Files needed to build and compile the main.c file:

```
CC= arm-none-eabi-
MACH= cortex-m4
CFLAGS= -mcpu=$(MACH) -mthumb -gdwarf-2 -g
INCS= -I.
LIBS=
O_OPERATOR=-
O_FILE= main.o
OBJ= main.o Startup.o
PROJECT_NAME=Cortex-m4_lab3

all: $(PROJECT_NAME).bin



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

$(PROJECT_NAME).elf:LinkerScript.ld $(OBJ)
	$(CC)ld -T $(LIBS) $^ -o $@  -Map=$(PROJECT_NAME).map


$(PROJECT_NAME).bin:$(PROJECT_NAME).elf
	$(CC)objcopy $^ -O binary $@

nm:
	$(CC)nm   $(PROJECT_NAME).elf

O_analyze:
	$(CC)objdump $(O_OPERATOR) $(O_FILE)

clean:
	rm *.o *.bin
clean_all:
	rm *.o *.bin *.elf *.map

sim:
	qemu-system-arm -M versatilepb -m 128M -nographic -kernel $(PROJECT_NAME).bin
start_debug:
	qemu-system-arm -M versatilepb -m 128M -nographic -s -S -kernel $(PROJECT_NAME).bin
load:
	openocd -f 'C:\Users\Sence79\Desktop\xpack-openocd-0.10.0-15\scripts\board\stm32f4discovery.cfg'

```

```
ENTRY(Reset_Handler)

MEMORY{
    flash (rx)  : ORIGIN = 0x00000000, LENGTH = 128k
    SRAM (rwx)  : ORIGIN = 0x20000000, LENGTH = 20k
}
SECTIONS{

    .text : {
        *(.vectors)
        *(.text*)
        *(.rodata)
        _E_text = .;    /*tracking the flash*/
    }> flash

    .data : {
        _S_data = .;
        *(.data)      /*is put originally on the flash*/
        _E_data = .;    /*tracking the ram*/
    }> SRAM AT> flash

    .bss : {
        _S_bss = .;
        *(.bss)
        _E_bss = .;
       }>SRAM

}
```

The building process:

```
Sence79@DESKTOP-JJFQ8S2 MINGW32 ~/Desktop/lab3
$ make
arm-none-eabi-gcc -mcpu=cortex-m4 -mthumb -gdwarf-2 -g  -c -I. main.c -o main.o
arm-none-eabi-gcc -mcpu=cortex-m4 -mthumb -gdwarf-2 -g  -c -I. Startup.c -o Star
tup.o
arm-none-eabi-ld -T  LinkerScript.ld main.o Startup.o -o Cortex-m4_lab3.elf  -Ma
p=Cortex-m4_lab3.map
arm-none-eabi-objcopy Cortex-m4_lab3.elf -O binary Cortex-m4_lab3.bin

Sence79@DESKTOP-JJFQ8S2 MINGW32 ~/Desktop/lab3
$ ls
 Cortex-m4_lab3.axf   Cortex-m4_lab3.map   main.o           'Screenshot (102).png'
 Cortex-m4_lab3.bin   LinkerScript.ld      makefile          Startup.c
 Cortex-m4_lab3.elf   main.c               Platform_Types.h  Startup.o
```

Analyzing some .o files :

```
$ make O_analyze
arm-none-eabi-objdump -h  Cortex-m4_lab3.elf

Cortex-m4_lab3.elf:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000208  00000000  00000000  00008000  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .bss          00000400  20000000  00000208  00010000  2**2
                  ALLOC
  2 .debug_info   00000250  00000000  00000000  00008208  2**0
                  CONTENTS, READONLY, DEBUGGING
  3 .debug_abbrev 0000013a  00000000  00000000  00008458  2**0
                  CONTENTS, READONLY, DEBUGGING
  4 .debug_loc    0000009c  00000000  00000000  00008592  2**0
                  CONTENTS, READONLY, DEBUGGING
  5 .debug_aranges 00000040  00000000  00000000  0000862e  2**0
                  CONTENTS, READONLY, DEBUGGING
  6 .debug_line   000000f2  00000000  00000000  0000866e  2**0
                  CONTENTS, READONLY, DEBUGGING
  7 .debug_str    0000013f  00000000  00000000  00008760  2**0
                  CONTENTS, READONLY, DEBUGGING
  8 .comment      00000011  00000000  00000000  0000889f  2**0
                  CONTENTS, READONLY
  9 .ARM.attributes 00000033  00000000  00000000  000088b0  2**0
                  CONTENTS, READONLY
 10 .debug_frame  00000078  00000000  00000000  000088e4  2**2
                  CONTENTS, READONLY, DEBUGGING
```

```
Sence79@DESKTOP-JJFQ8S2 MINGW32 ~/Desktop/lab3
$ make nm
arm-none-eabi-nm   Cortex-m4_lab3.elf
20000400 B _E_bss
20000000 T _E_data
00000208 T _E_text
20000000 B _S_bss
20000000 T _S_data
0000014c W Bus_Fault_Handler
0000014c T Default_Handler
0000014c W H_Fault_Handler
0000001c T main
0000014c W MM_Fault_Handler
0000014c W NMI_Handler
00000158 T Reset_Handler
20000000 b stack
0000014c W Usage_Fault_Handler
00000000 T vectors
```