

# Unit3\_Lesson2\_Lab\_1

## 1. Code Files

### 1.1: uart.c

```
D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1\u...
File Edit Selection Find View Goto Tools Project Preferences Help

uart.c x app.c x uarth x startup.s x linker_script.ld x + v
1  /**
2  *****
3  * @file      : uart.c
4  * @author    : Mina Gamil
5  * @date      : 27 Dec 2024
6  * @brief     : Uart Program To Send String
7  *****
8  */
9
10
11 /** INCLUDE USED HEADER FILES **/
12 #include "uart.h"
13
14 /** Define UART0DR LOCATION ADDRESS ***/
15 #define UART0DR *((vuint32*)((uint32*)0x101f1000))
16
17 /** void vUart0_Send_String to send data to UART **/
18 void vUart0_Send_String(uint8* P_TX_String)
19 {
20     /* Loop while NULL */
21     while(*P_TX_String != '\0')
22     {
23         UART0DR = (uint8) *P_TX_String;    /* Send Byte for Uart */
24         P_TX_String++;                      /* Increment Address */
25     }
26 }
```

### 1.2: uart.h

```
D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1\u...
File Edit Selection Find View Goto Tools Project Preferences Help

uart.c x app.c x uarth x startup.s x linker_script.ld x + v
1  #ifndef UART_H
2  #define UART_H
3
4  /** INCLUDE USED HEADER FILES **/
5  #include "Platform_Types.h"
6
7  /** void vUart0_Send_String to send data to UART **/
8  void vUart0_Send_String(uint8* P_TX_String);
9
10 #endif
```

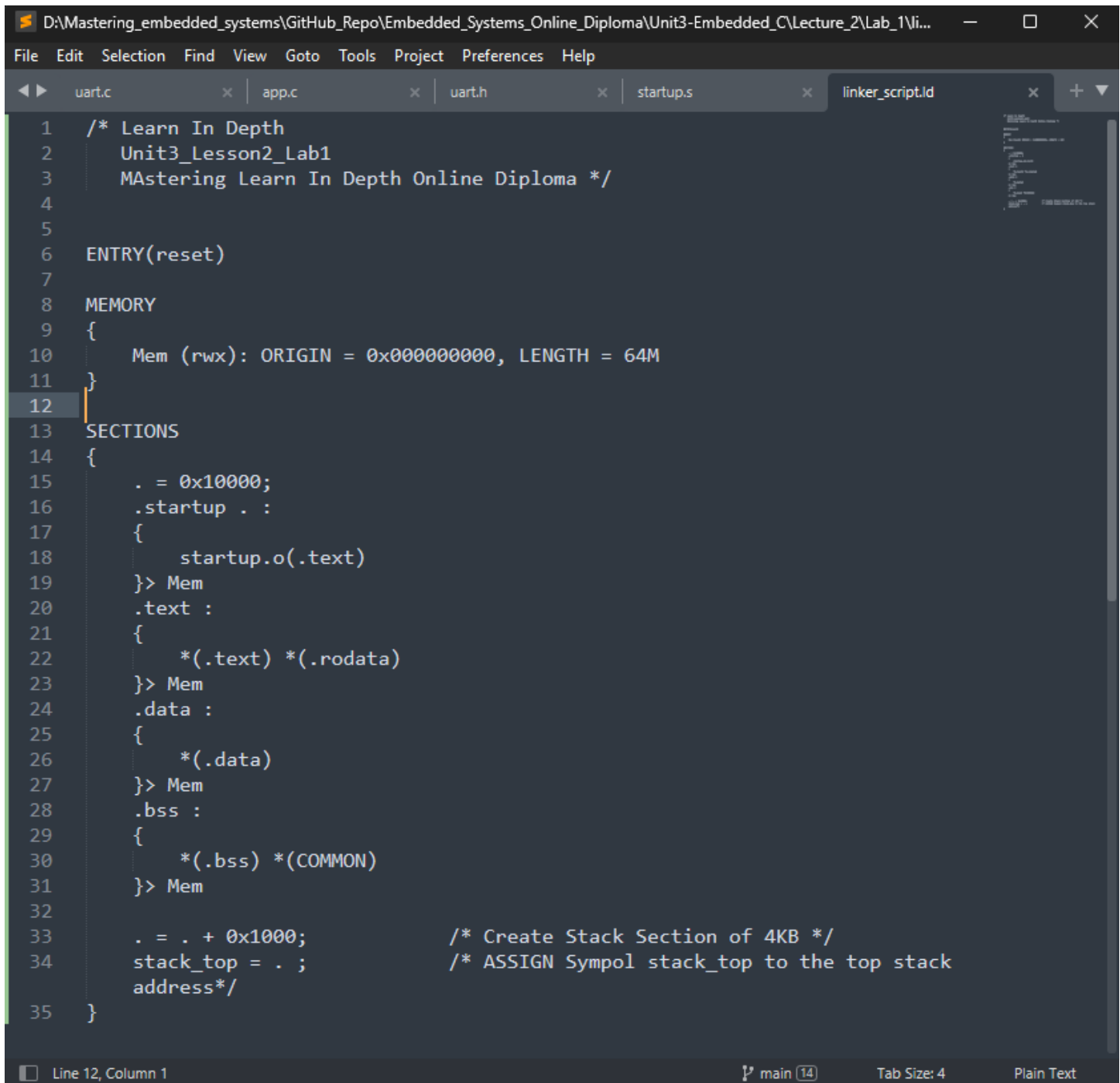
### 1.3: app.c

```
D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1\app.c
File Edit Selection Find View Goto Tools Project Preferences Help
uart.c x app.c x uart.h x startup.s x linker_script.ld x + v
1  /**
2  ****
3  * @file      : app.c
4  * @author    : Mina Gamil
5  * @date      : 27 Dec 2024
6  * @brief     : First Embedded lab Send data To UART
7  ****
8  */
9
10
11 /** INCLUDE USED HEADER FILES **/
12 #include "uart.h"
13
14 /*** Declare And Initialize String To Send ***/
15 uint8 string_buffer[100] = "Learn-In-Depth : Mina";
16
17 /** Main Body Program **/
18 void main(void)
19 {
20     /** Call vUart0_Send_String to send data to UART **/
21     vUart0_Send_String(string_buffer);
22 }
```

### 1.4: startup.s

```
D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1\startup.s
File Edit Selection Find View Goto Tools Project Preferences Help
uart.c x app.c x uart.h x startup.s x linker_script.ld x + v
1  .global reset                                @@ make reset label Global to be seen from
2  all files
3
4  reset :
5      ldr sp, = stack_top                      @@ Init. Stack Pointer to stack top
6      bl main                                  @@ Branch label to main label
7      b stop                                    @@ Branch stop if we return from main
8
9  stop :
10     b stop                                    @@ Branch to stop and enter infinite loop
```

## 1.5: linker\_script.ls



```
1 /* Learn In Depth
2 Unit3_Lesson2_Lab1
3 MAsTering Learn In Depth Online Diploma */
4
5
6 ENTRY(reset)
7
8 MEMORY
9 {
10 Mem (rwx): ORIGIN = 0x00000000, LENGTH = 64M
11 }
12
13 SECTIONS
14 {
15 . = 0x10000;
16 .startup . :
17 {
18 startup.o(.text)
19 }> Mem
20 .text :
21 {
22 *(.text) *(.rodata)
23 }> Mem
24 .data :
25 {
26 *(.data)
27 }> Mem
28 .bss :
29 {
30 *(.bss) *(COMMON)
31 }> Mem
32
33 . = . + 0x1000; /* Create Stack Section of 4KB */
34 stack_top = . ; /* ASSIGN Sympol stack_top to the top stack
35 address*/
36 }
```

Line 12, Column 1

main 14

Tab Size: 4

Plain Text

## 2. Object Files and Sections analyze with debug sections

### 2.1: uart.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-gcc -c -g -I . -mcpu=arm926ej-s uart.c -o uart.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -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 .debug_info     000000c1  00000000  00000000  00000084  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev   00000070  00000000  00000000  00000145  2**0
    CONTENTS, READONLY, DEBUGGING
  5 .debug_loc      0000002c  00000000  00000000  000001b5  2**0
    CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges  00000020  00000000  00000000  000001e1  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line     00000053  00000000  00000000  00000201  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str      00000128  00000000  00000000  00000254  2**0
    CONTENTS, READONLY, DEBUGGING
  9 .comment        00000012  00000000  00000000  0000037c  2**0
    CONTENTS, READONLY
10 .ARM.attributes 00000032  00000000  00000000  0000038e  2**0
    CONTENTS, READONLY
11 .debug_frame     00000028  00000000  00000000  000003c0  2**2
    CONTENTS, RELOC, READONLY, DEBUGGING
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

## 2.2: app.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embe
dded_C\Lecture_2\Lab_1> arm-none-eabi-gcc -c -g -I . -mcpu=arm926ej-s app.c -o app.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embe
dded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -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 .debug_info     000000b6  00000000  00000000  000000b0  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  4 .debug_abbrev   00000072  00000000  00000000  00000166  2**0
    CONTENTS, READONLY, DEBUGGING
  5 .debug_loc      0000002c  00000000  00000000  000001d8  2**0
    CONTENTS, READONLY, DEBUGGING
  6 .debug_aranges  00000020  00000000  00000000  00000204  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  7 .debug_line     0000004b  00000000  00000000  00000224  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str       00000115  00000000  00000000  0000026f  2**0
    CONTENTS, READONLY, DEBUGGING
  9 .comment        00000012  00000000  00000000  00000384  2**0
    CONTENTS, READONLY
 10 .ARM.attributes 00000032  00000000  00000000  00000396  2**0
    CONTENTS, READONLY
 11 .debug_frame     0000002c  00000000  00000000  000003c8  2**2
    CONTENTS, RELOC, READONLY, DEBUGGING
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embe
dded_C\Lecture_2\Lab_1>
```

## 2.3: startup.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-as -g -mcpu=arm926ej-s startup.s -o startup.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -h startup.o

startup.o:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          00000014  00000000  00000000  00000034  2**2
    CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data          00000000  00000000  00000000  00000048  2**0
    CONTENTS, ALLOC, LOAD, DATA
  2 .bss           00000000  00000000  00000000  00000048  2**0
    ALLOC
  3 .ARM.attributes 00000022  00000000  00000000  00000048  2**0
    CONTENTS, READONLY
  4 .debug_line     0000003b  00000000  00000000  0000006a  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  5 .debug_info     0000009d  00000000  00000000  000000a5  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  6 .debug_abbrev   00000014  00000000  00000000  00000142  2**0
    CONTENTS, READONLY, DEBUGGING
  7 .debug_aranges  00000020  00000000  00000000  00000158  2**3
    CONTENTS, RELOC, READONLY, DEBUGGING
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

## 2.4: lab\_1.elf Sections

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-ld -T linker_script.ld startup.o app.o uart.o -o lab_1.elf
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -h lab_1.elf

lab_1.elf:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .startup        00000010  00010000  00010000  00008000  2**2
    CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .text           00000068  00010010  00010010  00008010  2**2
    CONTENTS, ALLOC, LOAD, READONLY, CODE
  2 .data           00000064  00010078  00010078  00008078  2**2
    CONTENTS, ALLOC, LOAD, DATA
  3 .ARM.attributes 0000002e  00000000  00000000  000080dc  2**0
    CONTENTS, READONLY
  4 .comment         00000011  00000000  00000000  0000810a  2**0
    CONTENTS, READONLY
  5 .debug_line       000000d8  00000000  00000000  0000811b  2**0
    CONTENTS, READONLY, DEBUGGING
  6 .debug_info       00000214  00000000  00000000  000081f3  2**0
    CONTENTS, READONLY, DEBUGGING
  7 .debug_abbrev     000000f6  00000000  00000000  00008407  2**0
    CONTENTS, READONLY, DEBUGGING
  8 .debug_aranges    00000060  00000000  00000000  00008500  2**3
    CONTENTS, READONLY, DEBUGGING
  9 .debug_loc        00000058  00000000  00000000  00008560  2**0
    CONTENTS, READONLY, DEBUGGING
 10 .debug_str        0000012a  00000000  00000000  000085b8  2**0
    CONTENTS, READONLY, DEBUGGING
 11 .debug_frame      00000054  00000000  00000000  000086e4  2**2
    CONTENTS, READONLY, DEBUGGING
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```



### 3. Object Files and Sections analyze without debug sections

#### 3.1: uart.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-gcc -c -I . -mcpu=arm926ej-s uart.c -o uart.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -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					

```
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

#### 3.2: app.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-gcc -c -I . -mcpu=arm926ej-s app.c -o app.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -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	.comment	00000012	00000000	00000000	000000b0	2**0
	CONTENTS, READONLY					
4	.ARM.attributes	00000032	00000000	00000000	000000c2	2**0
	CONTENTS, READONLY					

```
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```



### 3.3: startup.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-as -mcpu=arm926ej-s startup.s -o startup.o
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -h startup.o

startup.o:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          00000014  00000000  00000000  00000034  2**2
    CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data          00000000  00000000  00000000  00000048  2**0
    CONTENTS, ALLOC, LOAD, DATA
  2 .bss           00000000  00000000  00000000  00000048  2**0
    ALLOC
  3 .ARM.attributes 00000022  00000000  00000000  00000048  2**0
    CONTENTS, READONLY
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

### 3.4: lab\_1.elf Sections

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-ld -T linker_script.ld startup.o app.o uart.o -o lab_1.elf
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-objdump -h lab_1.elf

lab_1.elf:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .startup       00000014  00010000  00010000  00008000  2**2
    CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .text          00000068  00010014  00010014  00008014  2**2
    CONTENTS, ALLOC, LOAD, READONLY, CODE
  2 .data          00000064  0001007c  0001007c  0000807c  2**2
    CONTENTS, ALLOC, LOAD, DATA
  3 .ARM.attributes 0000002e  00000000  00000000  000080e0  2**0
    CONTENTS, READONLY
  4 .comment       00000011  00000000  00000000  0000810e  2**0
    CONTENTS, READONLY
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

## 4. Symbols table

### 4.1: uart.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-nm uart.o
00000000 T vUart0_Send_String
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

### 4.2: app.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-nm app.o
00000000 T main
00000000 D string_buffer
          U vUart0_Send_String
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

### 4.3: startup.o

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-nm startup.o
          U main
00000000 T reset
          U stack_top
0000000c t stop
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

### 4.4: lab\_1.elf

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-nm lab_1.elf
00010014 T main
00010000 T reset
000110e0 D stack_top
0001000c t stop
0001007c D string_buffer
0001002c T vUart0_Send_String
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

## 5. readelf Binary Utilities

```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-readelf -a lab_1.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: 33128 (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   00  0   0   0
[ 1] .startup              PROGBITS       00010000       000000       000014 00   AX  0   0   4
[ 2] .text                PROGBITS       00010014       000014       000068 00   AX  0   0   4
[ 3] .data                PROGBITS       0001007c       00007c       000064 00   WA  0   0   4
[ 4] .ARM.attributes      ARM_ATTRIBUTES 00000000       0000e0       00002e 00   00  0   0   1
[ 5] .comment             PROGBITS       00000000       00010e       000011 01   MS  0   0   1
[ 6] .shstrtab            STRTAB         00000000       00011f       000049 00   00  0   0   1
[ 7] .symtab              SYMTAB         00000000       0002d0       000170 10   00 18  4
[ 8] .strtab              STRTAB         00000000       000440       000059 00   00  0   0   1

Key to Flags:
W (write), A (alloc), X (execute), M (merge), S (strings)
I (info), L (link order), G (group), T (TLS), E (exclude), x (unknown)
O (extra OS processing required) o (OS specific), p (processor specific)

There are no section groups in this file.

Program Headers:
Type           Offset           VirtAddr       PhysAddr       FileSiz MemSiz   Flg Align
LOAD           0x00000000      0x00010000     0x00010000     0x000e0 0x000e0 RWE 0x0000

Section to Segment mapping:
Segment Sections...
00          .startup .text .data

There is no dynamic section in this file.

There are no relocations in this file.

There are no unwind sections in this file.

Symbol table '.symtab' contains 23 entries:
Num:  Value           Size  Type   Bind  Vis    Ndx  Name
 0:  00000000           0 NOTYPE LOCAL DEFAULT UND
 1:  00010000           0 SECTION LOCAL DEFAULT 1
 2:  00010014           0 SECTION LOCAL DEFAULT 2
 3:  0001007c           0 SECTION LOCAL DEFAULT 3
 4:  00000000           0 SECTION LOCAL DEFAULT 4
 5:  00000000           0 SECTION LOCAL DEFAULT 5
 6:  00000000           0 FILE  LOCAL DEFAULT ABS startup.o
 7:  00010000           0 NOTYPE LOCAL DEFAULT 1 $a
 8:  0001000c           0 NOTYPE LOCAL DEFAULT 1 stop
 9:  00010010           0 NOTYPE LOCAL DEFAULT 1 $d
10:  00000000           0 FILE  LOCAL DEFAULT ABS app.c
11:  0001007c           0 NOTYPE LOCAL DEFAULT 3 $d
12:  00010014           0 NOTYPE LOCAL DEFAULT 2 $a
13:  00010028           0 NOTYPE LOCAL DEFAULT 2 $d
14:  00000000           0 FILE  LOCAL DEFAULT ABS uart.c
15:  0001002c           0 NOTYPE LOCAL DEFAULT 2 $a
16:  00010078           0 NOTYPE LOCAL DEFAULT 2 $d
17:  00000000           0 FILE  LOCAL DEFAULT ABS
18:  00010000           0 NOTYPE GLOBAL DEFAULT 1 reset
19:  000110e0           0 NOTYPE GLOBAL DEFAULT 3 stack_top
20:  00010014          24 FUNC  GLOBAL DEFAULT 2 main
21:  0001007c         100 OBJECT GLOBAL DEFAULT 3 string_buffer
22:  0001002c          80 FUNC  GLOBAL DEFAULT 2 vUart0_Send_String

No version information found in this file.
Attribute Section: aeabi
File Attributes
Tag_CPU_name: "ARM926EJ-S"
Tag_CPU_arch: v5TEJ
Tag_ARM_ISA_use: Yes
Tag_THUMB_ISA_use: Thumb-1
Tag_ABI_PCS_wchar_t: 4
Tag_ABI_FP_denormal: Needed
Tag_ABI_FP_exceptions: Needed
Tag_ABI_FP_number_model: IEEE 754
Tag_ABI_align_needed: 8-byte
Tag_ABI_enum_size: small
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

## 6. Map file

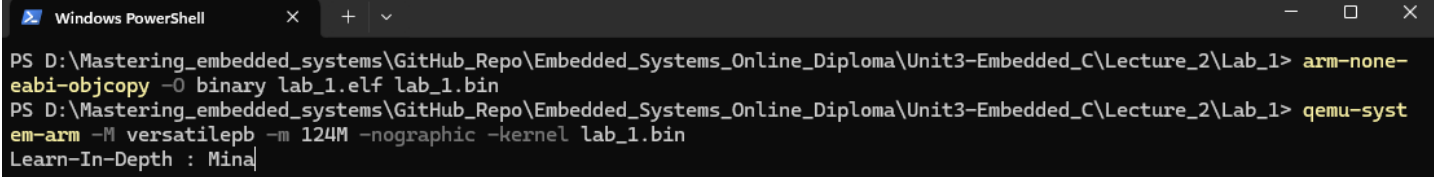
```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-eabi-ld -T linker_script.ld startup.o app.o uart.o -o lab_1.elf -M >lab_1_Map.map
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> |
```

```
D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\...
File Edit Selection Find View Goto Tools Project Preferences Help
uart.c x app.c x uart.h x startup.s x linker_script.ld x lab_1_Map.map x + v

1
2 Memory Configuration
3
4 Name          Origin          Length          Attributes
5 Mem           0x00000000      0x04000000      xrw
6 *default*     0x00000000      0xffffffff
7
8 Linker script and memory map
9
10              0x00010000      . = 0x1000
11
12 .startup      0x00010000      0x14
13 startup.o(.text)
14 .text        0x00010000      0x14 startup.o
15              0x00010000      reset
16
17 .text        0x00010014      0x68
18 *(.text)
19 .text        0x00010014      0x18 app.o
20              0x00010014      main
21 .text        0x0001002c      0x50 uart.o
22              0x0001002c      vUart0_Send_String
23 *(.rodata)
24
25 .glue_7      0x0001007c      0x0
26 .glue_7      0x00000000      0x0 linker stubs
27
28 .glue_7t     0x0001007c      0x0
29 .glue_7t     0x00000000      0x0 linker stubs
30
31 .vfp11_veneer 0x0001007c      0x0
32 .vfp11_veneer 0x00000000      0x0 linker stubs
33
34 .v4_bx       0x0001007c      0x0
35 .v4_bx       0x00000000      0x0 linker stubs
36
37 .iplt        0x0001007c      0x0
38 .iplt        0x00000000      0x0 startup.o
39
40 .rel.dyn     0x0001007c      0x0
41 .rel.iplt    0x00000000      0x0 startup.o
42
43 .data        0x0001007c      0x64
44 *(.data)
45 .data        0x0001007c      0x0 startup.o
46 .data        0x0001007c      0x64 app.o
47              0x0001007c      string_buffer
48 .data        0x000100e0      0x0 uart.o
49
50 .igot.plt    0x000100e0      0x0
51 .igot.plt    0x00000000      0x0 startup.o
52
53 .bss         0x000100e0      0x0
54 *(.bss)
55 .bss         0x000100e0      0x0 startup.o
56 .bss         0x000100e0      0x0 app.o
57 .bss         0x000100e0      0x0 uart.o
58 *(COMMON)
59              0x000110e0      . = (. + 0x1000)
60              0x000110e0      stack_top = .
61 LOAD startup.o
62 LOAD app.o
63 LOAD uart.o
64 OUTPUT(lab_1.elf elf32-littlearm)
65
66 .ARM.attributes
67              0x00000000      0x2e
68 .ARM.attributes
69              0x00000000      0x22 startup.o
70 .ARM.attributes
71              0x00000022      0x32 app.o

Line 39, Column 1      main (15)      Spaces: 4      Plain Text
```

## 7. Extract binary code and run program



```
Windows PowerShell
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> arm-none-
eabi-objcopy -O binary lab_1.elf lab_1.bin
PS D:\Mastering_embedded_systems\GitHub_Repo\Embedded_Systems_Online_Diploma\Unit3-Embedded_C\Lecture_2\Lab_1> qemu-syst
em-arm -M versatilepb -m 124M -nographic -kernel lab_1.bin
Learn-In-Depth : Mina
```