

1

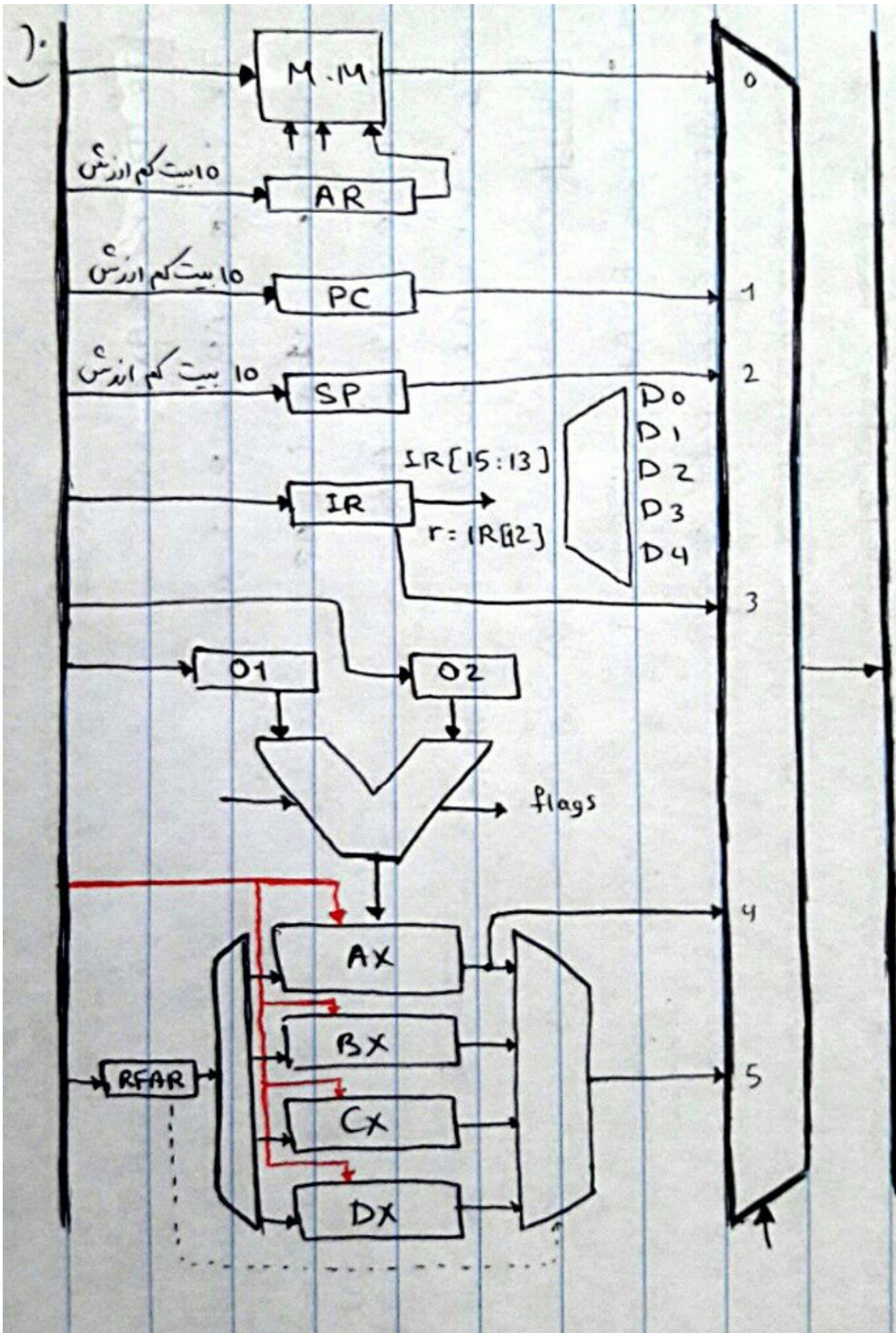
الف) از آن جایی که 6 دستور داریم، 3 بیت برای opcode لازم است و چون 4 بیت داریم، حداقل 2 بیت برای کد کردن بیت ها کافی است.

طولانی ترین دستور العمل مربوط به دستور ADD با عملوند حافظه است که 3 بیت برای opcode و 10 بیت برای آدرس می خواهد. در نتیجه 13 بیت حداقل لازم داریم. از آن جایی که هر کلمه حافظه 16 بیت است، قالب آدرس را به صورت 16 بیتی در نظر می گیریم.

* پ.ن 1: با توجه به فضای هر کلمه می توانستیم opcode را 6 بیتی هم در نظر بگیریم که این مورد و هم چنین 4 بیتی در نظر گرفتن آدرس بیت ها در طول قالب آدرس تأثیر نخواهد داشت. در نتیجه همواره طول بچینه برای قالب 16 بیت خواهد بود.

* پ.ن 2: می توانیم برای دستورهای ADD یک opcode یکسان در نظر بگیریم و ^{بودن} ~~immediate~~ immediate یا بودن به کمک بیت I مشخص شود.

MOV immediate	0 0 0 x x x x x i i i i i i i i i i
ADD immediate	0 0 1 0 x x x x i i i i i i i i i i
ADD Memory address	0 0 1 1 x x a a a a a a a a a a
SUB reg1, reg2	0 1 0 x x x x x x x x x R ₁ R ₁ R ₂ R ₂
PUSH reg	0 1 1 x x x x x x x x x x x R R
POP reg	1 0 0 x x x x x x x x x x x R R



Instruction Fetch

T0 : $AR \leftarrow PC$

T1 : $IR \leftarrow M[AR]$, $PC \leftarrow PC + 1$

Instruction Decode

T2 : Decode $IR[15:13]$ into D_0, D_1, D_2, D_3, D_4 , $r = IR[12]$

MOV

T3.D0 : $AX \leftarrow IR[7:0]$, $sc \leftarrow 0$

* نکته : در اینجا چون AX به صورت خاص مشخص شده می توانیم مستقیماً بایری $load$ آن را
محل کنیم و مقدار $immed$ را در آن بنویسیم. اما در میر داده بایری $load$ را به کمک $RFAR$
مشخص کرده بودیم که در دستور SUB به آن نیاز خواهیم داشت.
در صورتی که می خواستیم تنها بایری $load$ این ثبات به کمک خروجی دیکدر مشخص شود باید قبل از این
مرحله مقدار $RFAR$ را برابر صفر قرار می دادیم. هم چنین درودی 0 را هم در میر داده به
ماکس اصلی اضافه می کردیم.

ADD immediate

T3.D1.r': $O1 \leftarrow IR[7:0]$

T4.D1.r': $O2 \leftarrow AX$ * این خط به دلیل اینکه خروجی مستقیم از AX به بایس داریم امکان پذیر

T5.D1.r': $AX \leftarrow O1 + O2$, $SC \leftarrow 0$

ADD Memory

T3.D1.r: $AR \leftarrow IR[9:0]$

T4.D1.r: $O1 \leftarrow M[AR]$

T5.D1.r: $O2 \leftarrow AX$

T6.D1.r: $AX \leftarrow O2 + O1$, $SC \leftarrow 0$

SUB

T3.D2: $RFAR \leftarrow IR[1:0]$

T4.D2: $O1 \leftarrow RF[RFAR]$

T5.D2: $RFAR \leftarrow IR[3:2]$

T6.D2: $O2 \leftarrow RF[RFAR]$

MAHAN T7.D2: $AX \leftarrow O2 - O1$, $Flags \leftarrow O2 - O1$, $SC \leftarrow 0$

PUSH

T3.D3: $AR \leftarrow SP$

T4.D3: $RFAR \leftarrow IR[1:0]$

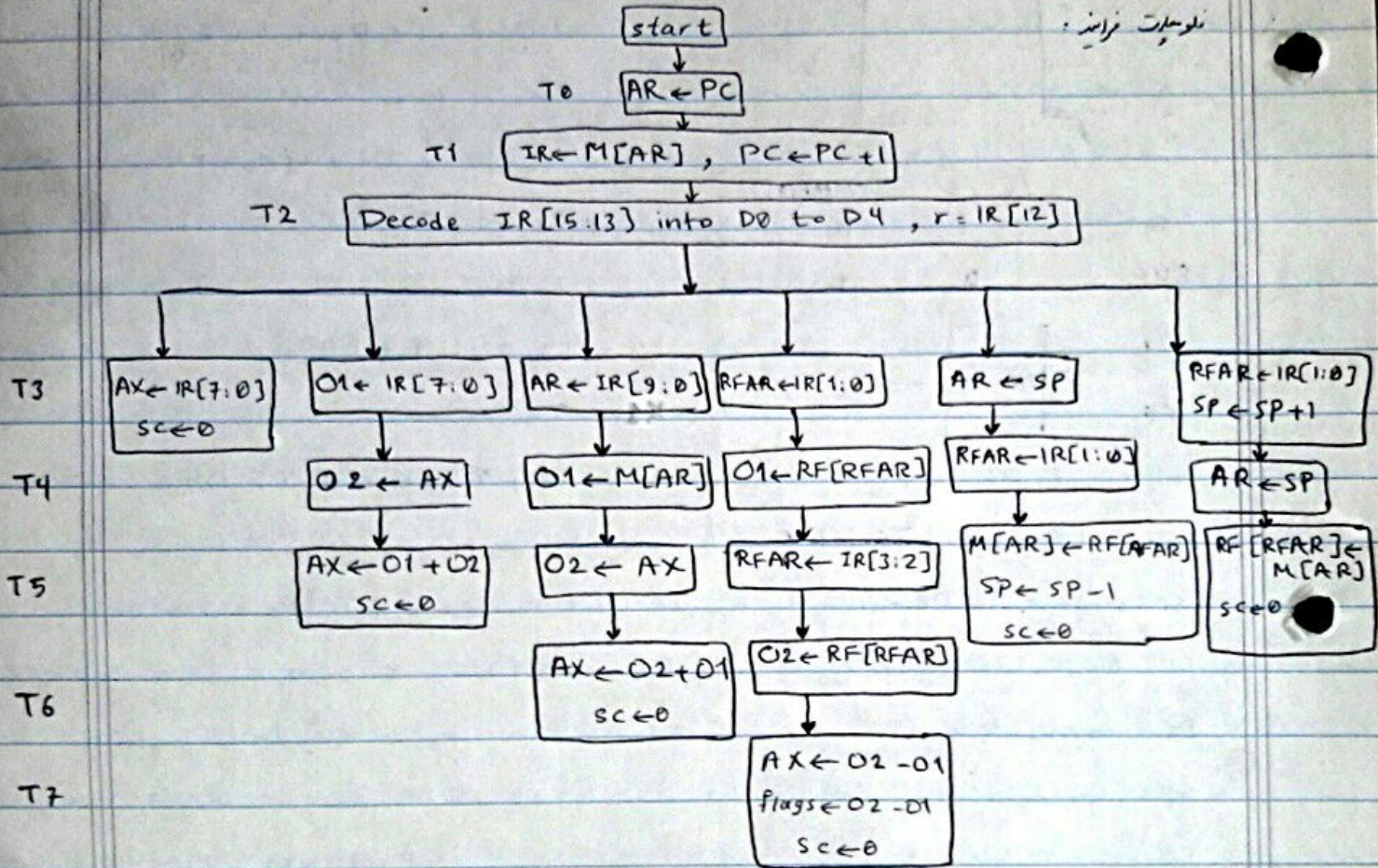
T5.D3: $M[AR] \leftarrow RF[RFAR]$, $SP \leftarrow SP - 1$, $SC \leftarrow 0$

POP

T3.D4: $RFAR \leftarrow IR[1:0]$, $SP \leftarrow SP + 1$

T4.D4: $AR \leftarrow SP$

T5.D4: $RF[RFAR] \leftarrow M[AR]$, $SC \leftarrow 0$



طولانی ترین دستور العمل مربوط به sub با 8 کلاک و کوتاه ترین مربوط به mov با 5 کلاک است.

واحد کشش :

	X_7	X_6	X_5	X_4	X_3	X_2	X_1	X_0	S_2	S_1	S_0
Main Memory	0	0	0	0	0	0	0	1	0	0	0
PC	0	0	0	0	0	0	1	0	0	0	1
SP	0	0	0	0	0	1	0	0	0	1	0
IR	0	0	0	0	1	0	0	0	0	1	1
AX	0	0	0	1	0	0	0	0	1	0	0
REG	0	0	1	0	0	0	0	0	1	0	1

Y1, Y0, COMMAND

ADD 0 1 0

SUB 1 0 1

Y1 = T7.D2

Y0 = T5.D1.r' + T6.D1.r

$X_7 = X_6 = 0$

$X_5 = T_4.D_2 + T_6.D_2 + T_5.D_3$

$X_4 = T_4.D_1.r' + T_5.D_1.r$

$X_3 = T_3.D_0 + T_3.D_1(r' + r)$

$+ T_3.D_2 + T_5.D_2 + T_4.D_3$

$+ T_3.D_4$

$X_2 = T_3.D_3 + T_4.D_4$

$X_1 = T_0$

$X_0 = T_1 + T_4.D_1.r + T_5.D_4$

$$SC-clear = T3.D0 + T5.D1.r' + T6.D1.r + T7.D2 + T5.D3 + T5.D4$$

$$AR-load = T0 + T3.D1.r' + T3.D3 + T4.D4$$

$$PC-increment = T1$$

$$SP-increment = T3.D4$$

$$SP-decrement = T5.D3$$

$$IR-load = T1$$

$$O1-load = T3.D1.r' + T4.D1.r + T4.D2$$

$$O2-load = T4.D1.r' + T5.D1.r + T6.D2$$

$$RFAR-load = T3.D2 + T5.D2 + T4.D3 + T3.D4$$

$$RF Decoder Enable = T5.D4$$

$$\text{MAHAN RF MUX Enable} = T4.D2 + T6.D2 + T5.D3$$

Main Memory

$$\text{Read} = T1 + T4.D1.r$$

$$+ T5.D4$$

$$\text{Write} = T5.D3$$

نمایش: $1\text{GHz} = 1000\text{MHz} \rightsquigarrow \text{MIPS} = \frac{1000}{1.25} = 800$

۲

تعدادها: $1 \times \frac{60}{100} + 3 \times \frac{20}{100} + 1 \times \frac{30}{100} + 2 \times \frac{10}{100} = \frac{150}{100} = 1.5$

۳

تعدادهاک در هر ثانیه

متوسط تعدادهاک برای هر دستور

$$\rightsquigarrow \frac{90\text{M}}{1.5} = 60\text{M}$$