

TN248

Mapping Battery-Backed SRAM When Running in Flash

This document explains how to map battery-backed SRAM when using the BIOS memory setting “Code and BIOS in Flash” on boards that have both fast non battery-backed SRAM as well as slower battery-backed SRAM. The battery-backed SRAM is not mapped into the physical address space by default when compiling with this BIOS memory setting.

IMPORTANT NOTE: Rabbit Semiconductor strongly discourages the use of “Code and BIOS in Flash” on boards that have both fast non battery-backed SRAM as well as slower battery-backed SRAM. Such boards are expected to use “Code and BIOS in Flash, Run in RAM.” The battery-backed SRAM is mapped by default when compiling with this optimal setting.

Before compiling the code below you must first map the battery-backed SRAM. Go to the Defines tab on the Options | Project Options dialog and enter the following in the Global Macro Definitions window:

```
MB2CR_SETTING = 851  
MB3CR_SETTING = 85
```

Run the following code.

```
int main(void) {  
    long foo;  
    long bytes;  
    int buf;  
  
    bytes = sizeof(buf);  
    printf("MB2CR_SETTING is %d (0x%X), MB3CR_SETTING is %d (0x%X)\n",  
        MB2CR_SETTING, MB2CR_SETTING, MB3CR_SETTING, MB3CR_SETTING);  
    foo = _xalloc(&bytes, 1, XALLOC_BB);  
    printf("allocated %d bytes at %lp\n", bytes, foo);  
    xmem2root(&buf, foo, (int) bytes);  
    printf("Xmem location contains 0x%04X\n", buf);  
    buf = 0xABDC;  
    root2xmem(foo, &buf, (int) bytes);  
    printf("Wrote 0x%04X to xmem, reading back...\n", buf);  
    xmem2root(&buf, foo, (int) bytes);  
    printf("Xmem location contains 0x%04X\n", buf);  
    return 0;  
}
```

1. The compiler will generate warnings regarding the redefinition of MB2CR_SETTING and MB3CR_SETTING that can be safely ignored.

You will see some random value at the allocated xmem location before “0xABDC” is written there. Now cycle power on the board and run the code a second time. If battery-backed RAM is mapped, the value “0xABDC” is retained during the power cycle and displayed by the printf statement instead of the random value that was displayed the first time the code was run.