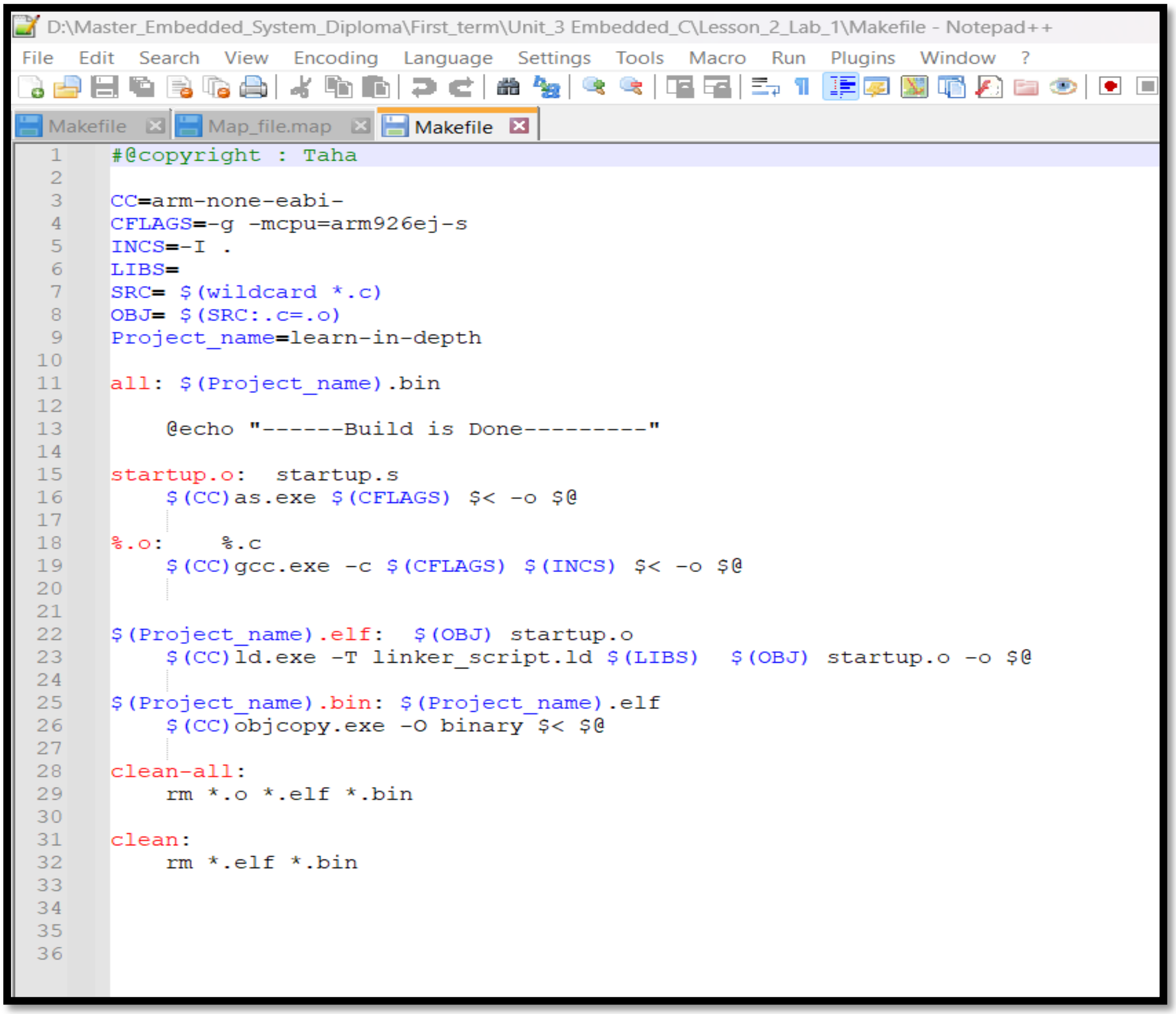
Embedded_C_Lesson_3

Created by : Eng. Taha Mohamed

Under Supervision : Eng. Keroles Shenouda

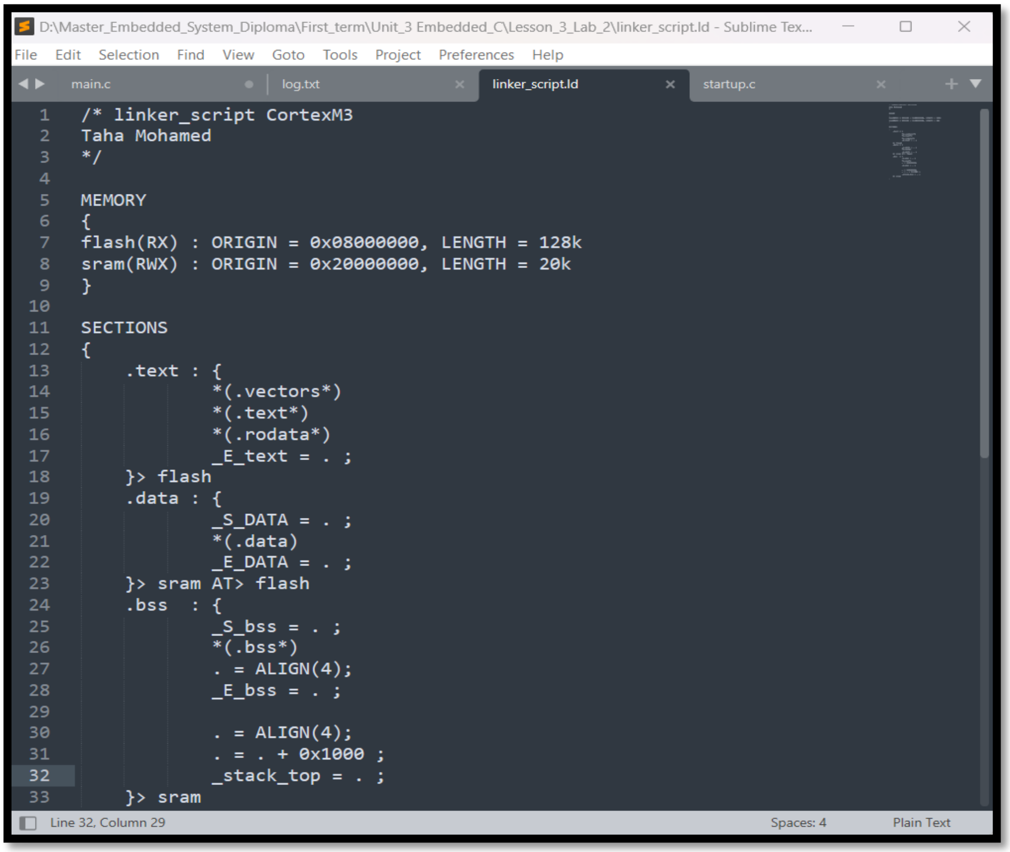
Lab-2 Assignment

1- Makefile of lab-1


D:\Master_Embedded_System_Diploma\First_term\Unit_3 Embedded_C\Lesson_2_Lab_1\Makefile - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Makefile
Map_file.map
Makefile

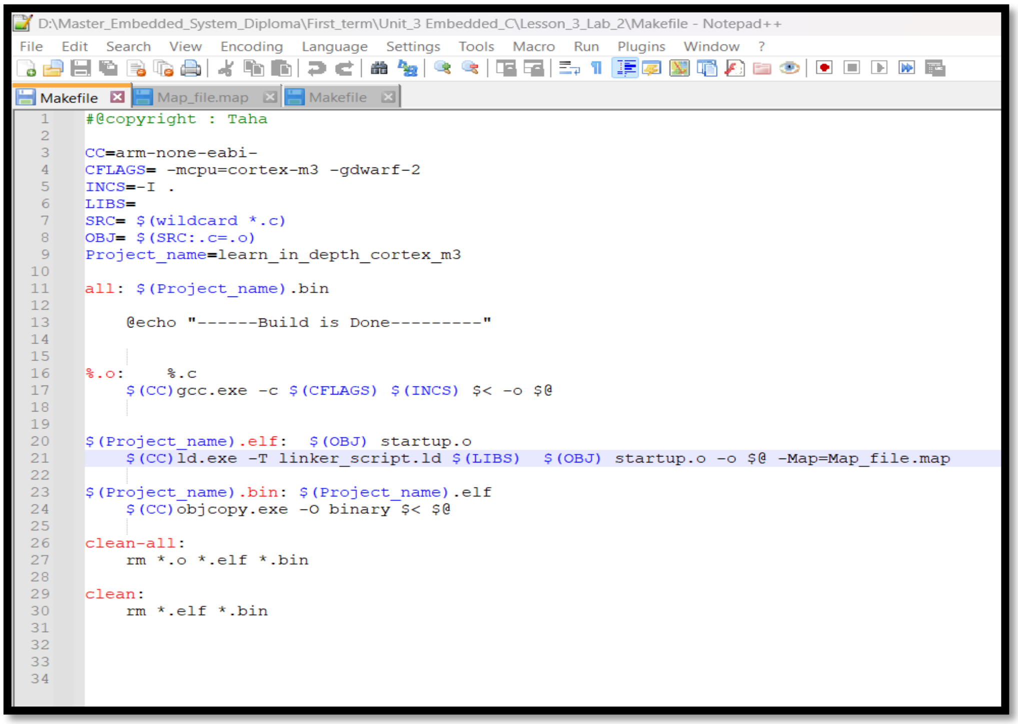
```
#@copyright : Taha

CC=arm-none-eabi-
CFLAGS=-g -mcpu=arm926ej-s
INCS=-I .
LIBS=
SRC= $(wildcard *.c)
OBJ= $(SRC:.c=.o)
Project_name=learn-in-depth

all: $(Project_name).bin

	@echo "------Build is Done---------"

startup.o:  startup.s
	$(CC)as.exe $(CFLAGS) $< -o $@

%.o:    %.c
	$(CC)gcc.exe -c $(CFLAGS) $(INCS) $< -o $@


$(Project_name).elf:  $(OBJ) startup.o
	$(CC)ld.exe -T linker_script.ld $(LIBS)  $(OBJ) startup.o -o $@

$(Project_name).bin: $(Project_name).elf
	$(CC)objcopy.exe -O binary $< $@

clean-all:
	rm *.o *.elf *.bin

clean:
	rm *.elf *.bin
```

2- Linker_script.ld


D:\Master_Embedded_System_Diploma\First_term\Unit_3 Embedded_C\Lesson_3_Lab_2\linker_script.ld - Sublime Tex...
File Edit Selection Find View Goto Tools Project Preferences Help
main.c
log.txt
linker_script.ld
startup.c
Line 32, Column 29
Spaces: 4
Plain Text

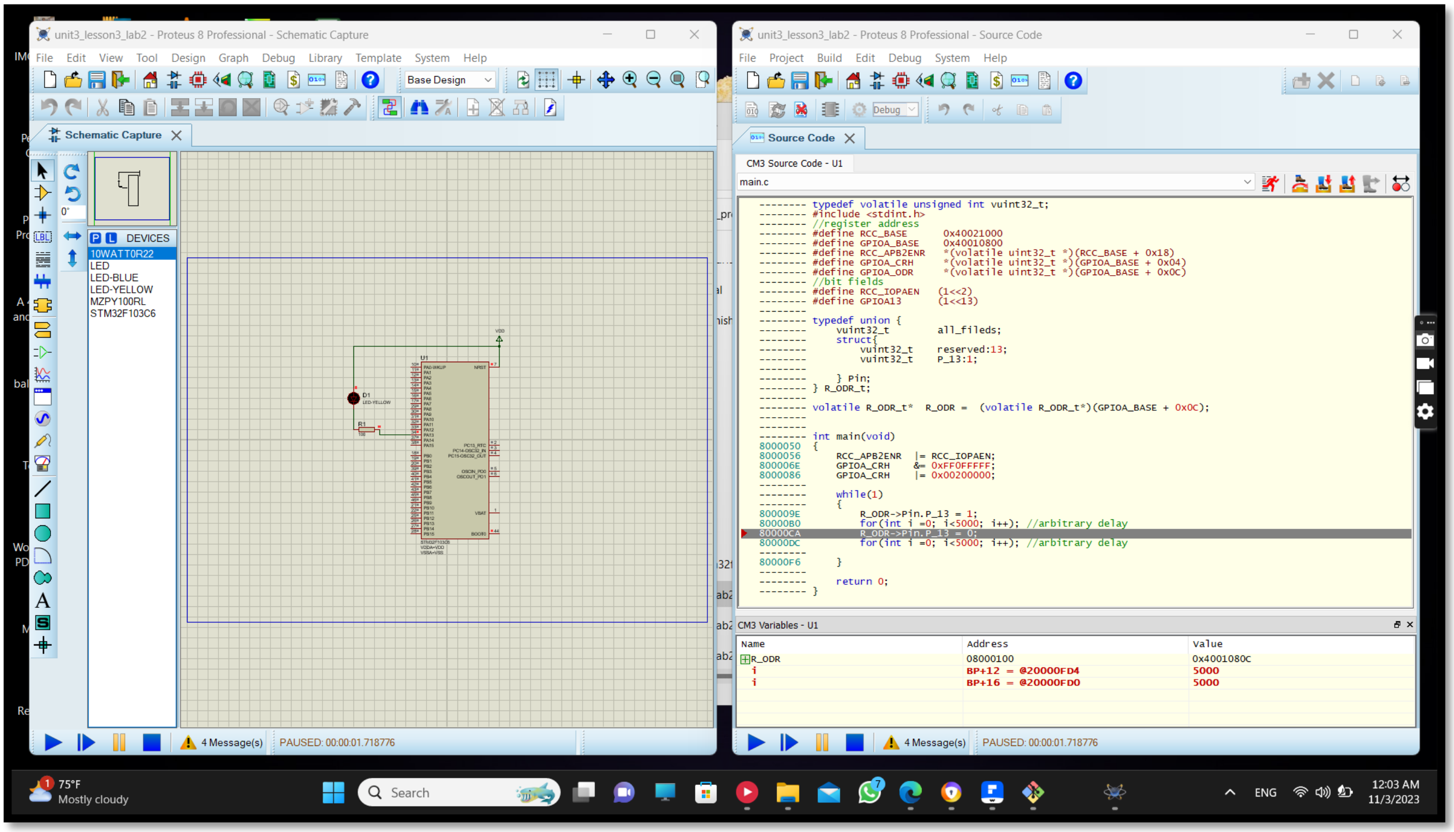
```
/* linker_script CortexM3
Taha Mohamed
*/

MEMORY
{
flash(RX) : ORIGIN = 0x08000000, LENGTH = 128k
sram(RWX) : ORIGIN = 0x20000000, LENGTH = 20k
}

SECTIONS
{
    .text : {
            *(.vectors*)
            *(.text*)
            *(.rodata*)
            _E_text = . ;
    }> flash
    .data : {
            _S_DATA = . ;
            *(.data)
            _E_DATA = . ;
    }> sram AT> flash
    .bss  : {
            _S_bss = . ;
            *(.bss*)
            . = ALIGN(4);
            _E_bss = . ;

            . = ALIGN(4);
            . = . + 0x1000 ;
            _stack_top = . ;
    }> sram
```

3- Makefile


D:\Master_Embedded_System_Diploma\First_term\Unit_3 Embedded_C\Lesson_3_Lab_2\Makefile - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Makefile
Map_file.map
Makefile

```
#@copyright : Taha

CC=arm-none-eabi-
CFLAGS= -mcpu=cortex-m3 -gdwarf-2
INCS=-I .
LIBS=
SRC= $(wildcard *.c)
OBJ= $(SRC:.c=.o)
Project_name=learn_in_depth_cortex_m3

all: $(Project_name).bin

    @echo "------Build is Done---------"


%.o:    %.c
    $(CC)gcc.exe -c $(CFLAGS) $(INCS) $< -o $@


$(Project_name).elf:  $(OBJ) startup.o
    $(CC)ld.exe -T linker_script.ld $(LIBS)  $(OBJ) startup.o -o $@ -Map=Map_file.map

$(Project_name).bin: $(Project_name).elf
    $(CC)objcopy.exe -O binary $< $@

clean-all:
    rm *.o *.elf *.bin

clean:
    rm *.elf *.bin
```

4- Run/Debug

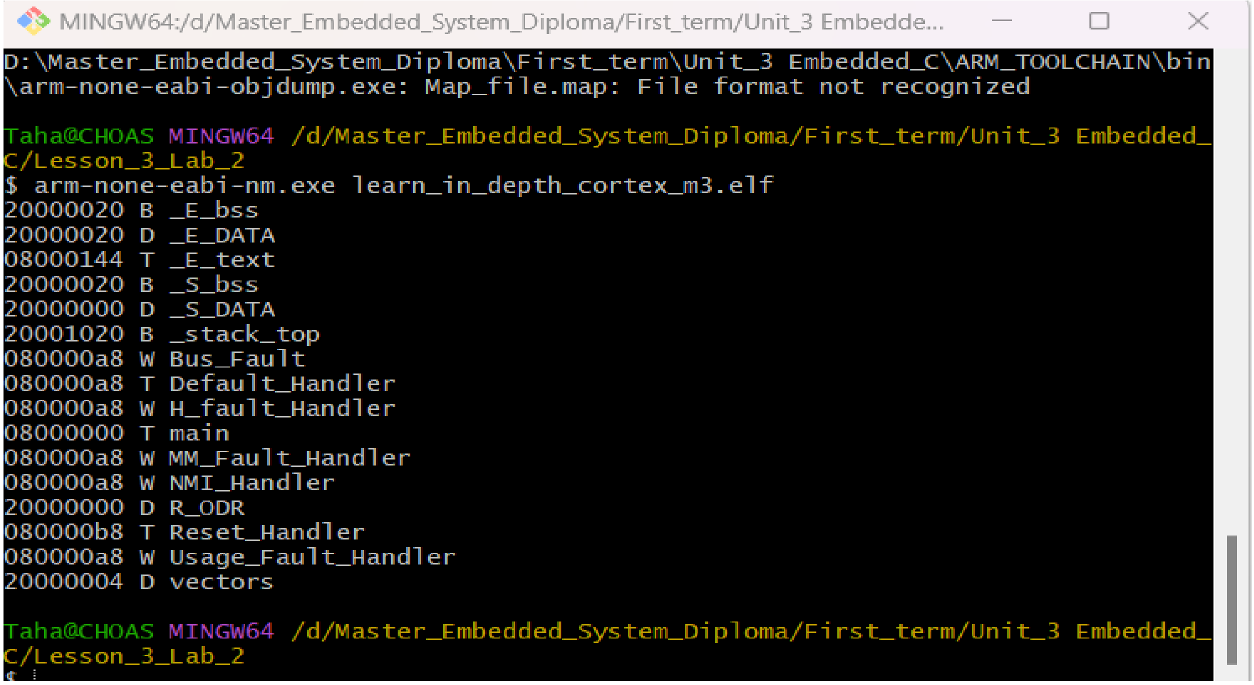


```
-------- typedef volatile unsigned int vuint32_t;
-------- #include <stdint.h>
-------- //register address
-------- #define RCC_BASE       0x40021000
-------- #define GPIOA_BASE     0x40010800
-------- #define RCC_APB2ENR    *(volatile uint32_t *)(RCC_BASE + 0x18)
-------- #define GPIOA_CRH      *(volatile uint32_t *)(GPIOA_BASE + 0x04)
-------- #define GPIOA_ODR      *(volatile uint32_t *)(GPIOA_BASE + 0x0C)
-------- //bit fields
-------- #define RCC_IOPAEN    (1<<2)
-------- #define GPIOA13       (1<<13)
--------
-------- typedef union {
--------     vuint32_t       all_fileds;
--------     struct{
--------         vuint32_t    reserved:13;
--------         vuint32_t    P_13:1;
--------
--------     } Pin;
-------- } R_ODR_t;
--------
-------- volatile R_ODR_t*  R_ODR =  (volatile R_ODR_t*)(GPIOA_BASE + 0x0C);
--------
--------
-------- int main(void)
8000050  {
8000056      RCC_APB2ENR  |= RCC_IOPAEN;
800006E      GPIOA_CRH    &= 0xFF0FFFFF;
8000086      GPIOA_CRH    |= 0x00200000;
--------
--------     while(1)
--------     {
800009E          R_ODR->Pin.P_13 = 1;
80000B0          for(int i =0; i<5000; i++); //arbitrary delay
80000CA          R_ODR->Pin.P_13 = 0;
80000DC          for(int i =0; i<5000; i++); //arbitrary delay
--------
80000F6      }
--------
--------     return 0;
-------- }
```

Name	Address	Value
R_ODR	08000100	0x4001080C
i	BP+12 = @20000FD4	5000
i	BP+16 = @20000FD0	5000

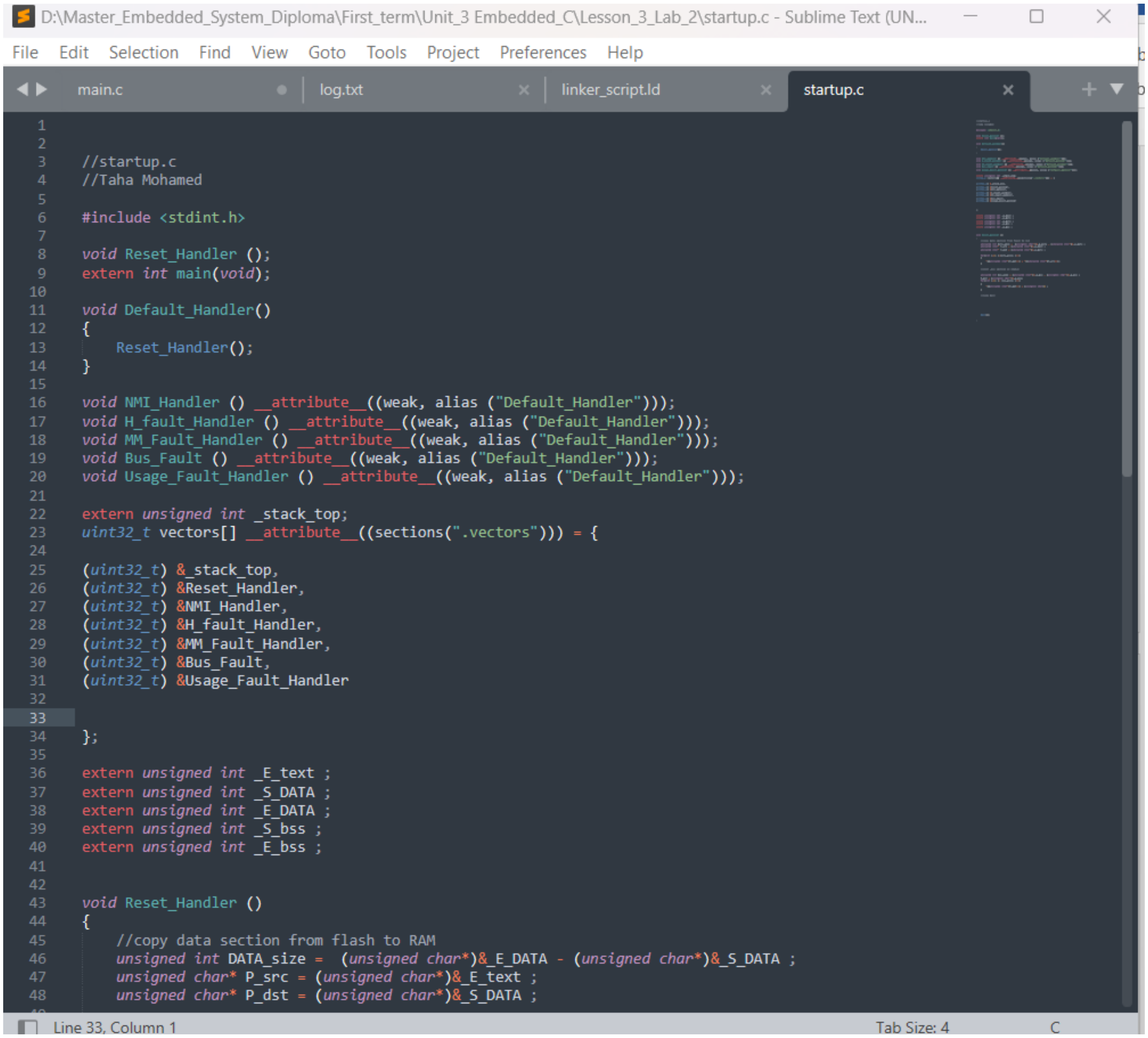
4 Message(s)   PAUSED: 00:00:01.718776

5- Text file showing symbols of each object


MINGW64:/d/Master_Embedded_System_Diploma/First_term/Unit_3 Embedde...

```
D:\Master_Embedded_System_Diploma\First_term\Unit_3 Embedded_C\ARM_TOOLCHAIN\bin
\arm-none-eabi-objdump.exe: Map_file.map: File format not recognized

Taha@CHOAS MINGW64 /d/Master_Embedded_System_Diploma/First_term/Unit_3 Embedded_
C/Lesson_3_Lab_2
$ arm-none-eabi-nm.exe learn_in_depth_cortex_m3.elf
20000020 B _E_bss
20000020 D _E_DATA
08000144 T _E_text
20000020 B _S_bss
20000000 D _S_DATA
20001020 B _stack_top
080000a8 W Bus_Fault
080000a8 T Default_Handler
080000a8 W H_fault_Handler
08000000 T main
080000a8 W MM_Fault_Handler
080000a8 W NMI_Handler
20000000 D R_ODR
080000b8 T Reset_Handler
080000a8 W Usage_Fault_Handler
20000004 D vectors

Taha@CHOAS MINGW64 /d/Master_Embedded_System_Diploma/First_term/Unit_3 Embedded_
C/Lesson_3_Lab_2
$
```

6- Startup.c



D:\Master_Embedded_System_Diploma\First_term\Unit_3 Embedded_C\Lesson_3_Lab_2\startup.c - Sublime Text (UN...
File Edit Selection Find View Goto Tools Project Preferences Help
main.c
log.txt
linker_script.ld
startup.c

```c


//startup.c
//Taha Mohamed

#include <stdint.h>

void Reset_Handler ();
extern int main(void);

void Default_Handler()
{
	Reset_Handler();
}

void NMI_Handler () __attribute__((weak, alias ("Default_Handler")));
void H_fault_Handler () __attribute__((weak, alias ("Default_Handler")));
void MM_Fault_Handler () __attribute__((weak, alias ("Default_Handler")));
void Bus_Fault () __attribute__((weak, alias ("Default_Handler")));
void Usage_Fault_Handler () __attribute__((weak, alias ("Default_Handler")));

extern unsigned int _stack_top;
uint32_t vectors[] __attribute__((sections(".vectors"))) = {

(uint32_t) &_stack_top,
(uint32_t) &Reset_Handler,
(uint32_t) &NMI_Handler,
(uint32_t) &H_fault_Handler,
(uint32_t) &MM_Fault_Handler,
(uint32_t) &Bus_Fault,
(uint32_t) &Usage_Fault_Handler



};

extern unsigned int _E_text ;
extern unsigned int _S_DATA ;
extern unsigned int _E_DATA ;
extern unsigned int _S_bss ;
extern unsigned int _E_bss ;


void Reset_Handler ()
{
	//copy data section from flash to RAM
	unsigned int DATA_size =  (unsigned char*)&_E_DATA - (unsigned char*)&_S_DATA ;
	unsigned char* P_src = (unsigned char*)&_E_text ;
	unsigned char* P_dst = (unsigned char*)&_S_DATA ;
```