

تمرین سوم

جواب سؤال ۱)

$$f(n) = f(n-1) + f(n-2) \quad f(1) = 1$$

Load $M[0], R_8$

Load $0, R_9$

Load $\emptyset, R_{10} \rightarrow$ عدد صفر در R_{10} ریخته شود

Load $1, R_{11} \rightarrow$ اولین عدد از سری فیبوناچی
 $f(1) = 1$ در R_{11} ریخته می شود

Cmp R_8, R_9

jc L_1

Push R_{11}

Add R_{10}, R_{11}

Add $1, R_9 \rightarrow$ چون دستور Inc وجود ندارد از Add
برای افزایش شماره حلقه استفاده شد

Pop R_{10}

L_1 : Load $R_{11}, M[1]$

نصف حاصل نهایی در حافظه

```
Module register_block (input [7:0] idata,  
input clk, input reset, input [3:0] select, input read,  
input write, output [7:0] odata);  
reg [7:0] register [7:0]  
  
always @ (posedge clk) begin  
    IF (reset == 1)  
        register [select] = 8'b 0;  
  
    else begin  
        IF (write == 1)  
            register [select] = idata;  
  
        else IF (read == 1)  
            odata = register [select];  
  
        else  
            odata = 8'b 0;  
  
    end  
  
end  
  
end module
```

ebx

exit

ret

Fibonacci endp

اینکه حافظه دستور داده هدایتش امکان اجرای حواس دستورات فرام
گفته در واقع بهای اجرای پایلایین حساب این مرتبه
و با خب باشد و نه از حافظه به درستی استغاثه نه چون ممکنه کله داده.
داشته باشد اما حافظه دستورات فاس باشد باشد و نه که کی بخشی از حافظه
فاس بونه در فاس که بهشت نیار داریم در فاسی که داده ها زیاد باشد فوب
سیت در واقع .

جواب سؤال ۴)

اگر فرض کنیم که قرار است ۱۲۲ دستور در آدرس داشته باشیم پس $\log_2 122 = ?$
 بیت برای دستوریابی نیاز داریم و چون دو آدرس نیاز داریم و هر آدرس ۴ = ۱۰ بیت است
 پس جمعاً $4 + 4 + 7 = 15$ بیت برای ساختار دستورات این محاسبات نیاز داریم

7 bit	4 bit	4 bit
Opcode	Address	Address

اگر فرض کنیم که به صورت ترکیبی دستورات در آدرس یک آدرس و بدون آدرس داریم
 آن گاه ساختار دستورات به صورت

I	A	Opcode	Address 1	Address 2
1 bit	1 bit	n bit	4 bit	4 bit

بیت I صفر باشد دستور دو آدرس است

یک باشد دستور یک آدرس یا بدون آدرس است

بیت A صفر باشد بدون آدرس

یک باشد یک آدرس است

1 bit	Opcode	Address 1	Address 2	ساختار دستور دو آدرس
I	(1 + n) bit	4 bit	4 bit	
1 bit				

ارام در صفحہ بعد

I A opcode 4 bit Address مقدار دستور یک آدرس
 1 bit 1 bit $(4+x)$ bit

I A opcode مقدار دستور بدون آدرس
 $(8+x)$ bit

پس در حالت کلی $2^{(1+x)}$ دستور در آدرس 2^{x+4} دستور یک آدرس و 2^{8+x} دستور بدون آدرس داریم و جمعاً تعداد این‌ها باید بیشتر از 122 شود پس

$$2^{1+x} + 2^{x+4} + 2^{8+x} = 122$$

↓

$$2^x (2 + 16 + 256) = 128$$

↓

$$x = 0$$

پس کلاً دستورات 10 بیتی فوایدید (هاتنم) یک دستور در آدرس فوایم داشت

اگر فرض کنیم دستور بدون آدرس فوایم آن‌گاه نیاز 11 بیت A نخواهد بود

$$2^x + 2^{x+4} = 128$$

↓

$$x = 3 \text{ bit}$$

1 bit	n bit	4 bit	4 bit
I	opcode	Address 1	Address 2

ساختار دستور دارای

1 bit	n + 4 bit	4 bit
I	opcode	Address 2

بنابراین دایره بیت $1 + 3 + 8 = 12 \text{ bit}$ برای هر دستور لازم داریم.