It's actually quite easy! As you can see in the example below, just enclose the ASM part with *__asm* and *__endasm;*

You might have to declare some additional variables, but the concept is quite straightforward.

Do not forget to include the header file of your processor from the SDCC header files, or variables like PRODL will not be defined: e.g. #include <pic18f4685.h>

```
unsigned char Scale_7bit(unsigned char evnt2, unsigned char min, unsigned
char max) {
  // scaled value is (<8-bit random> * ) >> 8
  PRODL = evnt2 << 1; // 8bit value
  PRODH = max-min+1;  // range
__asm
    movf _PRODL, W
    mulwf _PRODH, 0
__endasm;

  return min + PRODH;
}
```

Also, as well as the above example of 'inline ASM', it is possible to include ASM code in .inc include files, and call that ASM code from within your C application. Please see the forum thread using assembler in C - question about _ in front of macro names for more info.

From:
https://www.midibox.org/dokuwiki/ - **MIDIbox**

Permanent link:
**https://www.midibox.org/dokuwiki/doku.php?id=how_to_mix_c_and_asm**

Last update: **2008/09/29 12:27**