<u>Report about lab1</u>

1-Codes:

App.c

Uart.c

```
Excourses/embedded system diplom/content/Term 1/unit 3 (Embedded C)\lec2\Lap1\uartc - Sublime Text (ADMIN / UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

#include "uart.h"

#include "uart.h"

#define UARTODR *((volatile unsigned int* const)((unsigned int*)0x101f1000))

while (*p_tx_string(unsigned char* p_tx_string){

While (*p_tx_string != '\0')

{

UARTODR = (unsigned int)(*p_tx_string);

p_tx_string++;

}

}
```

Uart.h

```
Excourses\embedded system diplom\content\Term 1\unit 3 (Embedded C)\lec2\Lap1\uart.h - Sublime Text (ADMIN / UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

1 #ifndef _UART_H_
2 #define _UART_H_
3

4 void uart_send_string(unsigned char* p_tx_string);
5
6 #endif
```

Startup.s

Linker_script.ld

```
E:\courses\embedded system diplom\content\Term 1\unit 3 (Embedded C)\lec2\Lap1\linker script.ld - Sublime T
File Edit Selection Find View Goto Tools Project Preferences Help
      linker_script.ld
      ENTRY(reset)
   1
   2
      MEMORY
             Mem (rwx): ORIGIN = 0x00000000, LENGTH = 64M
       SECTIONS
             . = 0 \times 10000;
  10
             .startup . :
  11
                  startup.o(.text)
  12
  13
             }>Mem
  14
             .text:
  15
                  *(.text) *(.rodata)
 16
            }>Mem
  17
             .data:
 18
  19
                  *(.data)
  20
  21
             }>Mem
  22
             .bss:
 23
                  *(.bss) *(.common)
 25
             }>Mem
  26
 27
             . = . + 0 \times 1000;
  28
             stack_top = .;
  29
       }
```

2-get obj files from uart.c and app.c

App.o

arm-none-eabi-gcc.exe -c -mcpu=arm926ej-s -I app.c -o

uart.o

arm-none-eabi-gcc.exe -c -mcpu=arm926ej-s -I uart.c -o uart.o

startup.o

arm-none-eabi-as.exe -mcpu=arm926ej-s startup.s -ostartup.o

3-sectons for obj files

App.o

```
arm-none-eabi-objdump.exe -h app.o
            file format elf32-littlearm
app.o:
Sections:
Idx Name
0 .text
                    size
                               VMA
                                          LMA
                                                      File off
                                                                 Algn
                               00000000
                                                      00000034
                                                                 2**2
                    00000018
                                          00000000
                                                      READONLY,
                    CONTENTS,
                                                                 CODE
                               ALLOC, LOAD, RELOC,
  1 .data
                    00000064
                               00000000 00000000
                                                      0000004c
                    CONTENTS, ALLOC, LOAD, DATA 00000000 00000000 00000000
  2 .bss
                                                      000000b0
                                                                 2**0
                    ALLOC
                    00000012
                               00000000
                                          00000000
                                                      000000b0
                                                                 2**0
                    CONTENTS,
                              READONLY
  4 .ARM.attributes 00000032 00000000
                                             00000000
                                                        000000c2
                                                                   2**0
                    CONTENTS, READONLY
```

Uart.o

```
$ arm-none-eabi-objdump.exe -h startup.o
               file format elf32-littlearm
startup.o:
Sections:
Idx Name
                  Size
                            VMA
                                       LMA
                                                 File off
                  00000010
                            00000000
                                       00000000
                                                 00000034
                                                            2**2
  0 .text
                                                 READONLY,
                  CONTENTS, ALLOC, LOAD, RELOC,
                                                           CODE
                            00000000 00000000
 1 .data
                  00000000
                                                 00000044
                                                            2**0
                  CONTENTS, ALLOC, LOAD, DATA
                            00000000
                                                 00000044
                                                            2**0
  2 .bss
                  00000000
                                      00000000
                  ALLOC
  3 .ARM.attributes 00000022
                              00000000
                                         00000000
                                                   00000044
                                                              2**0
                  CONTENTS, READONLY
```

Startup.o

```
arm-none-eabi-objdump.exe -h startup.o
                 file format elf32-littlearm
startup.o:
Sections:
                                                                   Algn
Idx Name
                    Size
                                VMA
                                            LMA
                                                       File off
                    00000010 00000000 00000000 00000034
  0 .text
                    CONTENTS, ALLOC, LOAD, RELOC, READONLY, 0000000 00000000 00000044 CONTENTS, ALLOC, LOAD, DATA
                                                                   CODE
  1 .data
                                                                   2**0
                    00000000 00000000
  2 .bss
                                           00000000
                                                       00000044
                    ALLOC
  3 .ARM.attributes 00000022 00000000
                                              00000000
                                                          00000044 2**0
                    CONTENTS, READONLY
```

4-symbol tables for uart.o, app.o & startup.o

```
$ arm-none-eabi-nm.exe app.o
000000000 T main
000000000 D string_buffer
U uart_send_string

AMA@DESKTOP-DDNBIPK MINGW64 /e/coudiplom/content/Term 1/unit 3 (Embe $ arm-none-eabi-nm.exe uart.o
00000000 T uart_send_string

AMA@DESKTOP-DDNBIPK MINGW64 /e/coudiplom/content/Term 1/unit 3 (Embe $ arm-none-eabi-nm.exe startup.o
U main
000000000 T reset
U stack_top
000000008 t stop
```

5-get executable file (App.elf) and map file

arm-none-eabi-ld.exe -T Linker_Script.ld -Map=prog.map startup.o App.o Uart.o -o prog.elf

6-sections for prog.elf

```
$ arm-none-eabi-objdump.exe -h prog.elf
              file format elf32-littlearm
prog.elf:
Sections:
Idx Name
                  Size
                                                File off
                            VMA
                                      LMA
                                                           Algn
  0 .startup
                  00000010 00010000
                                      00010000
                                                000080000
                                                           2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .text
                                                           2**2
                  00000068 00010010 00010010
                                                00008010
                  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
                                      00000000
                                                0000810a
                  00000011
                            00000000
                  CONTENTS, READONLY
```

7-sembol table for prog.elf

```
$ arm-none-eabi-nm.exe prog.elf

00010010 T main

00010000 T reset

000110dc D stack_top

00010008 t stop

00010078 D string_buffer

00010028 T uart_send_string
```

8-get binary file

arm-none-eabi-objcopy.exe -O binary prog.elf prog.bin

9-burn binary file on board using gemu

\$ qemu-system-arm.exe -M versatilepb -m 128M -nographic -kernel prog.bin
learn-in-depth:<Ahmed>