

## HW2

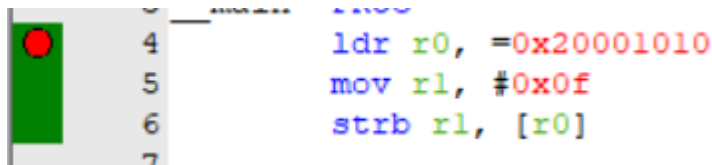
다음 질문에 대해서 답하기 위하여 MDK-ARM을 이용하여 수행하십시오.

(MDK-ARM에서 project를 만드는 방법은 HW1을 참고하십시오.)

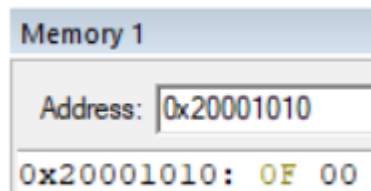
가능한 모든 답에서 디버거에 나타나는 적절한 부분을 캡처하여 첨부하여야 합니다.

1. 다음 (a) ~ (c) 문제는 별개의 program이 아니고, 단계적으로 실행되는 문제이다.

(a) 0x20001010 번지에 1 byte 0x0F 값을 저장시키는 assembly program을 작성하고, 디버거를 이용하여 확인하십시오. (5점)



```
4      ldr r0, =0x20001010
5      mov r1, #0x0f
6      strb r1, [r0]
```

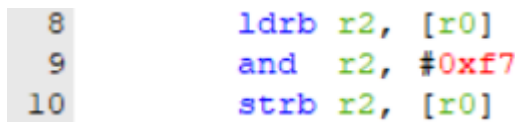


Memory 1

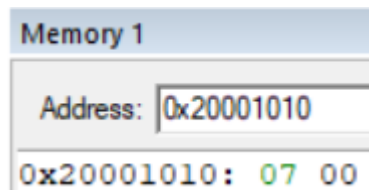
Address: 0x20001010

0x20001010: 0F 00

(b) 0x20001010 번지의 bit 3를 clear시키는 assembly program을 작성하고, 디버거를 이용하여 확인하십시오. (bit-band alias 주소를 이용하지 말 것!) (10 점)



```
8      ldrb r2, [r0]
9      and r2, #0xf7
10     strb r2, [r0]
```



Memory 1

Address: 0x20001010

0x20001010: 07 00

(c) 0x20001010 번지의 bit 3의 bit-band alias address는 무엇인가? 5 (점)

$$0x22000000 + 0x1010 \times 32 + 3 \times 4 = 0x2202020c$$

(d) 0x20001010 번지의 bit 3를 set시키는 assembly program을 작성하고, 디버거를 이용하여 확인하십시오. (bit-band alias 주소를 이용하십시오.) (5점)

```

12      ldr    r0, =0x2202020c
13      mov    r1, #1
14      str    r1, [r0]

```

Memory 1

Address: 0x20001010

0x20001010: 0F 00

2. 1번의 프로그램을 디버깅하기 위해서 디버거를 동작시켰을 때 (혹은 디버거 동작 중에 reset 버튼을 눌렀을 때)

(a) R13 (SP)의 값은 무엇인가? 그 값은 어떻게 정해진 것인가? (5점)

아래에서 보는 바와 같이 SP = 0x20000600 이다. 이 값은 reset 후에 0x00 번지에서 읽어 온 값을 SP (MSP)에 assign 시키기 때문이다.

(b) R15 (PC)의 값은 무엇인가? 그 값은 어떻게 정해진 것인가? (10점)

아래에서 보는 바와 같이 PC = 0x080001BC 이다. 이 값은 reset 후에 0x04 번지에서 읽어 온 값을 PC에 assign하기 때문이다. 아래 캡처한 부분에서 보면 0x04번지에는 0x080001BD 값이 저장되어 있으나 bit 0는 동작 모드를 표현하는 부분이기 때문에 reset 후 프로그램이 시작되는 부분은 bit 0를 '0'으로 한 0x080001BC 가 되고 동작 모드는 Thumb 모드가 된다는 의미이다.

R13 (SP)	0x20000600
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x080001BC

Memory 1

Address: 0x00

0x00000000: 00 06 00 20 BD 01 00 08

3. 2번과 보는 바와 같은 결과가 어떻게 초래되는 것인지 (즉 0x00번지와 0x04번지에 특정 값이 주어져도록 code가 작성된 부분을 찾고, 그 부분을 설명하시오. (startup\_stm32f411xe.s 를 참조하여야 한다) (10 점)

```

Heap_Size      EQU      0x00000200

__heap_base    AREA     HEAP, NOINIT, READWRITE, ALIGN=3
Heap_Mem       SPACE    Heap_Size
__heap_limit

Stack_Size     EQU      0x00000400

__stack_base   AREA     STACK, NOINIT, READWRITE, ALIGN=3
Stack_Mem      SPACE    Stack_Size
__initial_sp

```

여기서는 SRAM이 0x20000000 ~ 번지에 위치해 있고 heap memory 크기를 0x200으로 잡고 있기 때문에 heap memory 영역이 0x20000000 ~ 0x200001FF 까지가 되고, 그 다음 stack 크기를 0x400 으로 잡고 있기 때문에 stack 영역은 0x20000200 ~ 0x200005FF 까지 된다. 그리고 그 다음에 \_\_initial\_sp label이 위치해 있기 때문에 \_\_initial\_sp 값은 0x20000600이 된다.

```

56      AREA     RESET, DATA, READONLY
57      EXPORT   __Vectors
58      EXPORT   __Vectors_End
59      EXPORT   __Vectors_Size
60
61  __Vectors      DCD      __initial_sp          ; Top of Stack
62                DCD      Reset_Handler        ; Reset Handler

```

RESET area (section)이 링커에서 0x00번지부터 위치하도록 되어 있기 때문에 0x00 ~ 0x03 번지에 (DCD \_\_initial\_sp) 값이 들어가게 되고, 0x04 ~ 0x07 번지에는 Reset\_Handler label 값이 들어가게 된다. (즉 reset 후에 처음 실행되는 코드가 위치한 주소) 하지 assembler에서 알아서 Reset\_Handler가 위치한 값보다 +1을 증가시킨 값을 실질적으로 0x04 ~ 0x07 번지에 저장하게 된다. (이 부분이 reset handler 부분이라는 것을 이미 알고 있고 따라서 ARM Cortex-M 에서는 항상 thumb 모드에서 동작하므로 알아서 +1을 한 값이 저장된다.

참고로 프로젝트 directory에서 Listings/hw2.map 파일이 생성되게 되고 그 안의 내용을 보면 해당 프로그램이 어떤 주소에 할당되어 있는 가를 알 수 있다 관련 부분을 살펴 보면 다음과 같다.

Heap_Mem	0x20000000	Data	512	startup_stm32f411xe.o(HEAP)
HEAP	0x20000000	Section	512	startup_stm32f411xe.o(HEAP)
Stack_Mem	0x20000200	Data	1024	startup_stm32f411xe.o(STACK)
STACK	0x20000200	Section	1024	startup_stm32f411xe.o(STACK)
__initial_sp	0x20000600	Data	0	startup_stm32f411xe.o(STACK)
__Vectors_Size	0x00000198	Number	0	startup_stm32f411xe.o ABSOLUTE
__Vectors	0x08000000	Data	4	startup_stm32f411xe.o(RESET)
__Vectors_End	0x08000198	Data	0	startup_stm32f411xe.o(RESET)
__main	0x08000199	Thumb Code	26	hw2.o(.text)
Reset_Handler	0x080001bd	Thumb Code	8	startup_stm32f411xe.o(.text)

Exec Addr	Load Addr	Size	Type	Attr	Idx	E Section Name	object
0x20000000	-	0x00000200	Zero	RW	8	HEAP	startup_stm32f411xe.o
0x20000200	-	0x00000400	Zero	RW	7	STACK	startup_stm32f411xe.o

(a) SYSCFG memory remap register의 현재 값은 무엇인가? (디버거에서 Peripherals -> System viewer -> SYSCFG -> MEMRM 값을 보면 된다.) "stm32f411 reference manual"을 참고하여 SYSCFG memory remap register의 현재 값이 의미하는 것이 무엇인지 설명하시오. (10 점)

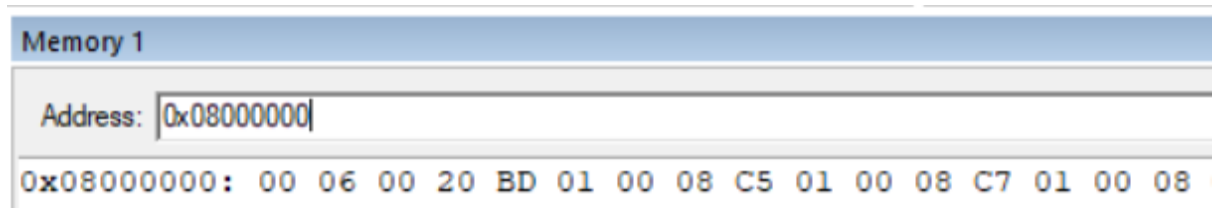
```
00: Main Flash memory mapped at 0x0000 0000
01: System Flash memory mapped at 0x0000 0000
11: Embedded SRAM mapped at 0x0000 0000
```

(b) 메모리 0x00000000 ~ 0x0000000F 영역의 값과 0x08000000 ~ 0x0800000F 영역의 값을 비교하시오. (5 점)

Memory 1

Address: 0x00

0x00000000: 00 06 00 20 BD 01 00 08 C5 01 00 08 C7 01 00 08



위의 캡처 내용에서 보는 바와 같이 0x00000000 ~ 메모리 영역의 값이 0x08000000 ~ 영역의 값과 동일 함을 알 수 있다. 즉 0x08000000 ~ flash memory 영역이 0x00000000 ~ 으로 remap 되어 있다.