

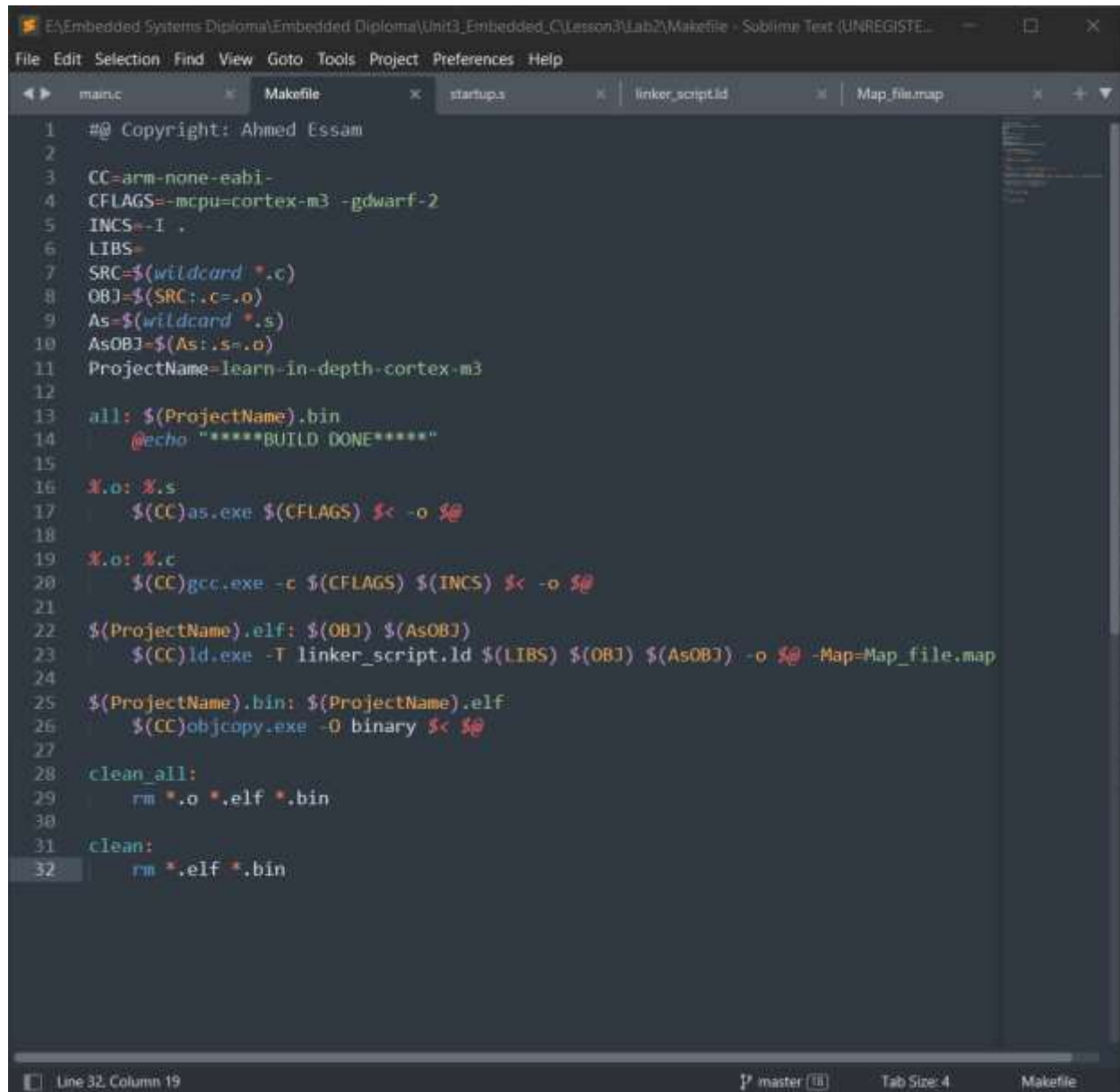
Mastering Embedded Systems

Unit 3

Embedded C

Lab 2

Makefile “As generic as possible”:



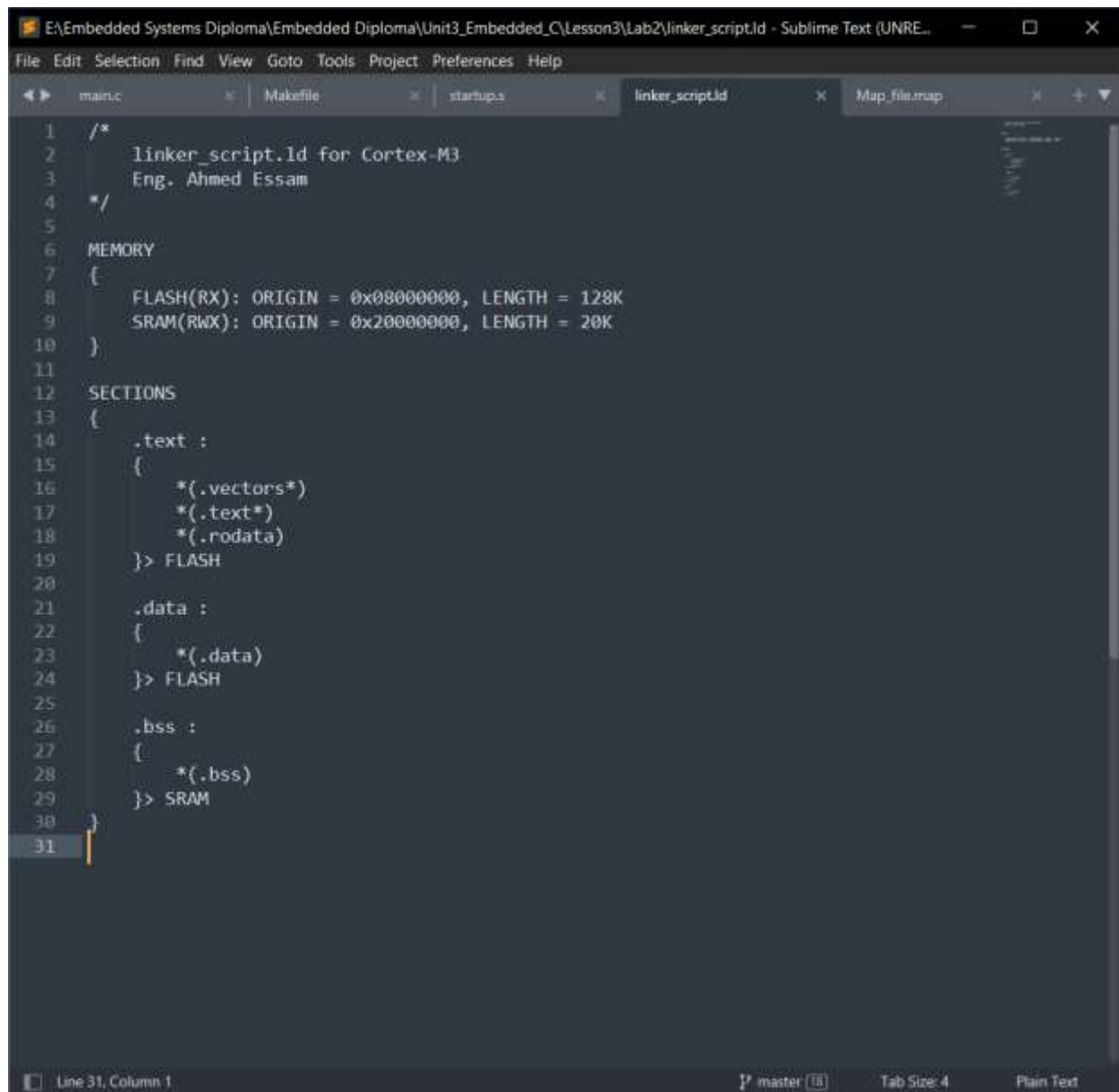
```
1  #@ Copyright: Ahmed Essam
2
3  CC=arm-none-eabi-
4  CFLAGS=-mcpu=cortex-m3 -gdwarf-2
5  INCS=-I .
6  LIBS=
7  SRC=$(wildcard *.c)
8  OBJ=$(SRC:.c=.o)
9  As=$(wildcard *.s)
10 AsOBJ=$(As:.s=.o)
11 ProjectName=learn-in-depth-cortex-m3
12
13 all: $(ProjectName).bin
14     @echo "*****BUILD DONE*****"
15
16 %.o: %.s
17     $(CC)as.exe $(CFLAGS) $< -o $@
18
19 %.o: %.c
20     $(CC)gcc.exe -c $(CFLAGS) $(INCS) $< -o $@
21
22 $(ProjectName).elf: $(OBJ) $(AsOBJ)
23     $(CC)ld.exe -T linker_script.ld $(LIBS) $(OBJ) $(AsOBJ) -o $@ -Map=Map_file.map
24
25 $(ProjectName).bin: $(ProjectName).elf
26     $(CC)objcopy.exe -O binary $< $@
27
28 clean_all:
29     rm *.o *.elf *.bin
30
31 clean:
32     rm *.elf *.bin
```

Part 1:

Startup.s:

```
E:\Embedded Systems Diploma\Embedded Diploma\Unit3_Embedded_C\Lesson3\Lab2\startup.s - Sublime Text (UNREGISTE...
File Edit Selection Find View Goto Tools Project Preferences Help
main.c Makefile startup.s linker_script.ld Map_file.map
1  /*
2      startup.s for Cortex-M3
3      Eng. Ahmed Essam
4  */
5
6  /*SRAM 0x20000000 */
7
8  .section .vectors
9
10 .word 0x20001000 /* STACK_top address */
11 .word _reset /* 1 Reset */
12 .word Vector_handler /* 2 NMI */
13 .word Vector_handler /* 3 Hard Fault */
14 .word Vector_handler /* 4 MM Fault */
15 .word Vector_handler /* 5 Bus Fault */
16 .word Vector_handler /* 6 Usage Fault */
17 .word Vector_handler /* 7 RESERVED */
18 .word Vector_handler /* 8 RESERVED */
19 .word Vector_handler /* 9 RESERVED */
20 .word Vector_handler /* 10 RESERVED */
21 .word Vector_handler /* 11 SV call */
22 .word Vector_handler /* 12 Debug reserved */
23 .word Vector_handler /* 13 RESERVED */
24 .word Vector_handler /* 14 PendSV */
25 .word Vector_handler /* 15 SysTick */
26 .word Vector_handler /* 16 IRQ0 */
27 .word Vector_handler /* 17 IRQ1 */
28 .word Vector_handler /* 18 IRQ2 */
29 .word Vector_handler /* 19 ... */
30 /* On to IRQ67 */
31
32 _reset:
33     bl main
34     b .
35
36 .thumb_func
37
38 Vector_handler:
39     b _reset
40
Line 9, Column 1 master Tab Size: 4 Plain Text
```

Linker_script.ld:



```
1  /*
2     linker_script.ld for Cortex-M3
3     Eng. Ahmed Essam
4  */
5
6  MEMORY
7  {
8     FLASH(RX): ORIGIN = 0x08000000, LENGTH = 128K
9     SRAM(RWX): ORIGIN = 0x20000000, LENGTH = 20K
10 }
11
12 SECTIONS
13 {
14     .text :
15     {
16         *(.vectors*)
17         *(.text*)
18         *(.rodata)
19     }> FLASH
20
21     .data :
22     {
23         *(.data)
24     }> FLASH
25
26     .bss :
27     {
28         *(.bss)
29     }> SRAM
30 }
31
```

Line 31, Column 1

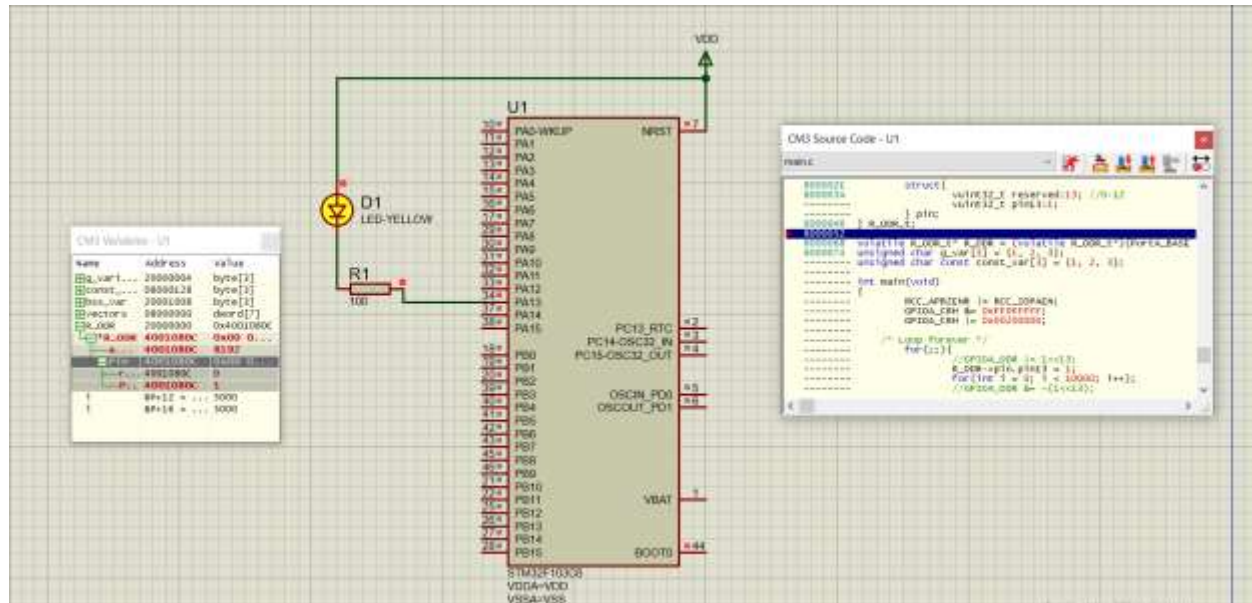
master 18 Tab Size: 4 Plain Text

Main.c:

```
E:\Embedded Systems Diploma\Embedded Diploma\Unit3_Embedded_C\Lesson3\Lab2 - Part1\main.c - Sublime Text (UNREG...
File Edit Selection Find View Goto Tools Project Preferences Help
linker_script.ld x main.c x startup.s x + ▾
17
20 #include <stdint.h>
21
22 #define RCC_BASE 0x40021000
23 #define PORTA_BASE 0x40010800
24 #define RCC_APB2ENR *(volatile uint32_t *) (RCC_BASE + 0x18)
25 #define GPIOA_CRH *(volatile uint32_t *) (PORTA_BASE + 0x04)
26 #define GPIOA_ODR *(volatile uint32_t *) (PORTA_BASE + 0x0C)
27 #define RCC_IOPAEN (1<<2)
28
29 typedef volatile unsigned int vuint32_t;
30
31 typedef union{
32     vuint32_t allFields;
33     struct{
34         vuint32_t reserved:13; //0-12
35         vuint32_t pin13:1;
36     } pin;
37 } R_ODR_t;
38
39 volatile R_ODR_t* R_ODR = (volatile R_ODR_t*)(PORTA_BASE + 0x0C);
40 unsigned char g_var[3] = {1, 2, 3};
41 unsigned char const const_var[3] = {1, 2, 3};
42
43 int main(void)
44 {
45     RCC_APB2ENR |= RCC_IOPAEN;
46     GPIOA_CRH &= 0xFF0FFFFF;
47     GPIOA_CRH |= 0x00200000;
48
49     /* Loop forever */
50     for(;;){
51         //GPIOA_ODR |= 1<<13; //Set
52         R_ODR->pin.pin13 = 1;
53         for(int i = 0; i < 10000; i++);
54         //GPIOA_ODR &= ~(1<<13); //Clear
55         R_ODR->pin.pin13 = 0;
56         for(int i = 0; i < 10000; i++);
57     }
58 }
```

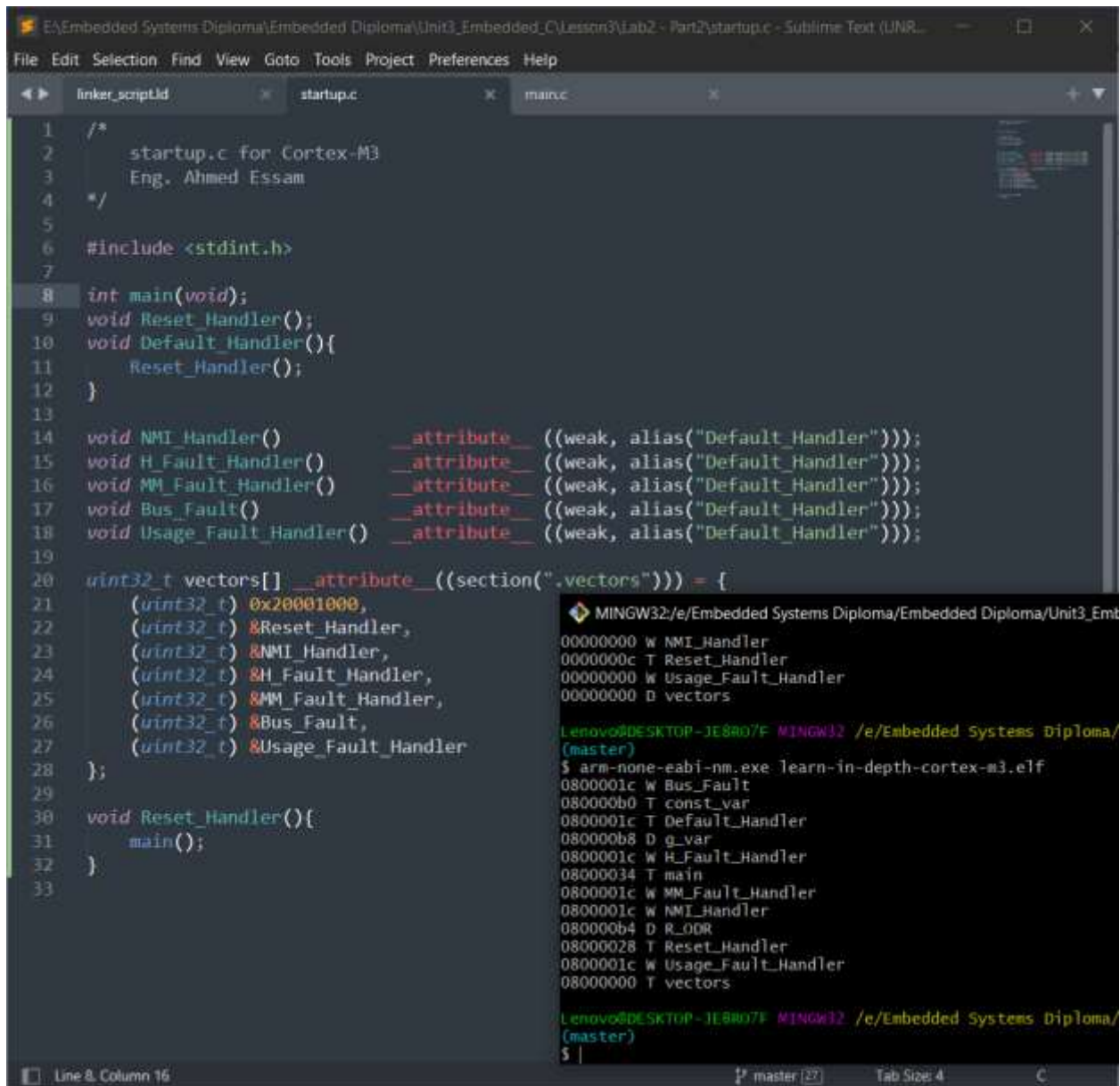
Line 1, Column 1 master (10) Tab Size: 4 C

Proteus screenshot:



Part 2:

Adding `__attribute__((weak, alias("Default_Handler")))` to current handlers and viewing their symbols corresponding to their address:



```
1  /*
2  startup.c for Cortex-M3
3  Eng. Ahmed Essam
4  */
5
6  #include <stdint.h>
7
8  int main(void);
9  void Reset_Handler();
10 void Default_Handler(){
11     Reset_Handler();
12 }
13
14 void NMI_Handler()      __attribute__((weak, alias("Default_Handler")));
15 void H_Fault_Handler()  __attribute__((weak, alias("Default_Handler")));
16 void MM_Fault_Handler() __attribute__((weak, alias("Default_Handler")));
17 void Bus_Fault()        __attribute__((weak, alias("Default_Handler")));
18 void Usage_Fault_Handler() __attribute__((weak, alias("Default_Handler")));
19
20 uint32_t vectors[] __attribute__((section(".vectors"))) = {
21     (uint32_t) 0x20001000,
22     (uint32_t) &Reset_Handler,
23     (uint32_t) &NMI_Handler,
24     (uint32_t) &H_Fault_Handler,
25     (uint32_t) &MM_Fault_Handler,
26     (uint32_t) &Bus_Fault,
27     (uint32_t) &Usage_Fault_Handler
28 };
29
30 void Reset_Handler(){
31     main();
32 }
33
```

```
00000000 W NMI_Handler
0000000c T Reset_Handler
00000000 W Usage_Fault_Handler
00000000 B vectors

Lenovo@DESKTOP-JE8R07F MINGW32 /e/Embedded Systems Diploma/
(master)
$ arm-none-eabi-nm.exe learn-in-depth-cortex-m3.elf
0800001c W Bus_Fault
080000b0 T const_var
0800001c T Default_Handler
080000b8 D g_var
0800001c W H_Fault_Handler
08000034 T main
0800001c W MM_Fault_Handler
0800001c W NMI_Handler
080000b4 D R_ODR
08000028 T Reset_Handler
0800001c W Usage_Fault_Handler
08000000 T vectors

Lenovo@DESKTOP-JE8R07F MINGW32 /e/Embedded Systems Diploma/
(master)
$
```


Startup.c:

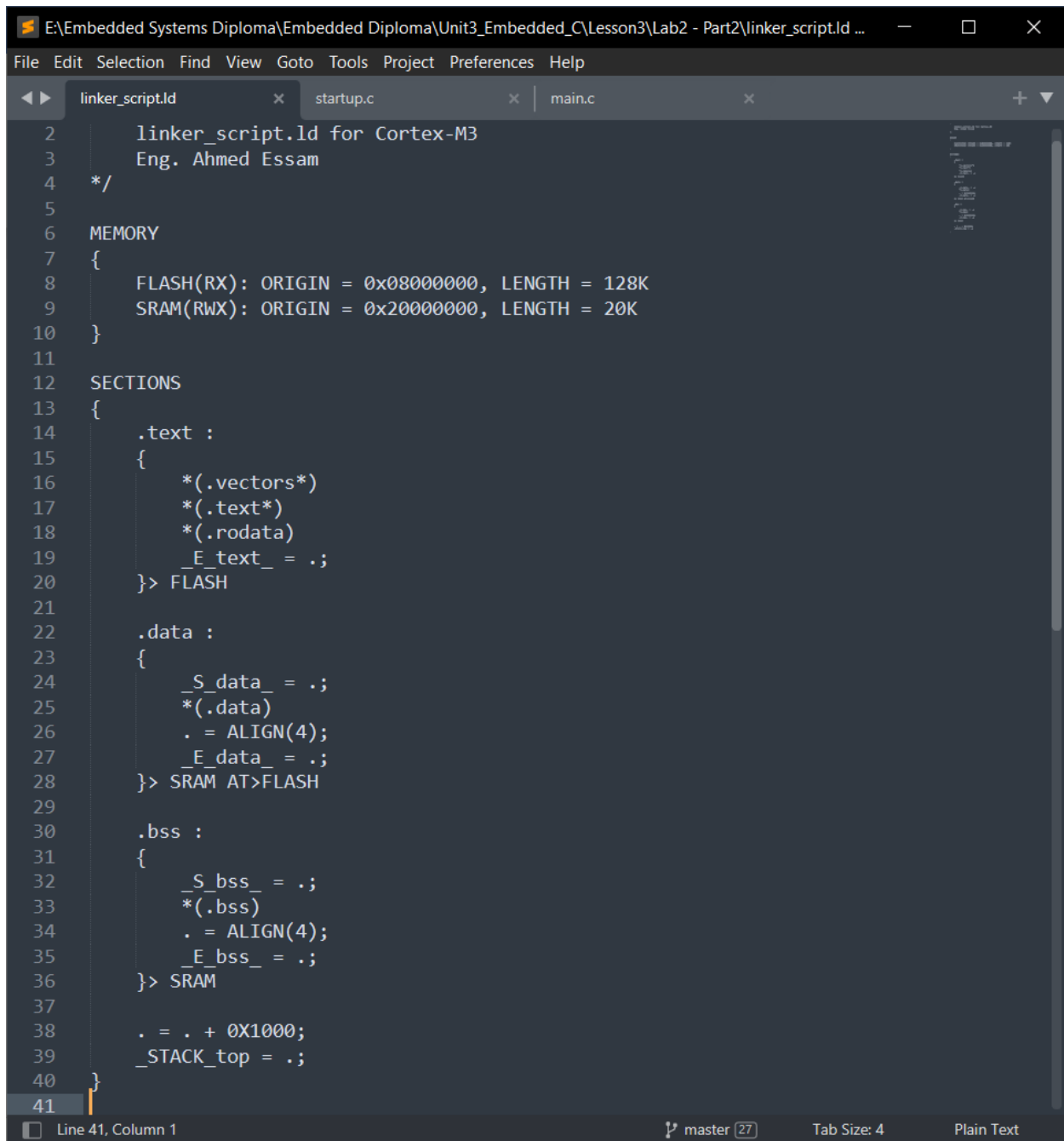
```
E:\Embedded Systems Diploma\Embedded Diploma\Unit3_Embedded_C\Lesson3\Lab2 - Part2\startup.c - Subli...
File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld x startup.c x main.c x + v

8  int main(void);
9  void Reset_Handler();
10 void Default_Handler(){
11     Reset_Handler();
12 }
13
14 void NMI_Handler()          __attribute__((weak, alias("Default_Handler")));
15 void H_Fault_Handler()      __attribute__((weak, alias("Default_Handler")));
16 void MM_Fault_Handler()     __attribute__((weak, alias("Default_Handler")));
17 void Bus_Fault_Handler()    __attribute__((weak, alias("Default_Handler")));
18 void Usage_Fault_Handler()  __attribute__((weak, alias("Default_Handler")));
19
20 extern uint32_t _STACK_top;
21
22 uint32_t vectors[] __attribute__((section(".vectors"))) = {
23     (uint32_t) &_STACK_top,
24     (uint32_t) &Reset_Handler,
25     (uint32_t) &NMI_Handler,
26     (uint32_t) &H_Fault_Handler,
27     (uint32_t) &MM_Fault_Handler,
28     (uint32_t) &Bus_Fault_Handler,
29     (uint32_t) &Usage_Fault_Handler
30 };
31
32 extern unsigned int _E_text_;
33 extern unsigned int _S_data_;
34 extern unsigned int _E_data_;
35 extern unsigned int _S_bss_;
36 extern unsigned int _E_bss_;
37
38 void Reset_Handler(){
39     //Copy .data from FLASH to SRAM
40     unsigned int data_SIZE =
41         (unsigned char*)&_E_data_ - (unsigned char*)&_S_data_;
42
43     unsigned char *P_src = (unsigned char*)&_E_text_;
44     unsigned char *P_dest = (unsigned char*)&_S_data_;
45
46     for(int i = 0; i < data_SIZE; i++){
47         *((unsigned char *)P_dest++) = *((unsigned char *)P_src++);
48     }
49
50     //Initialize .bss in SRAM by zeros
51     unsigned int bss_SIZE =
52         (unsigned char*)&_E_bss_ - (unsigned char*)&_S_bss_;
53
54     P_dest = (unsigned char*)&_S_bss_;
55
56     for(int i = 0; i < bss_SIZE; i++){
57         *((unsigned char *)P_dest++) = (unsigned char) 0;
58     }
59
60     //Proceed to main function
61     main();
62 }
```

Line 59, Column 1 master 27 Tab Size: 4 C

Linker_script.ld:



```
1  linker_script.ld for Cortex-M3
2  Eng. Ahmed Essam
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          *(.vectors*)
16          *(.text*)
17          *(.rodata)
18          _E_text_ = .;
19      }> FLASH
20
21      .data :
22      {
23          _S_data_ = .;
24          *(.data)
25          . = ALIGN(4);
26          _E_data_ = .;
27      }> SRAM AT>FLASH
28
29      .bss :
30      {
31          _S_bss_ = .;
32          *(.bss)
33          . = ALIGN(4);
34          _E_bss_ = .;
35      }> SRAM
36
37      . = . + 0X1000;
38      _STACK_top = .;
39
40  }
41
```

Line 41, Column 1 master (27) Tab Size: 4 Plain Text

Main.c:

```
E:\Embedded Systems Diploma\Embedded Diploma\Unit3_Embedded_C\Lesson3\Lab2 - Part2\main.c - Sublim...
File Edit Selection Find View Goto Tools Project Preferences Help

linker_script.ld x startup.c x main.c x + ▼

20 #include <stdint.h>
21
22 #define RCC_BASE 0x40021000
23 #define PortA_BASE 0x40010800
24 #define RCC_APB2ENR *(volatile uint32_t *) (RCC_BASE + 0x18)
25 #define GPIOA_CRH *(volatile uint32_t *) (PortA_BASE + 0x04)
26 #define GPIOA_ODR *(volatile uint32_t *) (PortA_BASE + 0x0C)
27 #define RCC_IOPAEN (1<<2)
28
29 typedef volatile unsigned int vuint32_t;
30
31 typedef union{
32     vuint32_t allFields;
33     struct{
34         vuint32_t reserved:13; //0-12
35         vuint32_t pin13:1;
36     } pin;
37 } R_ODR_t;
38
39 volatile R_ODR_t* R_ODR = (volatile R_ODR_t*)(PortA_BASE + 0x0C);
40 unsigned char g_var[3] = {1, 2, 3};
41 unsigned char const const_var[3] = {1, 2, 3};
42
43 int main(void)
44 {
45     RCC_APB2ENR |= RCC_IOPAEN;
46     GPIOA_CRH &= 0xFF0FFFFF;
47     GPIOA_CRH |= 0x00200000;
48
49     /* Loop forever */
50     for(;;){
51         //GPIOA_ODR |= 1<<13; //Set
52         R_ODR->pin.pin13 = 1;
53         for(int i = 0; i < 10000; i++);
54         //GPIOA_ODR &= ~(1<<13); //Clear
55         R_ODR->pin.pin13 = 0;
56         for(int i = 0; i < 10000; i++);
57     }
58 }
```

Line 48, Column 1 master 27 Tab Size: 4 C

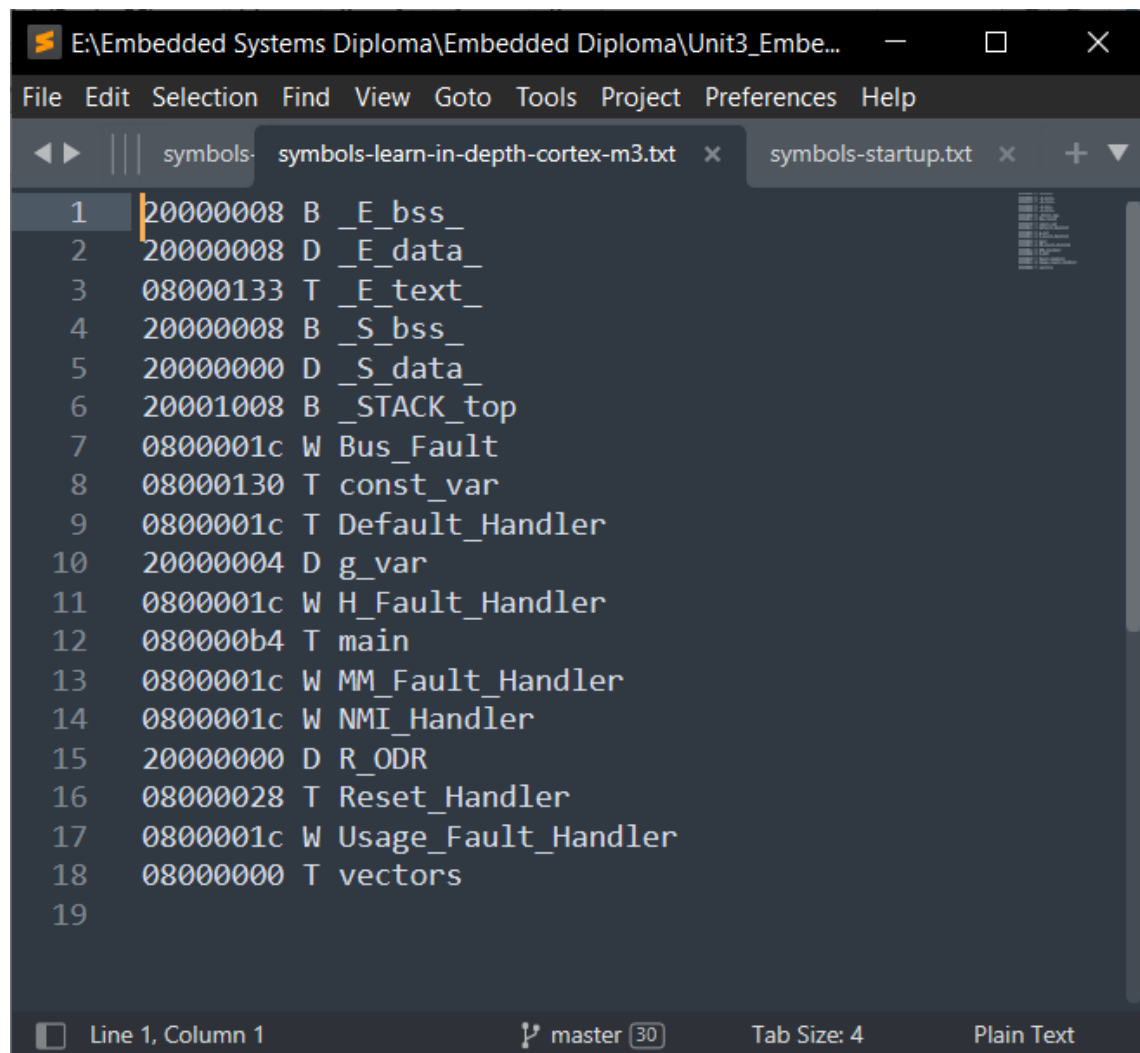
Symbols of startup.o:

```
1      U _E_bss_  
2      U _E_data_  
3      U _E_text_  
4      U _S_bss_  
5      U _S_data_  
6      U _STACK_top  
7 00000000 W Bus_Fault  
8 00000000 T Default_Handler  
9 00000000 W H_Fault_Handler  
10     U main  
11 00000000 W MM_Fault_Handler  
12 00000000 W NMI_Handler  
13 0000000c T Reset_Handler  
14 00000000 W Usage_Fault_Handler  
15 00000000 D vectors  
16
```

Symbols of main.o:

```
1 00000000 R const_var  
2 00000004 D g_var  
3 00000000 T main  
4 00000000 D R_ODR  
5
```

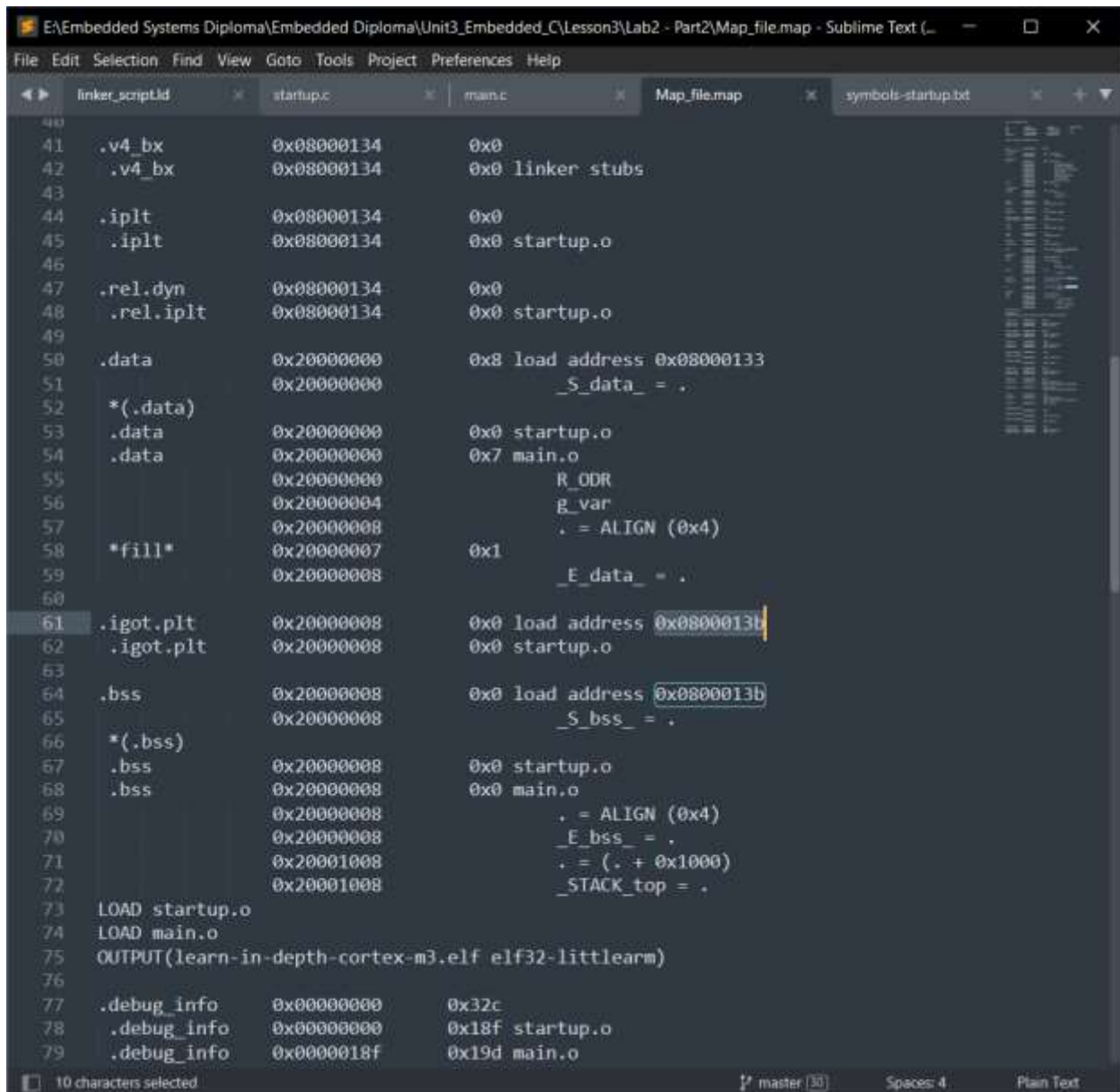
Symbols of learn-in-depth-cortex-m3.elf:



```
1 20000008 B _E_bss_  
2 20000008 D _E_data_  
3 08000133 T _E_text_  
4 20000008 B _S_bss_  
5 20000000 D _S_data_  
6 20001008 B _STACK_top  
7 0800001c W Bus_Fault  
8 08000130 T const_var  
9 0800001c T Default_Handler  
10 20000004 D g_var  
11 0800001c W H_Fault_Handler  
12 080000b4 T main  
13 0800001c W MM_Fault_Handler  
14 0800001c W NMI_Handler  
15 20000000 D R_ODR  
16 08000028 T Reset_Handler  
17 0800001c W Usage_Fault_Handler  
18 08000000 T vectors  
19
```

Line 1, Column 1 master 30 Tab Size: 4 Plain Text

Mapfile.map:



```
40
41 .v4_bx      0x08000134      0x0
42 .v4_bx      0x08000134      0x0 linker stubs
43
44 .iplt       0x08000134      0x0
45 .iplt       0x08000134      0x0 startup.o
46
47 .rel.dyn    0x08000134      0x0
48 .rel.iplt   0x08000134      0x0 startup.o
49
50 .data       0x20000000      0x8 load address 0x08000133
51             0x20000000      _S_data_ = .
52 *(.data)
53 .data       0x20000000      0x0 startup.o
54 .data       0x20000000      0x7 main.o
55             0x20000000      R_ODR
56             0x20000004      g_var
57             0x20000008      . = ALIGN (0x4)
58 *fill*      0x20000007      0x1
59             0x20000008      _E_data_ = .
60
61 .igot.plt    0x20000008      0x0 load address 0x0800013b
62 .igot.plt    0x20000008      0x0 startup.o
63
64 .bss         0x20000008      0x0 load address 0x0800013b
65             0x20000008      _S_bss_ = .
66 *(.bss)
67 .bss         0x20000008      0x0 startup.o
68 .bss         0x20000008      0x0 main.o
69             0x20000008      . = ALIGN (0x4)
70             0x20000008      _E_bss_ = .
71             0x20001008      . = (. + 0x1000)
72             0x20001008      _STACK_top = .
73 LOAD startup.o
74 LOAD main.o
75 OUTPUT(learn-in-depth-cortex-m3.elf elf32-littlearm)
76
77 .debug_info  0x00000000      0x32c
78 .debug_info  0x00000000      0x18f startup.o
79 .debug_info  0x0000018f      0x19d main.o
```

10 characters selected master [30] Spaces: 4 Plain Text

Proteus screenshot:

