In this lab, I will write a bare-metal software to toggle led is connected to GPIO port C13, to make a GPIO toggling in STM32. After reading its specs I Found that:

From reading the specs, I have all what I need to write application files.

1. Stm32pins.c (for setting clock control settings).
2. Stm32pins.h (including typedef, macros and prototypes).
3. Main.c (for main process of project).

Next step it to read also its specs because some parts of startup code dependent on the target processor.

The startup file has been written in two ways that feature is provided only for cortex-M3 and its family. So, I wrote the 2 files.

1. Startup.s
2. Startup.c

The 3rd step is to write linker_script.ld file.

The last step is to build the project and generate the .elf and .bin files.

I have written a make file to automate building process.

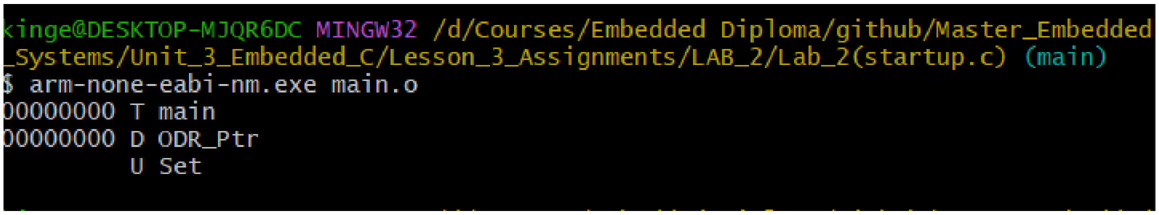
All the above information that I need it, to write a bare-metal software to toggle the LED start.

Appendix

A- Symbols in output objects and elf file:

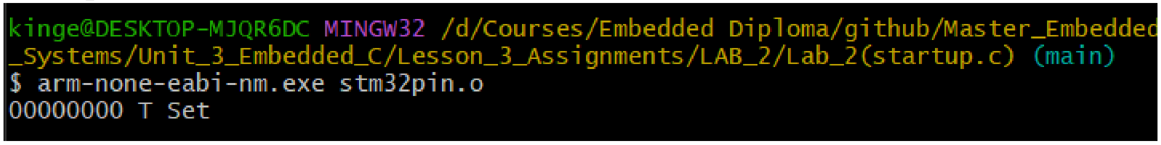
1. For startup.c version

1.1 main.o



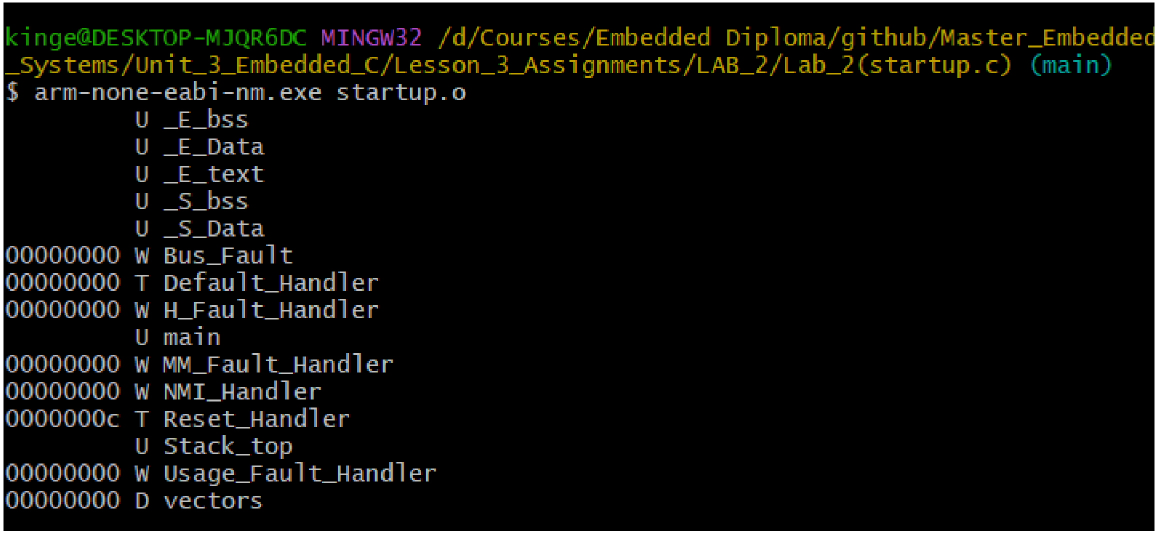
```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.c) (main)
$ arm-none-eabi-nm.exe main.o
00000000 T main
00000000 D ODR_Ptr
         U Set
```

1.2 stm32pin.o



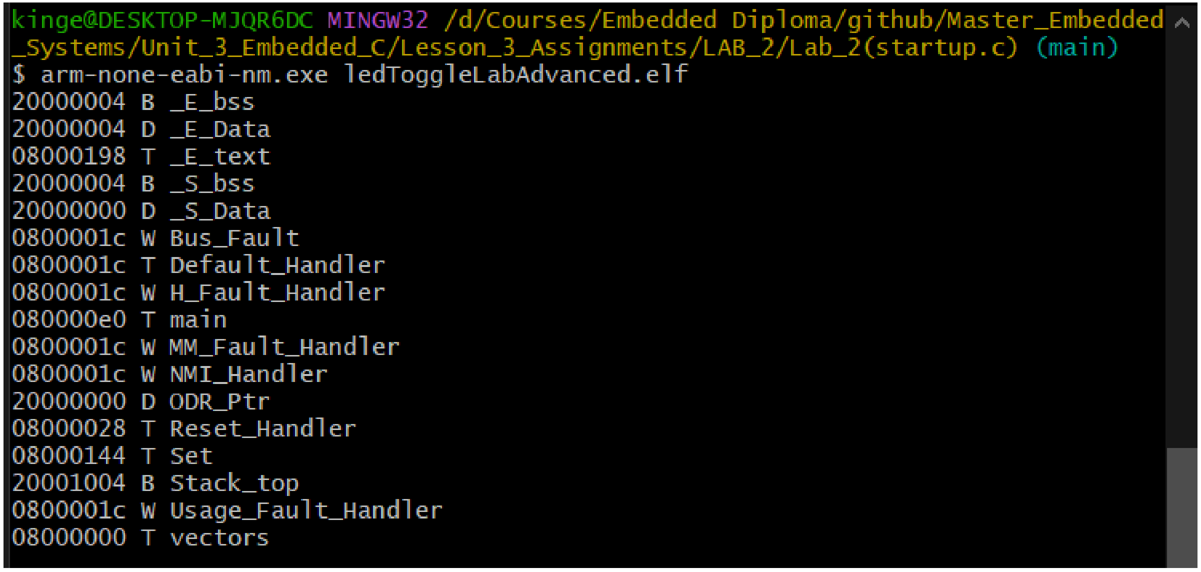
```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.c) (main)
$ arm-none-eabi-nm.exe stm32pin.o
00000000 T Set
```

1.3 startup.o



```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.c) (main)
$ arm-none-eabi-nm.exe startup.o
         U _E_bss
         U _E_Data
         U _E_text
         U _S_bss
         U _S_Data
00000000 W Bus_Fault
00000000 T Default_Handler
00000000 W H_Fault_Handler
         U main
00000000 W MM_Fault_Handler
00000000 W NMI_Handler
0000000c T Reset_Handler
         U Stack_top
00000000 W Usage_Fault_Handler
00000000 D vectors
```

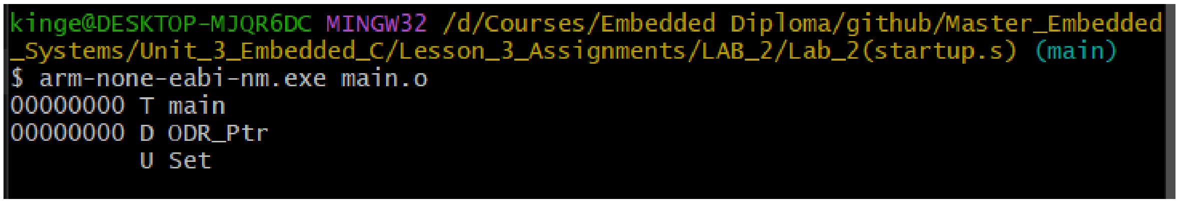
1.4 ledToggleLabAdvanced.elf



```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.c) (main)
$ arm-none-eabi-nm.exe ledToggleLabAdvanced.elf
20000004 B _E_bss
20000004 D _E_Data
08000198 T _E_text
20000004 B _S_bss
20000000 D _S_Data
0800001c W Bus_Fault
0800001c T Default_Handler
0800001c W H_Fault_Handler
080000e0 T main
0800001c W MM_Fault_Handler
0800001c W NMI_Handler
20000000 D ODR_Ptr
08000028 T Reset_Handler
08000144 T Set
20001004 B Stack_top
0800001c W Usage_Fault_Handler
08000000 T vectors
```

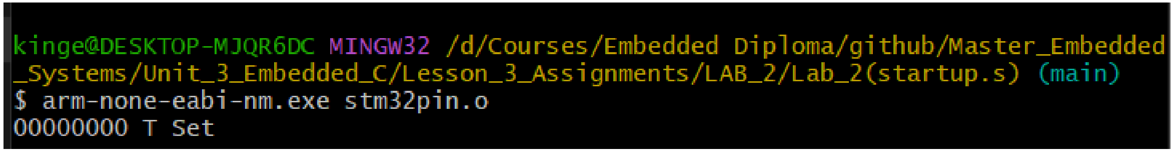
2. **For startup.s version**

2.1 main.o



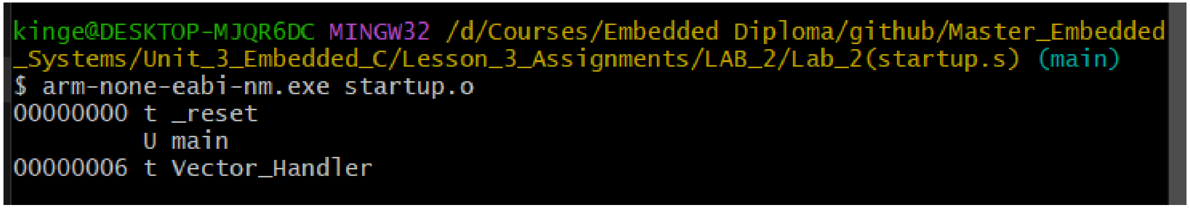
```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.s) (main)
$ arm-none-eabi-nm.exe main.o
00000000 T main
00000000 D ODR_Ptr
         U Set
```

2.2 stm32pin.o



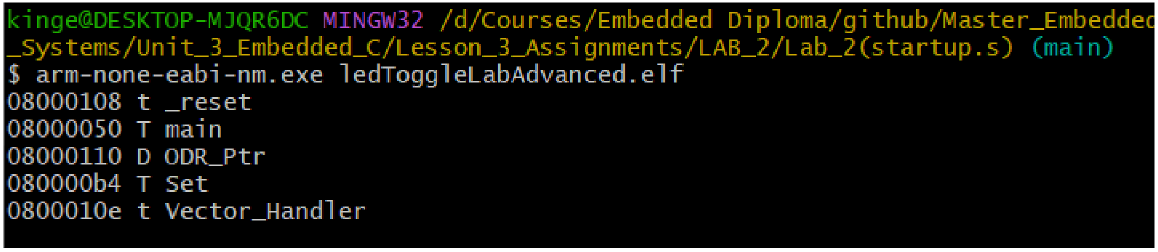
```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.s) (main)
$ arm-none-eabi-nm.exe stm32pin.o
00000000 T Set
```

2.3 startup.o



```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embedded
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.s) (main)
$ arm-none-eabi-nm.exe startup.o
00000000 t _reset
         U main
00000006 t Vector_Handler
```

2.4 ledToggleLabAdvanced.elf



```
kinge@DESKTOP-MJQR6DC MINGW32 /d/Courses/Embedded Diploma/github/Master_Embeddec
_Systems/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2/Lab_2(startup.s) (main)
$ arm-none-eabi-nm.exe ledToggleLabAdvanced.elf
08000108 t _reset
08000050 T main
08000110 D ODR_Ptr
080000b4 T Set
0800010e t Vector_Handler
```

B- **You can see all (mapfiles, C codes, startups, linker_scripts and makefile) in my github repo.**

https://github.com/aaref5720/Master_Embedded_Systems/tree/main/Unit_3_Embedded_C/Lesson_3_Assignments/LAB_2