB *(unsigned char volatile *) (_IO_BASE + 0x04)” with “#define PORTB *(unsigned char volatile *) (_IO_BASE + 0x404)”.

Please refer to the Application Note, *Run “Hello World” Example on EVBPlus2 Using ImageCraft’s ICC11 and Buffalo*, for how to use ICC11 C compiler.

Refer to Jonathan W. Valvano’s “*Developing Embedded Software in C Using ICC11/ICC12/Hiware*”[1] for more information about assembly programming in ICC11 and ICC12.

References:

1. Jonathan W. Valvano. “*Developing Embedded Software in C Using ICC11/ICC12/Hiware*”. <<http://www.ece.utexas.edu/~valvano/embed/toc1.htm>>