Faculty of Engineering & Technology – Electrical & Computer Engineering Department

Second Semester 2020 – 2021

COMPUTER ORGANIZATION AND MICROPROCESSOR

ENCS2380

Assembly Project

Name: Ahmaide Al-Awawdah.

ID: 1190823

Section: 2

Date: 9th June 2021

The string that will be saved is = “hello world”

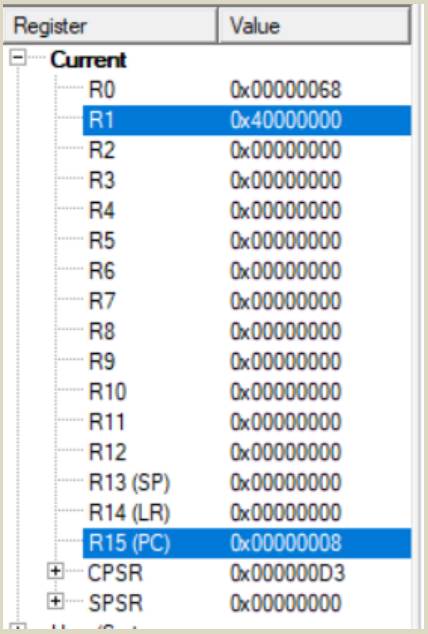
Encryption File:

```
    area TheData, DATA, readonly
addr1   DCB     "hello world", 0; the first location of the string
D   equ 0x40000000; the first location of the encrypted string
E   equ 0x40001008; the first location of the decrypted string
F   equ 0xFFFFFFFF
Z   equ 0x00000000


    area aEncryption, code, readonly
    ENTRY
    LDR     r0, =addr1; r0 = the address of the string
    LDR     r1,= D; r1 = the address of the encrypted string
looping     LDRB    r2,[r0]; starting a loop to store the encrypted string
    ROR     r2,r2,#2; Rotate the value in r2 to the right by 2
    add     r0,r0,#1; go to the address of the next character of the string
    STR     r2,[r1]; store the encrypted character in 32 bits, because the rotate goes on 32 bits
    MOVS    r3,r2; set the zero flag
    add     r1,r1,#4; point at the next memory address after 4 cells to save the next encrypted character in
    BNE     looping; loop end when Z flag = 0
    LDR     r4,= F; when r4=FFFFFFFF, that means that all the string characters are encrypted
    END
```

Decryption File

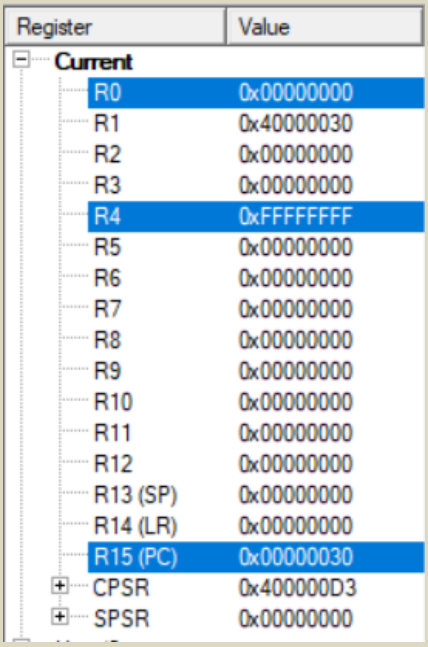
```
    area MyData, DATA, readonly
addr1   DCB     "hello world", 0
D   equ 0x40000000; the first location of the encrypted string
E   equ 0x40001008; the first location of the decrypted string
F   equ 0xFFFFFFFF
Z   equ 0x00000000

    area bDecryption, code, readonly
    ENTRY
    LDR     r1, = D; the address of the encrypted string
    LDR     r2, = E; the address of the decrypted string
    LDR     r4, = Z; set r4=0
looping2    LDR     r3,[r1]; starting a loop to store the decrypted string
    ROR     r3,r3,#30; Rotate the value in r2 to the right by 30 (which is the same as to the left by 2)
    MOVS    r4,r3; set the zero flag
    add     r1,r1,#4; increase r1 so it points on the location of the next encrypted character
    STRB    r3,[r2]; store the decrypted character in r3 in the memory location that is in r2
    add     r2,r2,#1; r2 will point at the next memory location to store the next decrypted character
    BNE     looping2; end of the loop when Z flag = 0
    LDR     r5,= F; when r4=FFFFFFFF, that means that all the string characters are decrypted
here    B   here; stay here
    END
```

The Registers before the encryption loop starts



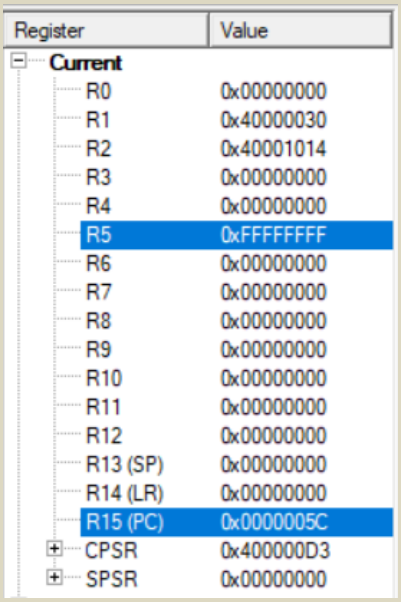
Register	Value
Current	
R0	0x00000068
R1	0x40000000
R2	0x00000000
R3	0x00000000
R4	0x00000000
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x00000000
R14 (LR)	0x00000000
R15 (PC)	0x00000008
CPSR	0x000000D3
SPSR	0x00000000

The Registers after the encryption ends



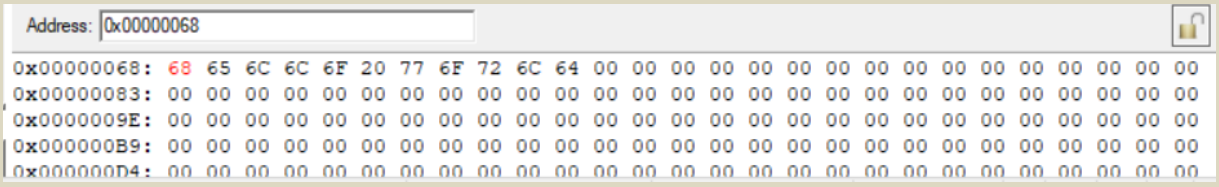
Register	Value
Current	
R0	0x00000000
R1	0x40000030
R2	0x00000000
R3	0x00000000
R4	0xFFFFFFFF
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x00000000
R14 (LR)	0x00000000
R15 (PC)	0x00000030
CPSR	0x400000D3
SPSR	0x00000000

The registers after the Decryption ends



Register	Value
Current	
R0	0x00000000
R1	0x40000030
R2	0x40001014
R3	0x00000000
R4	0x00000000
R5	0xFFFFFFFF
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x00000000
R14 (LR)	0x00000000
R15 (PC)	0x0000005C
CPSR	0x400000D3
SPSR	0x00000000

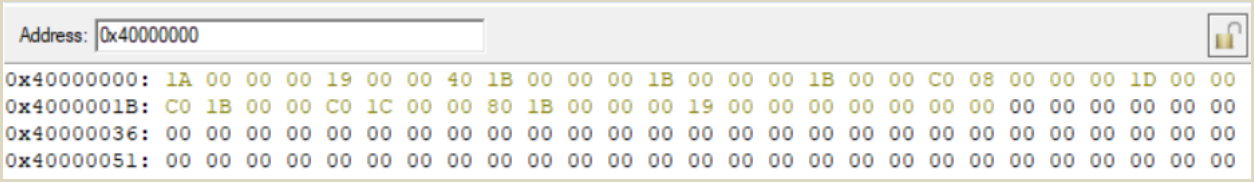
The string stored in the memory before encryption



Address: 0x00000068

```
0x00000068: 68 65 6C 6C 6F 20 77 6F 72 6C 64 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x00000083: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0000009E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x000000B9: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x000000D4: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

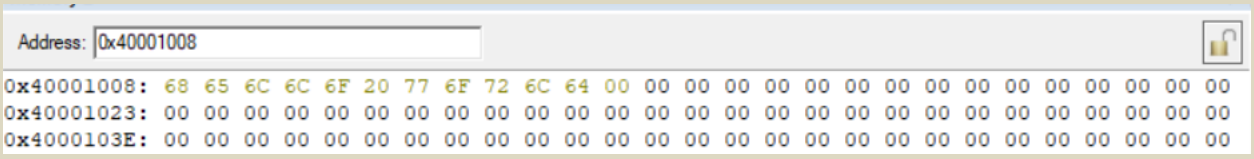
The Encrypted string stored in the memory



Address: 0x40000000

```
0x40000000: 1A 00 00 00 19 00 00 40 1B 00 00 00 1B 00 00 00 1B 00 00 C0 08 00 00 00 1D 00 00
0x4000001B: C0 1B 00 00 C0 1C 00 00 80 1B 00 00 00 19 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40000036: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40000051: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

The Decrypted string stored in the memory



Address: 0x40001008

```
0x40001008: 68 65 6C 6C 6F 20 77 6F 72 6C 64 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x40001023: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x4000103E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```