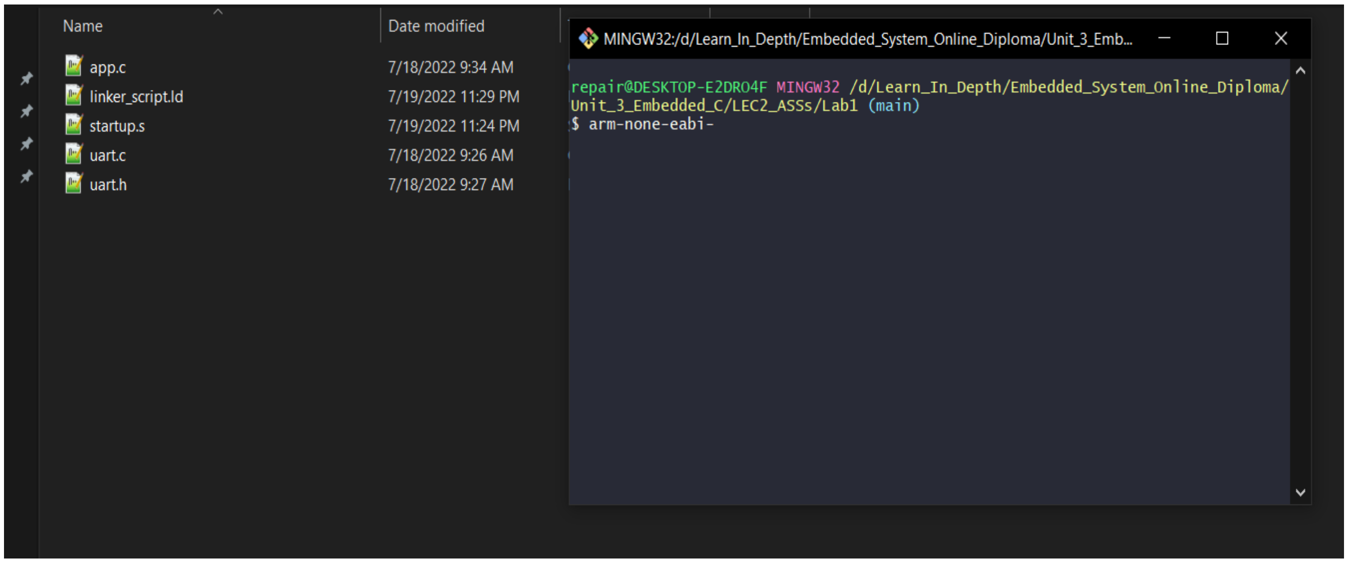
LAB1

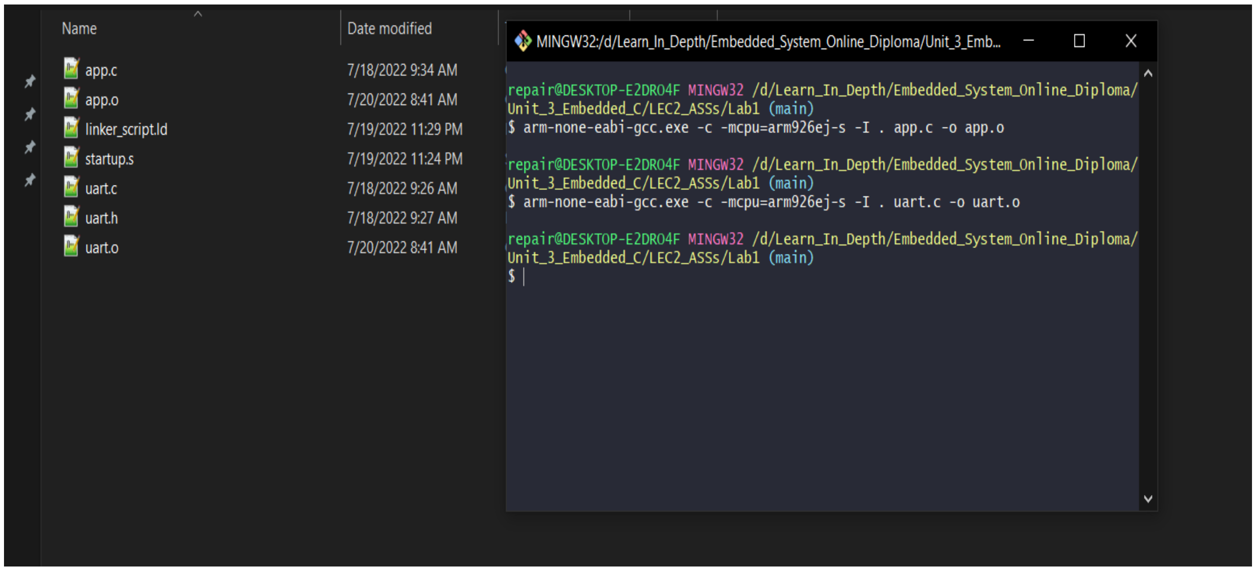


Name	Date modified
app.c	7/18/2022 9:34 AM
linker_script.ld	7/19/2022 11:29 PM
startup.s	7/19/2022 11:24 PM
uart.c	7/18/2022 9:26 AM
uart.h	7/18/2022 9:27 AM

MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

```
repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-
```

$ Compile uart.c & app.c



Name	Date modified
app.c	7/18/2022 9:34 AM
app.o	7/20/2022 8:41 AM
linker_script.ld	7/19/2022 11:29 PM
startup.s	7/19/2022 11:24 PM
uart.c	7/18/2022 9:26 AM
uart.h	7/18/2022 9:27 AM
uart.o	7/20/2022 8:41 AM

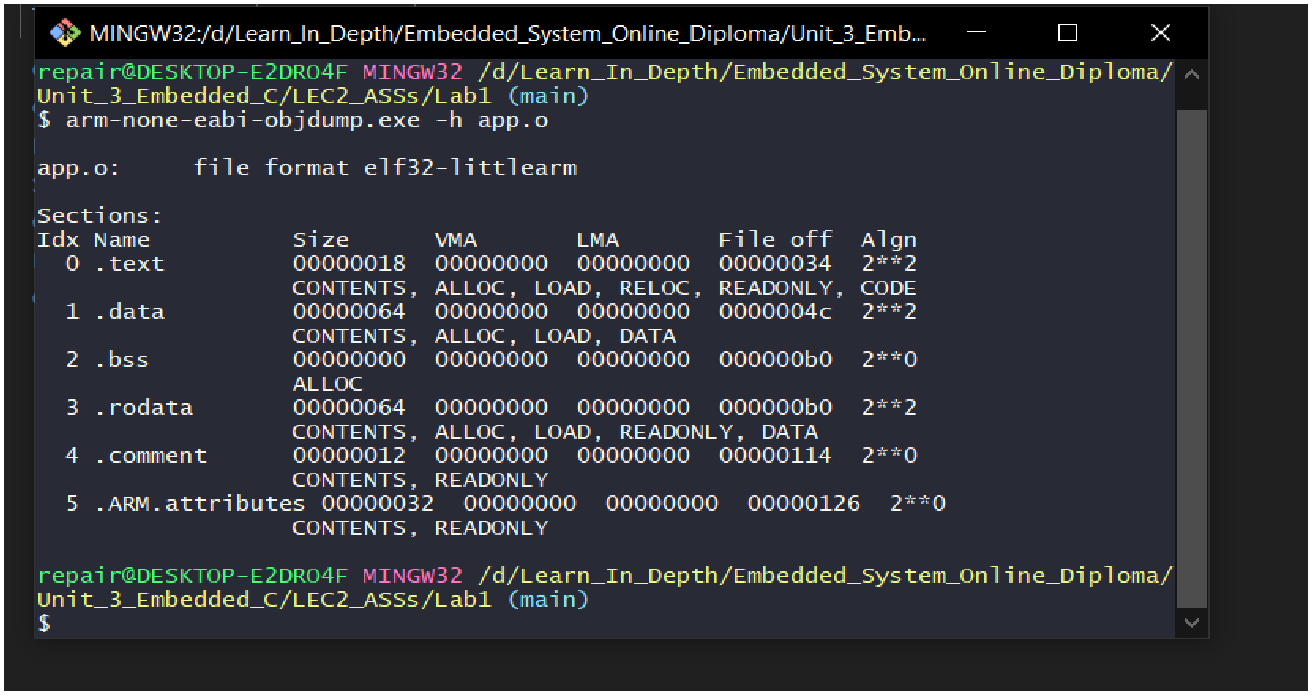
MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

```
repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-gcc.exe -c -mcpu=arm926ej-s -I . app.c -o app.o

repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-gcc.exe -c -mcpu=arm926ej-s -I . uart.c -o uart.o

repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ |
```

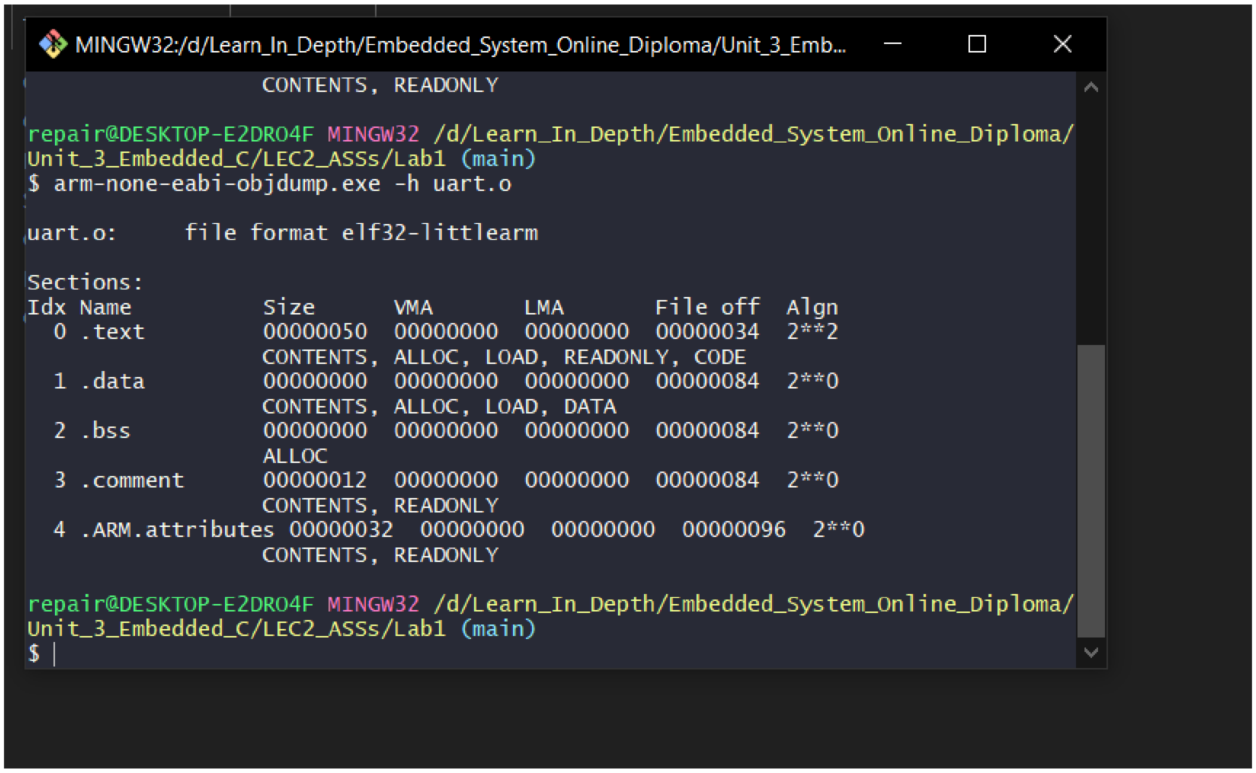
Note : This is relocatable object file

So, if we show the memory segments of the object files , we will see that the addresses (VMA &LMA) are zeros .

We will use object utility → objdum → -h



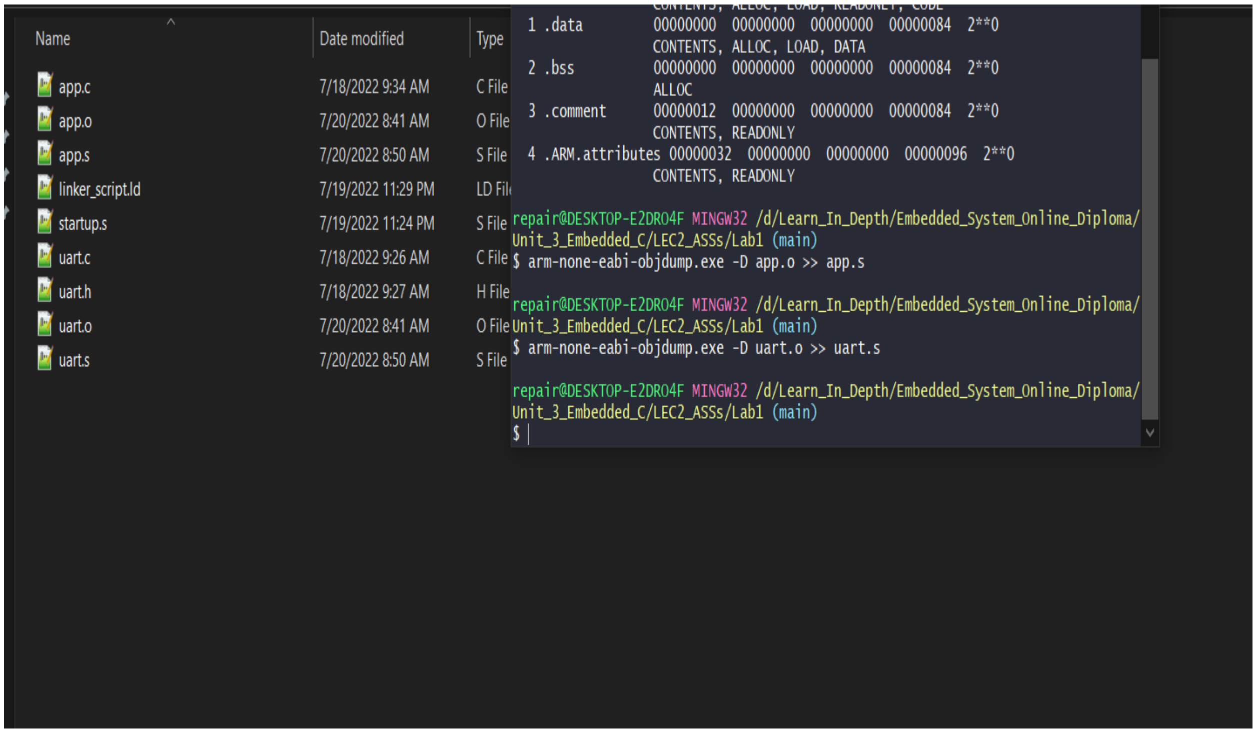
```
repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objdump.exe -h app.o

app.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000018  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000064  00000000  00000000  0000004c  2**2
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  000000b0  2**0
                  ALLOC
  3 .rodata       00000064  00000000  00000000  000000b0  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 .comment      00000012  00000000  00000000  00000114  2**0
                  CONTENTS, READONLY
  5 .ARM.attributes 00000032  00000000  00000000  00000126  2**0
                  CONTENTS, READONLY

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$
```



```
                  CONTENTS, READONLY

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objdump.exe -h uart.o

uart.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000050  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data         00000000  00000000  00000000  00000084  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000084  2**0
                  ALLOC
  3 .comment      00000012  00000000  00000000  00000084  2**0
                  CONTENTS, READONLY
  4 .ARM.attributes 00000032  00000000  00000000  00000096  2**0
                  CONTENTS, READONLY

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ |
```

$ The Disassembly of the object files

We will use object utility → objdum → -D

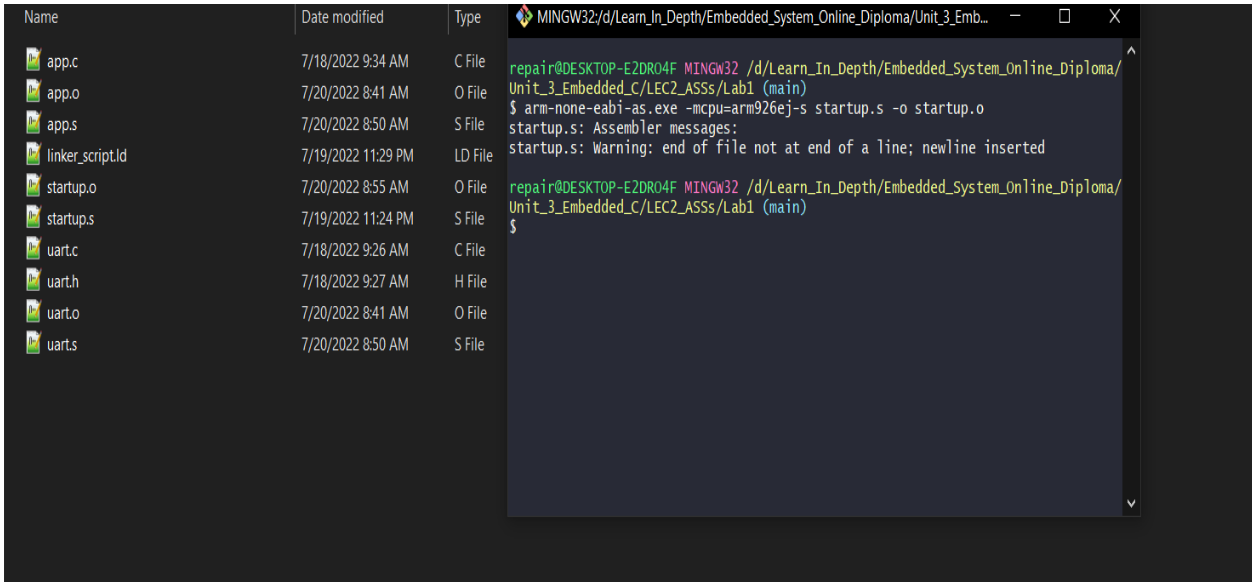


Name	Date modified	Type
app.c	7/18/2022 9:34 AM	C File
app.o	7/20/2022 8:41 AM	O File
app.s	7/20/2022 8:50 AM	S File
linker_script.ld	7/19/2022 11:29 PM	LD Fil
startup.s	7/19/2022 11:24 PM	S File
uart.c	7/18/2022 9:26 AM	C File
uart.h	7/18/2022 9:27 AM	H File
uart.o	7/20/2022 8:41 AM	O File
uart.s	7/20/2022 8:50 AM	S File

```
  1 .data         00000000  00000000  00000000  00000084  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000084  2**0
                  ALLOC
  3 .comment      00000012  00000000  00000000  00000084  2**0
                  CONTENTS, READONLY
  4 .ARM.attributes 00000032  00000000  00000000  00000096  2**0
                  CONTENTS, READONLY

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objdump.exe -D app.o >> app.s

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objdump.exe -D uart.o >> uart.s

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ |
```

The Assembly files are app.d & uart.s

$ Compile startup.s → startup.o



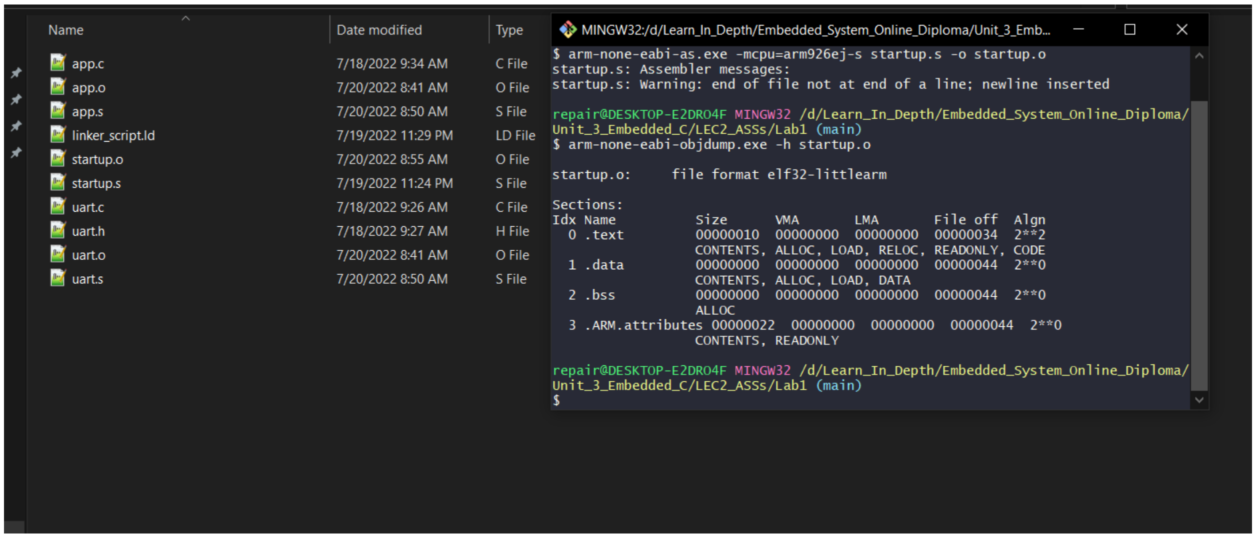
Name	Date modified	Type
app.c	7/18/2022 9:34 AM	C File
app.o	7/20/2022 8:41 AM	O File
app.s	7/20/2022 8:50 AM	S File
linker_script.ld	7/19/2022 11:29 PM	LD File
startup.o	7/20/2022 8:55 AM	O File
startup.s	7/19/2022 11:24 PM	S File
uart.c	7/18/2022 9:26 AM	C File
uart.h	7/18/2022 9:27 AM	H File
uart.o	7/20/2022 8:41 AM	O File
uart.s	7/20/2022 8:50 AM	S File

MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

```
repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-as.exe -mcpu=arm926ej-s startup.s -o startup.o
startup.s: Assembler messages:
startup.s: Warning: end of file not at end of a line; newline inserted

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$
```

$ Show → startup.o memory segments

We will use object utility → objdum → -h

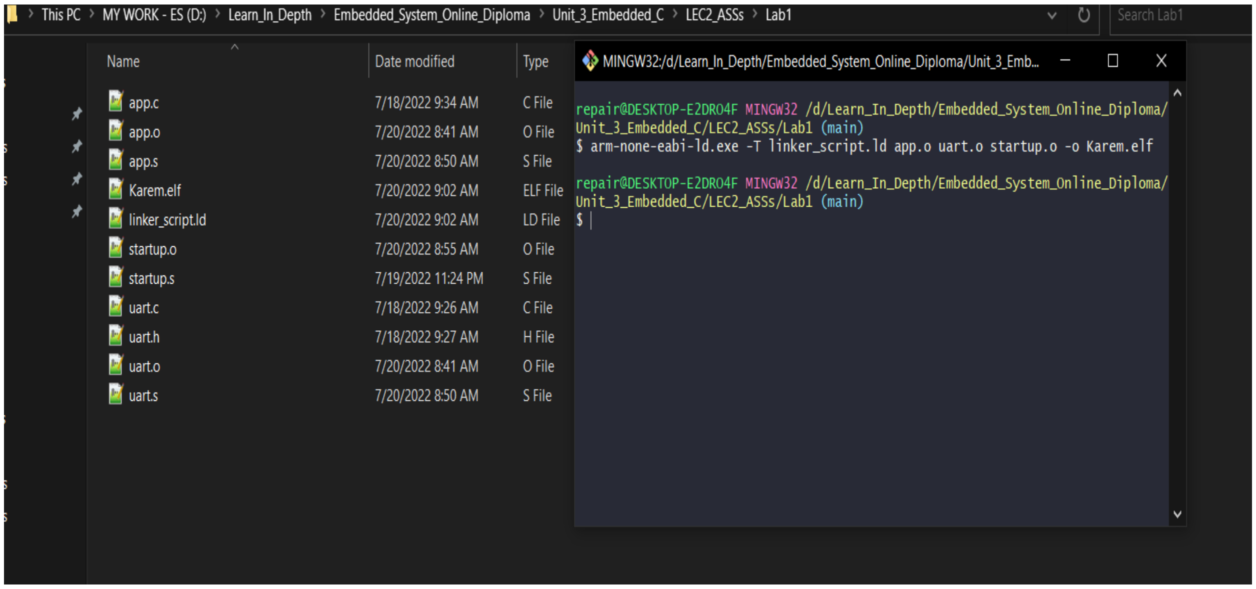

MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

Name	Date modified	Type
app.c	7/18/2022 9:34 AM	C File
app.o	7/20/2022 8:41 AM	O File
app.s	7/20/2022 8:50 AM	S File
linker_script.ld	7/19/2022 11:29 PM	LD File
startup.o	7/20/2022 8:55 AM	O File
startup.s	7/19/2022 11:24 PM	S File
uart.c	7/18/2022 9:26 AM	C File
uart.h	7/18/2022 9:27 AM	H File
uart.o	7/20/2022 8:41 AM	O File
uart.s	7/20/2022 8:50 AM	S File

```
$ arm-none-eabi-as.exe -mcpu=arm926ej-s startup.s -o startup.o
startup.s: Assembler messages:
startup.s: Warning: end of file not at end of a line; newline inserted

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objdump.exe -h startup.o

startup.o:     file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .text         00000010  00000000  00000000  00000034  2**2
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  00000000  00000000  00000044  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000000  00000000  00000000  00000044  2**0
                  ALLOC
  3 .ARM.attributes 00000022  00000000  00000000  00000044  2**0
                  CONTENTS, READONLY

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$
```

Note :- Only .text segment has a size

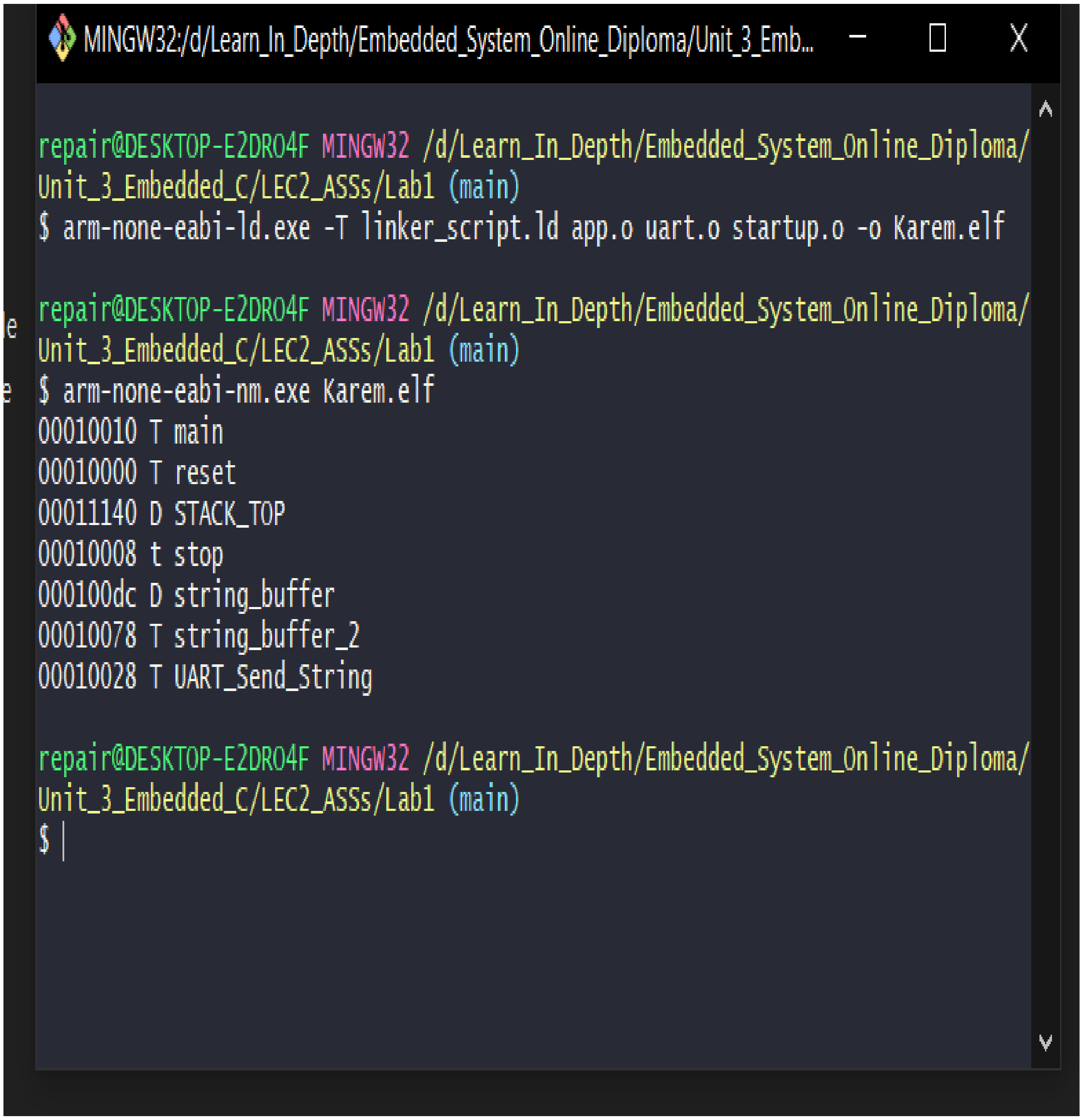
$ Linking with linker script


This PC › MY WORK - ES (D:) › Learn_In_Depth › Embedded_System_Online_Diploma › Unit_3_Embedded_C › LEC2_ASSs › Lab1
Search Lab1
MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

Name	Date modified	Type
app.c	7/18/2022 9:34 AM	C File
app.o	7/20/2022 8:41 AM	O File
app.s	7/20/2022 8:50 AM	S File
Karem.elf	7/20/2022 9:02 AM	ELF File
linker_script.ld	7/20/2022 9:02 AM	LD File
startup.o	7/20/2022 8:55 AM	O File
startup.s	7/19/2022 11:24 PM	S File
uart.c	7/18/2022 9:26 AM	C File
uart.h	7/18/2022 9:27 AM	H File
uart.o	7/20/2022 8:41 AM	O File
uart.s	7/20/2022 8:50 AM	S File

```
repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-ld.exe -T linker_script.ld app.o uart.o startup.o -o Karem.elf

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$
```

$ Show the Symbols

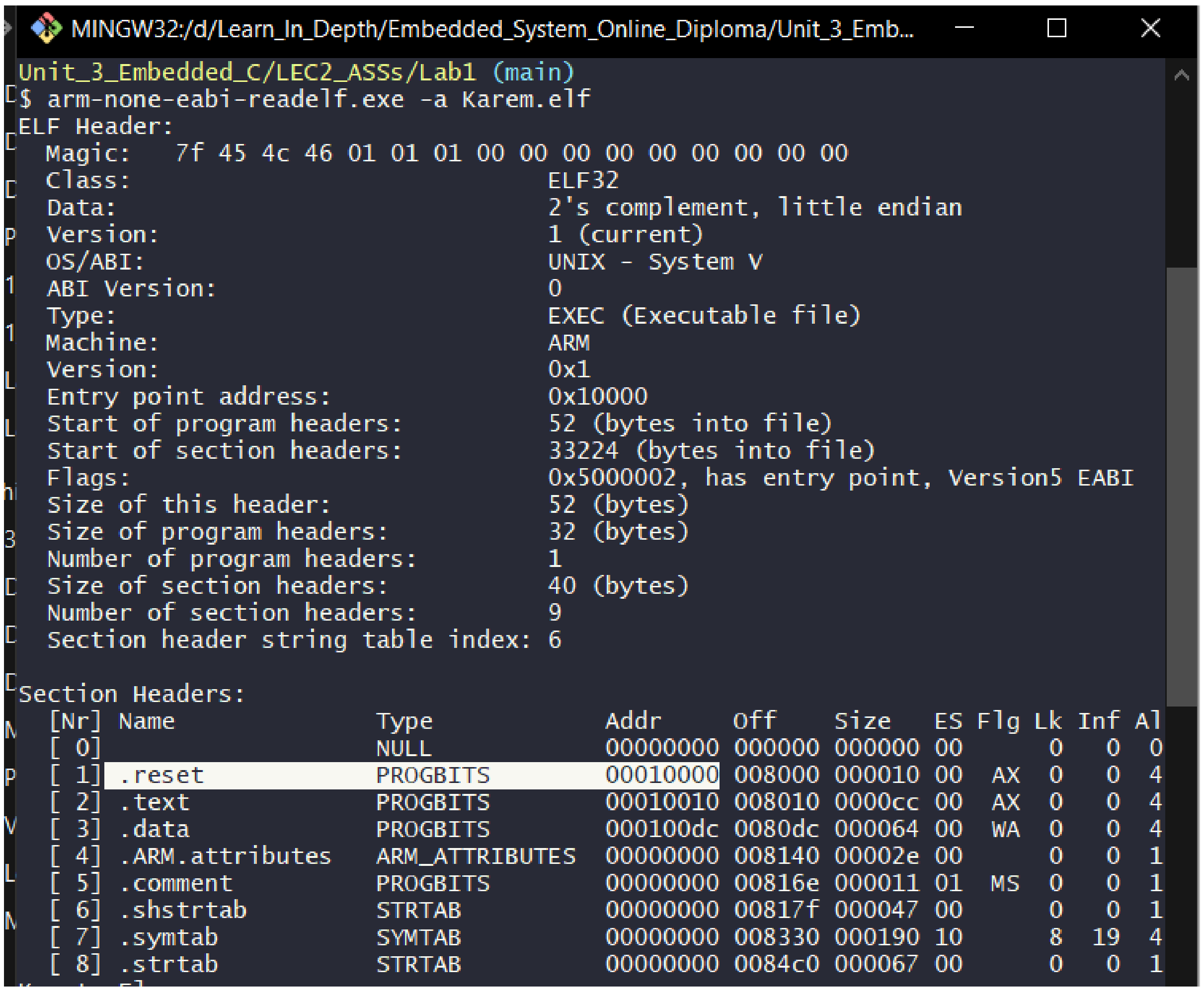

MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

```
repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-ld.exe -T linker_script.ld app.o uart.o startup.o -o Karem.elf

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-nm.exe Karem.elf
00010010 T main
00010000 T reset
00011140 D STACK_TOP
00010008 t stop
000100dc D string_buffer
00010078 T string_buffer_2
00010028 T UART_Send_String

repair@DESKTOP-E2DR04F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$
```

Note :-

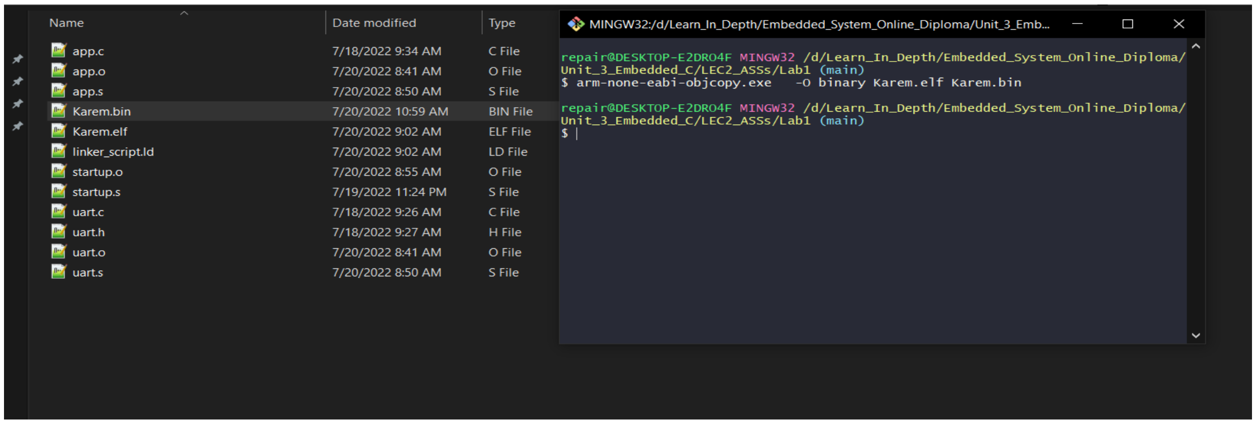
Now each symbol has its address (Physical Address) & its memory segment.

$ Check the Address of the reset section @ Startup code.



```
MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-readelf.exe -a Karem.elf
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           ARM
  Version:                           0x1
  Entry point address:               0x10000
  Start of program headers:          52 (bytes into file)
  Start of section headers:          33224 (bytes into file)
  Flags:                             0x5000002, has entry point, Version5 EABI
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         1
  Size of section headers:           40 (bytes)
  Number of section headers:         9
  Section header string table index: 6

Section Headers:
  [Nr] Name              Type            Addr     Off    Size   ES Flg Lk Inf Al
  [ 0]                   NULL            00000000 000000 000000 00      0   0  0
  [ 1] .reset            PROGBITS        00010000 008000 000010 00  AX  0   0  4
  [ 2] .text             PROGBITS        00010010 008010 0000cc 00  AX  0   0  4
  [ 3] .data             PROGBITS        000100dc 0080dc 000064 00  WA  0   0  4
  [ 4] .ARM.attributes   ARM_ATTRIBUTES  00000000 008140 00002e 00      0   0  1
  [ 5] .comment          PROGBITS        00000000 00816e 000011 01  MS  0   0  1
  [ 6] .shstrtab         STRTAB          00000000 00817f 000047 00      0   0  1
  [ 7] .symtab           SYMTAB          00000000 008330 000190 10      8  19  4
  [ 8] .strtab           STRTAB          00000000 0084c0 000067 00      0   0  1
```

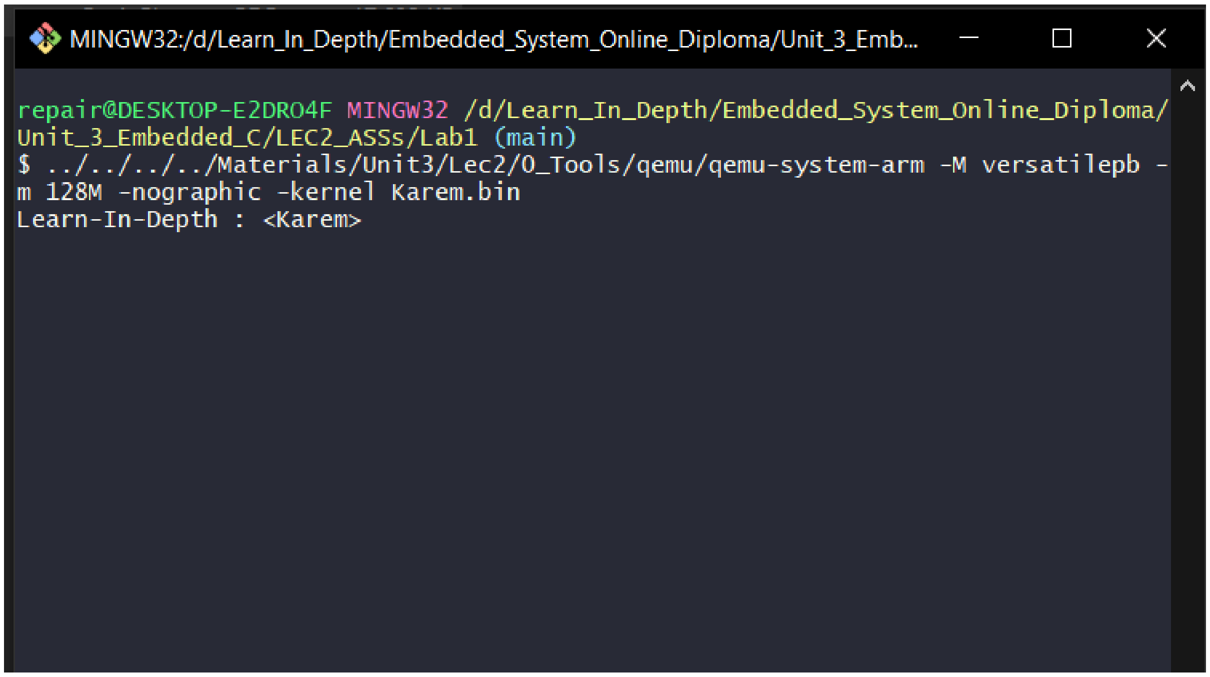
$ Extract .bin from .elf (delete debugging info) :



Name	Date modified	Type
app.c	7/18/2022 9:34 AM	C File
app.o	7/20/2022 8:41 AM	O File
app.s	7/20/2022 8:50 AM	S File
Karem.bin	7/20/2022 10:59 AM	BIN File
Karem.elf	7/20/2022 9:02 AM	ELF File
linker_script.ld	7/20/2022 9:02 AM	LD File
startup.o	7/20/2022 8:55 AM	O File
startup.s	7/19/2022 11:24 PM	S File
uart.c	7/18/2022 9:26 AM	C File
uart.h	7/18/2022 9:27 AM	H File
uart.o	7/20/2022 8:41 AM	O File
uart.s	7/20/2022 8:50 AM	S File

```
MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ arm-none-eabi-objcopy.exe   -O binary Karem.elf Karem.bin

repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ |
```

$ Run The Code Using qemu :-



```
MINGW32:/d/Learn_In_Depth/Embedded_System_Online_Diploma/Unit_3_Emb...

repair@DESKTOP-E2DRO4F MINGW32 /d/Learn_In_Depth/Embedded_System_Online_Diploma/
Unit_3_Embedded_C/LEC2_ASSs/Lab1 (main)
$ ../../../../Materials/Unit3/Lec2/0_Tools/qemu/qemu-system-arm -M versatilepb -
m 128M -nographic -kernel Karem.bin
Learn-In-Depth : <Karem>
```