

Mastering Embedded System

Unit3 Lesson3

Lab_2

CORTEX-M3 STM32F103C6



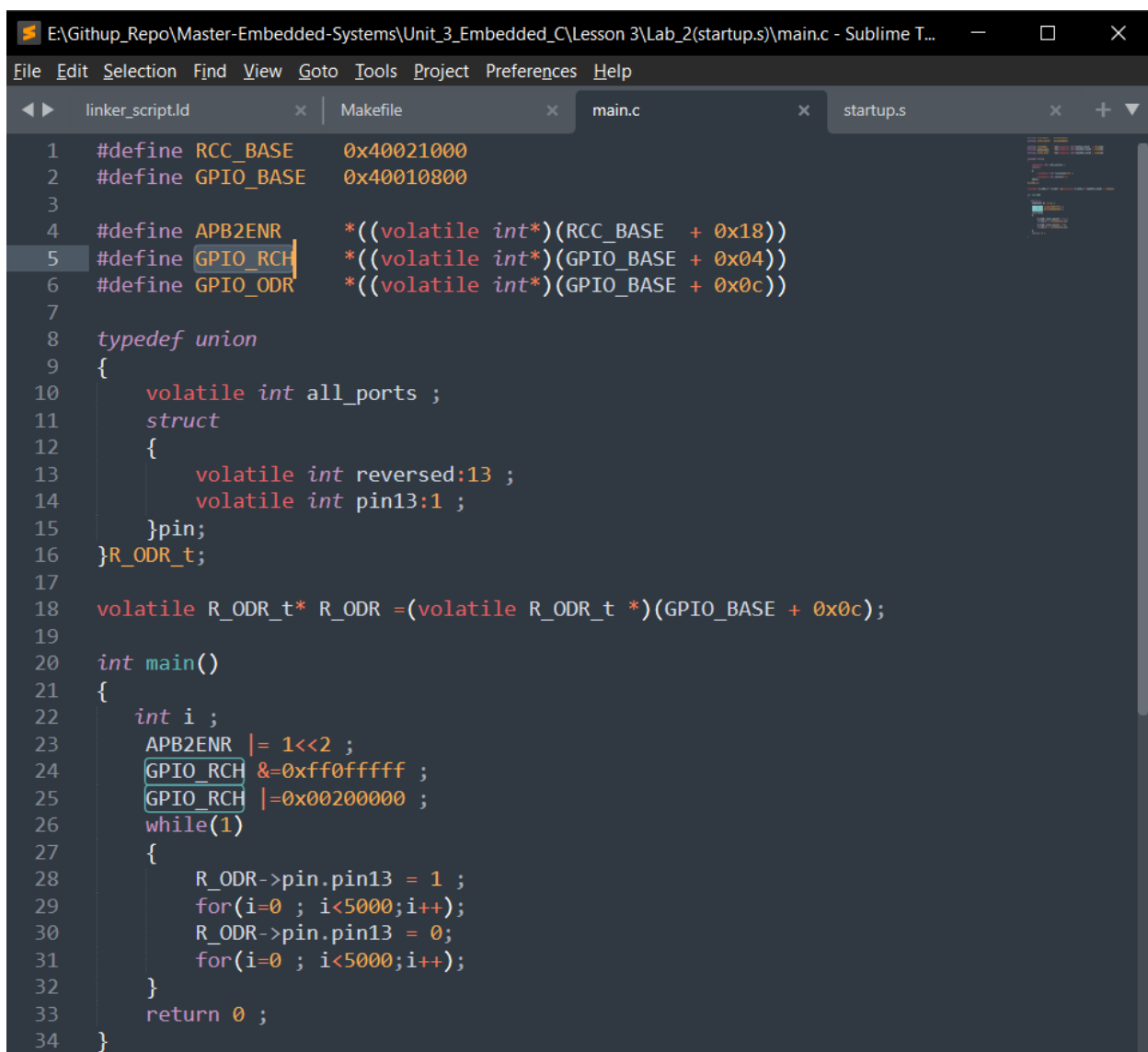
I will write bare-metal software for a SM32F103C6 to toggle the led on part A pin 13.

Peripherals:

RCC stands for reset and clock control, and its base address is 0x40021000. While APB2ENR has an offset from the base of 0x18 ,set its value with 1.

GPIO stands for general purpose input output, and its base address is 0x40010800. It has 2 registers:

- 1) GPIO_RCH has an offset from the base 0x04, set its bits (20:24) by 0x2.
- 2) GPIO_ODR has an offset from the base 0x0c, toggle value on pin13 to toggle the led.



```
E:\Github_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\main.c - Sublime T...
File Edit Selection Find View Goto Tools Project Preferences Help
linker_script.ld Makefile main.c startup.s
1 #define RCC_BASE 0x40021000
2 #define GPIO_BASE 0x40010800
3
4 #define APB2ENR *((volatile int*)(RCC_BASE + 0x18))
5 #define GPIO_RCH *((volatile int*)(GPIO_BASE + 0x04))
6 #define GPIO_ODR *((volatile int*)(GPIO_BASE + 0x0c))
7
8 typedef union
9 {
10     volatile int all_ports ;
11     struct
12     {
13         volatile int reversed:13 ;
14         volatile int pin13:1 ;
15     }pin;
16 }R_ODR_t;
17
18 volatile R_ODR_t* R_ODR =(volatile R_ODR_t *) (GPIO_BASE + 0x0c);
19
20 int main()
21 {
22     int i ;
23     APB2ENR |= 1<<2 ;
24     GPIO_RCH &=0xff0fffff ;
25     GPIO_RCH |=0x00200000 ;
26     while(1)
27     {
28         R_ODR->pin.pin13 = 1 ;
29         for(i=0 ; i<5000;i++);
30         R_ODR->pin.pin13 = 0;
31         for(i=0 ; i<5000;i++);
32     }
33     return 0 ;
34 }
```

```
E:\Github Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\startup.s - Sublime ...
File Edit Selection Find View Goto Tools Project Preferences Help
linker_script.ld x Makefile x main.c x startup.s x + ▼
1  /*startup.s
2  @Abdullah Kortam
3  */
4
5  /*Sram 0x20000000 */
6
7  .section .vectors
8  .word 0x20001000 /* stact top address */
9  .word _reset /* 1 Reset */
10 .word vector_handler /* 2 NMI */
11 .word vector_handler /* 3 Hard Fault */
12 .word vector_handler /* 4 MM Fault */
13 .word vector_handler /* 5 Bus Fault */
14 .word vector_handler /* 6 */
15 .word vector_handler /* 7 RESERVED */
16 .word vector_handler /* 8 RESERVED */
17 .word vector_handler /* 9 RESERVED */
18 .word vector_handler /* 10 RESERVED */
19 .word vector_handler /* 11 SV call */
20 .word vector_handler /* 12 Debug reserved */
21 .word vector_handler /* 13 RESERVED */
22 .word vector_handler /* 14 pendSV */
23 .word vector_handler /* 15 SysTick */
24 .word vector_handler /* 16 IRQ0 */
25 .word vector_handler /* 17 IRQ1 */
26 .word vector_handler /* 18 IRQ2 */
27 .word vector_handler /* 19 ... */
28     /* On to IRQ067 */
29
30 .section .text
31 _reset :
32     bl main
33     b .
34
35 .thumb_func
36
37 vector_handler:
38     b _reset
```

```
E:\Github_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\Makefile - Sublime...
File Edit Selection Find View Goto Tools Project Preferences Help
linker_script.ld x Makefile x main.c x startup.s x + ▼
1
2
3 CC =arm-none-eabi
4 CFLAGS = -mcpu=cortex-m3 -mthumb -gdwarf-2
5 INCS=-I .
6
7 SRC=$(wildcard *.c)
8 OBJ=$(SRC:.c=.o)
9
10 AS=$(wildcard *.s)
11 ASOBG=(AS:.s=.o)
12
13
14 project_name=abdullah_cortex_m3
15
16 all : $(project_name).bin
17     @echo "=====BUILD DONE=====
18 startup.o : startup.s
19     $(CC)-as.exe $(CFLAGS) $< -o $@
20
21 %.o : %.c
22     $(CC)-gcc.exe -c $(INCS) $(CFLAGS) $< -o $@
23
24 uart.o : uart.c
25     $(CC)-gcc.exe -c $(INCS) $(CFLAGS) $< -o $@
26
27 $(project_name).elf: $(OBJ) startup.o
28     $(CC)-ld.exe -T linker_script.ld $(OBJ) $(ASOBG) -o $@ -Map=Map_file.map
29
30 $(project_name).bin : $(project_name).elf
31     $(CC)-objcopy.exe -O binary $< $@
32
33 remove_all :
34     rm *.o *.elf *.bin
35 remove :
36     rm *.elf *.bin
```

```
E:\Githup_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\linker_script.ld - Su...
File Edit Selection Find View Goto Tools Project Preferences Help
linker_script.ld x Makefile x main.c x startup.s x + ▼
1  /*startup.s
2  @Abdullah Kortam
3  */
4
5  MEMORY
6  {
7      flash(RX) : ORIGIN = 0x08000000 , LENGTH = 128k
8      Sram (RWX): ORIGIN = 0x20000000 , LENGTH = 20k
9  }
10
11  SECTIONS
12  {
13      .text :
14      {
15          _S_text = . ;
16          *(.vectors)
17          *(.text)
18          *(.rodata)
19          _E_text = . ;
20      }>flash
21
22      .data :
23      {
24          _S_data = . ;
25          *(.data)
26          _E_data = . ;
27      }>Sram AT>flash
28
29      .bss :
30      {
31          _S_bss = . ;
32          *(.bss)
33          _E_bss = . ;
34          . = ALIGN(4);
35          . = . + 0x1000 ;
36          _STACK_TOP = . ;
37      }>Sram
38  }
```

E:\Github_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\Map_file.map - Su

File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld x Map_file.map x Makefile x main.c

```
1
2 Memory Configuration
3
4 Name          Origin          Length          Attributes
5 flash         0x08000000      0x00020000      xr
6 Sram          0x20000000      0x00005000      xrw
7 *default*     0x00000000      0xffffffff
8
9 Linker script and memory map
10
11
12 .text          0x08000000      0xa8             _S_text = .
13               0x08000000
14 *(.vectors)
15 *(.text)
16 .text          0x08000000      0xa8 main.o
17               0x08000000      main
18 *(.rodata)
19               0x080000a8      _E_text = .
20
21 .glue_7        0x080000a8      0x0
22 .glue_7        0x00000000      0x0 linker stubs
23
24 .glue_7t       0x080000a8      0x0
25 .glue_7t       0x00000000      0x0 linker stubs
26
27 .vfp11_veneer  0x080000a8      0x0
28 .vfp11_veneer  0x00000000      0x0 linker stubs
29
30 .v4_bx         0x080000a8      0x0
31 .v4_bx         0x00000000      0x0 linker stubs
32
33 .iplt          0x080000a8      0x0
34 .iplt          0x00000000      0x0 main.o
35
36 .rel.dyn       0x080000a8      0x0
37 .rel.iplt      0x00000000      0x0 main.o
38
39 .data          0x20000000      0x4 load address 0x080000a8
40               0x20000000      _S_data = .
```

E:\Github_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\Map_file.map - Su

File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld Map_file.map Makefile main.c

```
21 .glue_7      0x080000a8      0x0
22 .glue_7      0x00000000      0x0 linker stubs
23
24 .glue_7t     0x080000a8      0x0
25 .glue_7t     0x00000000      0x0 linker stubs
26
27 .vfp11_veneer 0x080000a8      0x0
28 .vfp11_veneer 0x00000000      0x0 linker stubs
29
30 .v4_bx       0x080000a8      0x0
31 .v4_bx       0x00000000      0x0 linker stubs
32
33 .iplt        0x080000a8      0x0
34 .iplt        0x00000000      0x0 main.o
35
36 .rel.dyn     0x080000a8      0x0
37 .rel.iplt    0x00000000      0x0 main.o
38
39 .data        0x20000000      0x4 load address 0x080000a8
40              0x20000000      _S_data = .
41 *(.data)
42 .data        0x20000000      0x4 main.o
43              0x20000000      R_ODR
44              0x20000004      _E_data = .
45
46 .igot.plt    0x20000004      0x0 load address 0x080000ac
47 .igot.plt    0x00000000      0x0 main.o
48
49 .bss         0x20000004      0x1000 load address 0x080000ac
50              0x20000004      _S_bss = .
51 *(.bss)
52 .bss         0x20000004      0x0 main.o
53              0x20000004      _E_bss = .
54              0x20000004      . = ALIGN (0x4)
55              0x20001004      . = (. + 0x1000)
56 *fill*       0x20000004      0x1000
57              0x20001004      _STACK_TOP = .
58 LOAD main.o
59 OUTPUT(abdullah_cortex_m3.elf elf32-littlearm)
60
```

E:\Github_Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.s)\Map_file.m

File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld

Map_file.map

Makefile

```
57      0x20001004      _STACK_TOP = .
58  LOAD main.o
59  OUTPUT(abdullah_cortex_m3.elf elf32-littlearm)
60
61  .debug_info      0x00000000      0xce
62  .debug_info      0x00000000      0xce main.o
63
64  .debug_abbrev     0x00000000      0xbf
65  .debug_abbrev     0x00000000      0xbf main.o
66
67  .debug_loc        0x00000000      0x38
68  .debug_loc        0x00000000      0x38 main.o
69
70  .debug_aranges     0x00000000      0x20
71  .debug_aranges
72      0x00000000      0x20 main.o
73
74  .debug_line        0x00000000      0x56
75  .debug_line        0x00000000      0x56 main.o
76
77  .debug_str         0x00000000      0x53
78  .debug_str         0x00000000      0x53 main.o
79
80  .comment           0x00000000      0x11
81  .comment           0x00000000      0x11 main.o
82      0x12 (size before relaxing)
83
84  .ARM.attributes
85      0x00000000      0x33
86  .ARM.attributes
87      0x00000000      0x33 main.o
88
89  .debug_frame       0x00000000      0x2c
90  .debug_frame       0x00000000      0x2c main.o
91
```


E:\Github Repo\Master-Embedded-Systems\Unit_3_Embedded_C\Lesson 3\Lab_2(startup.c)\startup.c -

File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld x Map_file.map x startup.c x

```
4  * @author      : Abdullah Kortam
5  ****
6  */
7  #include <stdint.h>
8
9  extern uint32_t _STACK_TOP ;
10
11 extern uint32_t main() ;
12 void Reset_Handler();
13
14 void Default_Hundler()
15 {
16     Reset_Handler();
17 }
18
19 void NMI_Handler ()      __attribute__((weak,alias("Default_Hundler")));
20 void H_Fault_Handler()   __attribute__((weak,alias("Default_Hundler")));
21 void Bus_Fault()         __attribute__((weak,alias("Default_Hundler")));
22 void Usage_Fault_Handler() __attribute__((weak,alias("Default_Hundler")));
23
24 uint32_t vector[] __attribute__((section(".vectors")))= {
25     (uint32_t) &_STACK_TOP ,
26     (uint32_t) &Reset_Handler,
27     (uint32_t) &NMI_Handler,
28     (uint32_t) &H_Fault_Handler,
29     (uint32_t) &Bus_Fault,
30     (uint32_t) &Usage_Fault_Handler
31 };
32
33 extern uint32_t _E_TEXT ;
34 extern uint32_t _S_DATA ;
35 extern uint32_t _E_DATA ;
36 extern uint32_t _S_BSS ;
37 extern uint32_t _E_BSS ;
38 uint32_t i ;
39 void Reset_Handler ()
40 {
41     uint32_t DATA_SIZE = (uint8_t*) &_E_DATA - (uint8_t*) &_S_DATA ;
42     uint8_t P_src = (uint8_t*) &_E_TEXT ;
43     uint8_t P_des = (uint8_t*) &_S_DATA ;
44
45     for(i=0; i<DATA_SIZE; i++)
46     {
47         *((uint8_t*)P_des++) = *((uint8_t*)P_src++) ;
48     }
49
50     uint32_t BSS_Size = (uint8_t*) &_E_BSS - (uint8_t*) &_S_BSS ;
51     P_des = (uint8_t*) &_S_BSS ;
52
53     for (i = 0; i < BSS_Size; ++i)
54     {
55         *((uint8_t*)P_des++) = (uint8_t)0 ;
56     }
57     main();
58 }
59
```