

12/14/2021



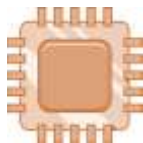
Homework 5

Lec 19-21



MICROPROCESSOR
AND
ASSEMBLY LANGUAGE

Fall 2021



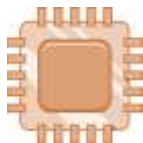
(۱) برنامه‌ای بنویسید که با استفاده از آن بتوان تشخیص داد که مقدار قرار گرفته در رجیستر R0 پالیندروم است یا خیر. (برای مثال، ۰۱۱۰ یک پالیندروم ۴ بیتی است).

Register	Value
Core	
R0	0x00000000
R1	0x00000141
R2	0x00000000
R3	0x00000000
R4	0x00000001
R5	0x00000000
R6	0x000000C6
R7	0x0000000A
R8	0x0000007B
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20000400
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x004001AE
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	64
Sec	0.00000533


```

1
2
3     EXPORT __main
4     AREA myCode , CODE , READONLY
5     ENTRY
6
7
8     __main
9     LDR r0 , =123
10    LDR r8 , =123
11    LDR r1 , =0
12    LDR r7 , =10
13    loop
14        UDIV r2 , r0 , r7
15        MUL r3 , r2 , r7
16        SUBS r4 , r0 , r3
17        MLA r1 , r1 , r7 , r4
18        UDIV r0 , r0 , r7
19        SUBS r5 , r0 , #0
20        BNE loop
21        SUBS r6 , r1 , r8
22    Hear B Hear
23        END
  
```

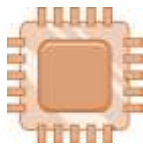
(۲) هنگامی که بر روی کیبورد، دو کاراکتر ۴ و ۶ را تایپ می‌کنیم، 0x34 و 0x36 در واقع به ما داده می‌شود. برنامه‌ای بنویسید که 0x34 و 0x36 را به packed BCD تبدیل کرده و نتیجه را در رجیستر R2 ذخیره نماید.



Core		0x004001A2 0000		MOVS r0, r0	
R0	0x00000004	0x004001A4 0000	MOVS	r0, r0	
R1	0x00000060	0x004001A6 0000	MOVS	r0, r0	
R2	0x00000064				
R3	0x00000000				
R4	0x00000000				
R5	0x00000000				
R6	0x00000000				
R7	0x00000000				
R8	0x00000000				
R9	0x00000000				
R10	0x00000000				
R11	0x00000000				
R12	0x00000000				
R13 (SP)	0x20000400				
R14 (LR)	0xFFFFFFFF				
R15 (PC)	0x0040019E				
xPSR	0x01000000				
Banked					
System					
Internal					
Mode	Thread				
Privilege	Privileged				
Stack	MSP				
States	12				
Sec	000000100				

startup_SAM3N.s	system_SAM3N.c	q2.s
1	export __main	
2	area start ,code, readonly	
3	entry	
4		
5	__main	
6	ldr r0, =0x34	
7	and r0, r0, #2_00001111	
8	ldr r1, =0x36	
9	and r1, r1, #2_00001111	
10	movs r1, r1, ror #28	
11	orr r2, r0, r1	
12	end	
13		
14		

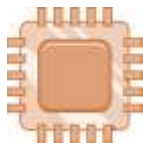
۳) الف) کد اسمبلی معادل قطعه کد زیر را بنویسید. (مقادیر متغیرهای استفاده شده در ثباتها طبق جدول زیر ذخیره شده است).



a	R4
b	R5
c	R6

```
while (a - b > 0) {  
    if (a > -b) {  
        c = c - a;  
        a = -a;  
    }  
    else {  
        b = c * b;  
        a = 2 - b;  
    }  
}
```

```
export __main  
area start ,code, readonly  
entry  
  
__main  
b while  
while  
    neg r7,r5  
    cmp r4,r5  
    bhi ifelse  
  
ifelse  
    cmp r4 ,r7  
    bhi iff  
    mul r5,r5,r6  
    rsb r5,r5,#2  
iff  
    sub r6,r6,r4  
    neg r4,r4  
    b while  
  
end
```



ب) اگر قصد داشته باشیم کد $R4++$ if $((R0==R1) \&\& (R2==R3))$ را به زبان اسمبلی بنویسیم، کد مناسب را فقط با سه دستور پیاده کنید.

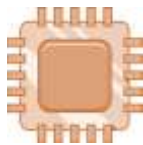
```
export __main
area start ,code, readonly
entry

__main
    cmp r0,r1
    beq cond

cond
    cmp r2,r3
    beq incr

incr
    add r4,r4,#1

end
```



MICROPROCESSOR AND ASSEMBLY LANGUAGE

Dr. Farbeh

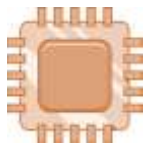
Homework 5



۴) برنامه‌ای بنویسید که مقدار ب م و ک م دو مقدار ذخیره شده در R0 و R1 را محاسبه کرده و به ترتیب در R2 و R3 ذخیره کند.

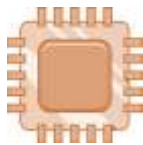
Register	Value
Core	
R0	0x0000000F
R1	0x00000014
R2	0x00000005
R3	0x0000003C
R4	0x00000010
R5	0x00000000
R6	0x00000005
R7	0x00000000
R8	0x00000000
R9	0x00000004
R10	0x0000003C
R11	0x00000003
R12	0x0000003C
R13 (SP)	0x20000400
R14 (LR)	0x004001DD
R15 (PC)	0x00400212
xPSR	0x61000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	1467
Sec	0.00012225

```
1  export __main
2  area start ,code, readonly
3  entry
4
5  __main
6  ldr r0, =15
7  ldr r1, =20
8
9  ldr r2, =1
10 ldr r4, =2
11 loop
12     udiv r5,r0,r4
13     mul r5,r5,r4
14     sub r5,r0,r5
15     ;;;;;;;;;;
16     udiv r6,r1,r4
17     mul r6,r6,r4
18     sub r6,r1,r6
19     cmp r6,r5
20     beq if0
21 adding
22     cmp r4,#15
23     add r4,#1
24     bne loop
25     bl kmm
26
27 if0
28     cmp r6,#0
29     beq is0
30     bl adding
31 is0
32     mov r2,r4
33     bl adding
34
```



```
35 kmm
36     cmp r0,r1
37     bhi bigger
38     mov r3,r1
39     bl kmm2
40
41 kmm2
42     mov r7,r3
43     mov r8,r3
44     bl kmm3
45 kmm3
46     udiv r9,r7,r0
47     mul r10,r9,r0
48     sub r7,r7,r10
49     udiv r11,r8,r1
50     mul r12,r11,r1
51     sub r8,r8,r12
52     cmp r7,r8
53     beq equall
54     bl kmm4
55 kmm4
56     add r3,#1
57     bl kmm2
58
59 bigger
60     mov r3,r0
61     bl kmm2
62 equall
63     cmp r7,#0
64     beq correct
65     bl kmm4
66 correct
67     b correct
68 end
```

۵) برنامه‌ای بنویسید که مقدار ذخیره شده در ثبات R0 را در یک آرایه ۱۰ عضوی به روش دودویی، جستجو کند (binary search) (فرض کنید که آرایه از قبل به صورت صعودی، مرتب شده است).



MICROPROCESSOR AND ASSEMBLY LANGUAGE

Dr. Farbeh

Homework 5



Registers

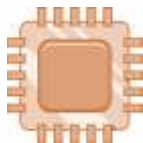
Register	Value
Core	
R0	0x0000000A
R1	0x00000009
R2	0x00000009
R3	0x004001A0
R4	0x00000000
R5	0x00000000
R6	0x00000009
R7	0x004001A9
R8	0x0000000A
R9	0x00000001
R10	0x00000002
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20000400
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x004001D8
xPSR	0x61000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	104
Sec	0.00000867

startup_SAM3N.s

system_SAM3N.c

q2.s

```
1      AREA myData , DATA
2      NUMBER EQU 10
3
4      TWO EQU 2
5      EXPORT __main
6      AREA myCode , CODE , READONLY
7      ENTRY
8      __main
9          LDR R0,=NUMBER
10         LDR R10,=TWO
11         MOV R1, #0
12         MOV R2, #9
13         LDR R3,=OUR_FIXED_DATA
14         MOV R9, #0
15         MOV R6, #0
16         B L1
17
18     OUR_FIXED_DATA
19         DCB 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
20
21
22
23 L1    CMP R1, R2
24        BHI FINISH
25        ADD R6, R1, R2
26        UDIV R6, R6, R10
27        LDR R7,=OUR_FIXED_DATA
28        ADD R7, R6, R7
29        LDRB R8, [R7]
30        CMP R8, R0
31        BCS L2
32        ADD R1, R6, #1
33        B L1
34
35 L2
36        CMP R8, R0
37        BLS L3
38        SUB R2, R6, #1
39        B L1
40 L3
41        MOV R9, #1
42        B FINISH
43
44 FINISH
45 HEAR B HEAR
46        END
47
48
```

۶) برنامه‌ای بنویسید که جمله‌ام دنباله‌ی فیبوناچی را در ثابت R1 قرار دهد (مقدار n در ثابت R0 قرار گرفته است).

Registers

Register	Value
Core	
R0	0x00000000
R1	0x00375F00
R2	0x0000000A
R3	0x00000000
R4	0x00000000
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20000400
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x0040019C
xPSR	0x61000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	66
Sec	0.00000550

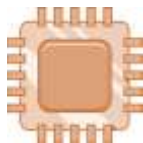
startup_SAM3N.s

system_SAM3N.c

q2.s

```
1      AREA myData , DATA
2      NUMBER EQU 10
3      EXPORT _main
4      AREA myCode , CODE , READONLY
5      ENTRY
6      _main
7      LDR r0 , =NUMBER
8      MOV r1 , #1
9      MOV r2 , #0
10     AGAIN
11     ADD r2 , r2 , #1
12     MUL r1 , r1 , r0
13     SUBS r0 , r0 , #1
14     BNE AGAIN
15     HEAR B HEAR
16     END
```

- مهلت ارسال تمرین ساعت ۲۳,۵۵ روز جمعه سوم دی ماه می باشد.
- سوالات خود را می توانید از طریق تلگرام از تدریسارهای گروه خود پرسید.
- کدهای اسمبلی را با استفاده از keil انجام دهید.
- ارائه پاسخ تمرین به بهتر است به روش های زیر باشد:



MICROPROCESSOR AND ASSEMBLY LANGUAGE

Dr. Farbeh

Homework 5



- (۱) ارائه اسکرین شات از کد و نتیجه اجرای آن در یک فایل pdf
 - (۲) قرار دادن فایل کد و اسکرین شات از نتیجه اجرای کد. در صورت استفاده از این روش حتما هر سوال را در پوشه جداگانه قرار دهید.
- فایل پاسخ تمرین را تنها با قالب **HW5-9731***.zip** یا **HW5-9731***.pdf** در مدل بارگذاری کنید.
 - نمونه: HW5-9731097