32-bit MCU Knowledge Base

Search 32-bit MCU Knowledge Base

How to Enable Hardware Floating Point Math for Cortex M4 with FPU in GCC

04/17/2014 | 01:50 am

h for Cortex M4 with FPU in GCC;hashtags=Knowledge Base Articles,32-bit MCUs,)

014/04/16/how_to_enable_hardwa-vM9u.html%0D%0A
Floating%20Point%0A%C2%A0%0A
ortex-m4%20-mfloat-abi%3Dhard%20-mfpu%3Dfpv4-sp-d16%C2
J%20hardware%3Avoid%20SystemInit(void)%7B%C2%A0%C2%A0%20%C2%A0
20Full%20Access%20*%2F%C2%A0%C2%A0%20%C2%A0%20%C2%A0%20%C2%A0
)%3B%20%2F*%20set%20CP11%20Full%20Access%20*%2F%C2%A0%7D%C2%A0
erify%20that%20Hardware%20Floating%20Point%20is%20Enabled%0A%C2%A0%0A
nd%20open%20the%20Disassembly%20view.%0AVerify%20that%20the%20FPU%20instructions
20to%20library%20functions%20and%20execute%20in%20software.%0A%0A%C2%A0%0A%0A

ChrisM (/community/profile.html/home/users/E/Ejm1zITxW2eiGtwmjzOe/profile)
Employee

Question

How do I enable hardware support for floating point math in GCC for EFM32WG (Wonder Gecko)?

Answer

How to Enable Hardware Floating Point

1. Add the following compiler symbol:

   ARM_MATH_CM4=1

2. Add the following flags to the GCC assembler, compiler, and linker:

   -mthumb -mcpu=cortex-m4 -mfloat-abi=hard -mfpu=fpv4-sp-d16

3. Add the following include to source files using floating point math:

   #include "arm_math.h"

4. Make sure **SystemInit()** turns on the FPU hardware:

```
void SystemInit(void)
{
  /* Set floating point coprosessor access mode. */
  SCB->CPACR |= ((3UL << 10*2) | /* set CP10 Full Access */
                (3UL << 11*2) ); /* set CP11 Full Access */
}
```

5. (Optional) Add CMSIS\Lib\GCC\libarm_cortexM4lf_math.a to your project for

Topics: *Knowledge Base Articles, 32-bit MCUs*