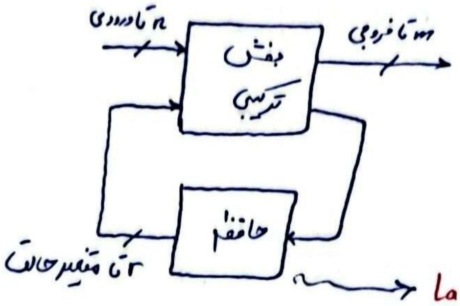
- مدار ترکیبی (combinational): خروجی آن فقط به ورودی‌های آن لحظه وابسته است و به گذشته مدار و به state مدار وابسته نیست.

← حافظه نداریم. مثل دیکودر - انکودر

- مدار ترتیبی ← حافظه برای خواندن و نوشتن وجود دارد و این حافظه باعث می‌شود که خروجی مدار نه فقط به ورودی‌های همان لحظه وابسته باشد و به گذشته مدار هم وابسته باشد.

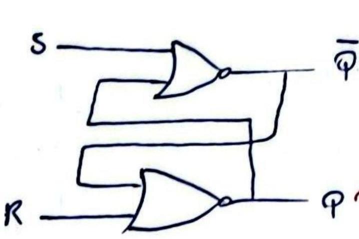


→ Latch یا Flip-Flop

* می‌توانند ۱ بیت ذخیره کننده

پس دو حالت دارند یا صفر هستند یا یک

- SR-Latch → تشکیل شده از ۲ تا NOR (لچ پایه)



Q ← بیت حافظه

← چرا حافظه است؟ چون فیدبک دارد ← از خروجی به ورودی مسیر داریم.

S	R	Q	Q^*	$\bar{Q}^*$	
0	0	0	0	1	hold
0	0	1	1	0	
0	1	0	0	1	Reset
0	1	1	0	1	
1	0	0	1	0	set
1	0	1	1	0	
1	1	0	0	0	forbiden غیرمجاز
1	1	1	0	0	

Q → حالت فعلی

Q^* یا Q^+ یا $Q(t+1)$ → حالت بعدی

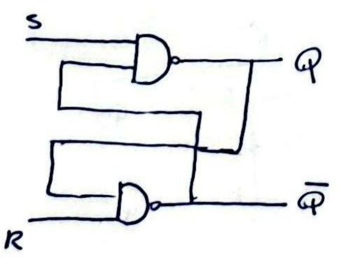
S	R	Q^*
0	0	Q
0	1	0
1	0	1
1	1	X

جدول مشخصه SR - جدول حالت SR

Q \ SR	00	01	11	10
0	0	0	X	1
1	1	0	X	1

$Q^* = S + \bar{R}.Q$ شرط SR=0

معادله مشخصه SR

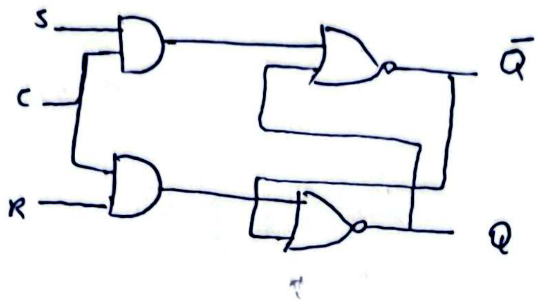


• ساخت SR-Latch با دو گیت "NAND"

S	R		Q^*
0	0	غیرمجاز	X
0	1	Set	1
1	0	Reset	0
1	1	hold	Q

- سوال: چگونه در SR-latch مقدار صفر را ذخیره کنیم؟ ($Q^* = 0$) کافی است $SR = 01$ بدهیم که $Q = 0$ شود و سپس $SR = 00$ بدهیم تا مقدار صفر حفظ شود.

- Gated - SR - latch



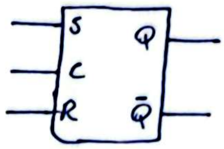
if $C = 1 \longrightarrow Q^* = S + \bar{R}.Q$

شرط $S.R = 0$

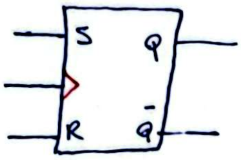
if $C = 0 \longrightarrow Q^* = Q$

به R, S توجه نکند و Q را حفظ کند.

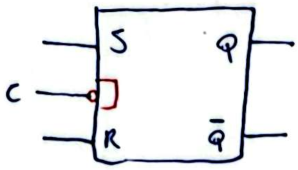
اگر ورودی کنترل (C) به کلاک متصل شود ← فلیپ فلاپ SR



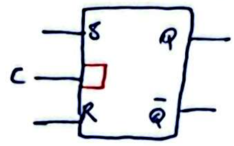
- نکته مهم:



حساس به لبه مثبت



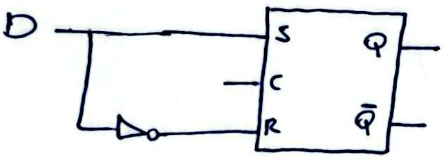
حساس به سطح منفی



حساس به سطح مثبت

- D-FF

Delay - Data

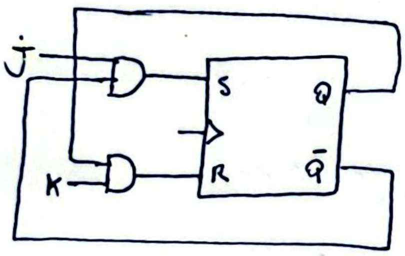


D	Q^*
0	0
1	1

$Q^* = D$

- JK-FF

← کلاک حساس به لبه می‌باشد و برای حساس به سطح تعریف نمی‌شود.



J	K	Q^*	
0	0	Q	hold
0	1	0	Kill
1	0	1	Jump
1	1	$\bar{Q}$	complement

Q \ JK	00	01	11	10
0	0	0	1	1
1	1	0	0	1

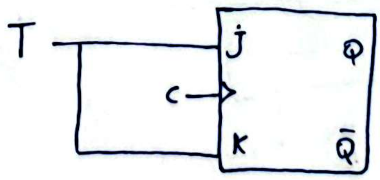
$Q^* = J\bar{Q} + \bar{K}Q$

- T-FF → (Toggle - Trigger)

← کلاک حساس به لبه هستند و برای حساس به سطح تعریف نمی‌شوند.

T	Q^*	
0	Q	hold
1	$\bar{Q}$	

$Q^* = T \oplus Q$



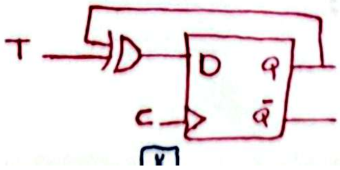
نکته: معمولاً از T-FF برای ساختن شمارنده و از D-FF برای ساخت رجیستر استفاده می‌شود.

مثال: با D-FF ، T-FF بسازید.

$Q^* = D$

$Q^* = T \oplus Q$

$\Rightarrow D = T \oplus Q$



** اعداد اعشاری

- برای نمایش اعداد اعشاری می‌توان از ۲ روش استفاده کرد: ① ممیز ثابت (Fixed Point) ② ممیز شناور (Floating Point)

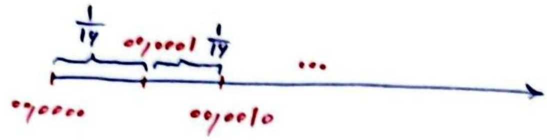
← دقت و Range خوبی ندارد.

مثال: فرض کنید یک سیستم ممیز ثابت ۶ بیتی بدون علامت که ۴ بیت آن برای ذخیره اعشار است.

الف) Range این سیستم را بیابید. یعنی Max و Min

$Min = 00,0000 = 0,0$

$Max = 11,1111 = 3\frac{15}{16} = 3,9375$



ب) دقت این سیستم را بیابید. دقت: فاصله بین اعداد متوالی قابل نمایش

ج) حداکثر خطا در این سیستم را بیابید. حداکثر خطا = $\frac{\text{دقت}}{2} = \frac{\frac{1}{16}}{2} = \frac{1}{32}$

- ممیز شناور: فرمت: S E M = F (مانتیس) = Fraction

S = Sign ← نشان دهنده علامت عدد

E = Exponent ← نشان دهنده نما یا توان عدد است

مقدار ممیز شناور برای حالت نرمال: $(-1)^S * 0,M * Base^E$

← در مبنا ۱۰ ← در مبنا Base است.

* جای ذخیره شده است و پیش فرض آن ۲ است.

نکته: E یک عدد علامت دار است پس می‌تواند مثبت یا منفی باشد ← باید برای آن یک سیستم در نظر بگیریم مثلا سیستم مکمل ۲ یا سیستم علامت مقدار

نکته: سیستم E همان Bias است. هر وقت بخواهیم عدد را برای E ذخیره کنیم خود آن عدد را ذخیره نمی‌کنیم بلکه با یک مقدار مثبت به اسم Bias جمع می‌کنیم و سپس ذخیره می‌کنیم. هنگامی که می‌خواهیم عدد ذخیره شده را به مبنا ۱۰ ببریم ابتدا آن را به صورت بی‌علامت محاسبه می‌کنیم و سپس Bias را از آن کم می‌کنیم.

نکته: اگر نما e بیتی باشد مقدار Bias برابر 2^{e-1} است. ← اگر نما ۴ بیتی باشد مقدار Bias برابر $2^{4-1} = 8$ است.

مثال: $(1111)_{Bias} = 15 - 8 = +7$ ← مقدار واقعی عدد

مثال: $(+2)_{\text{دهدهی}} \xrightarrow{+8} +10 \longrightarrow (1010)_{Bias}$

نکته: محاسبه در سیستم Bias راحت‌تر انجام می‌شود.

- نمایش اعداد ۴ بیتی در سیستم مکمل ۲ و سیستم Bias:

$Bias = 2^{e-1} = 2^{4-1} = 2^3 = 8$

← اگر بخواهیم در سیستم Bias، عدد دهدهی را ذخیره کنیم باید با این عدد جمع کنیم.

	مکمل ۲	Bias		مکمل ۲	Bias
+7	0111	1111	-1	1111	0111
+6	0110	1110	-2	1110	0110
+5	0101	1101	-3	1101	0101
+4	0100	1100	-4	1100	0100
+3	0011	1011	-5	1011	0011
+2	0010	1010	-6	1010	0010
+1	0001	1001	-7	1001	0001
0	0000	1000	-8	1000	0000

- مانتیس به صورت نرمال یا هنجار شده ذخیره می‌شود. یعنی اولین (با ارزش‌ترین) رقم مانتیس مخالف صفر است. یعنی اگر مانتیس در مبنا ۲ باشد با ارزش‌ترین رقم آن یک است $\longrightarrow$ ۱,۰ ....

مثال: با فرض اینکه نما بایاس شده ۵ بیتی است، مقدار عدد ممیز شناور زیر چند است؟

1100111101100000

S	E	M
1	10011	11011000

$e=5 \longrightarrow Bias = 2^{5-1} = 16$

$$\text{مقدار عدد} = (-1)^1 * 0/11011000 * 2^{10011} = (-1)^1 * (2^{-1}+2^{-2}+2^{-4}+2^{-5}) * 2^{19-16} = \;?$$

مثال: با فرض اینکه ۲۰ بیت فضا داشته باشیم و نما ۵ بیتی بایاس شده و مانتیس نرمال باشد عدد ۲,۴+ را نمایش بدهید.

$e-1$: $2^{e-1} = 2^{5-1} = 2^4 = 16$

$+2{,}4 = 10/0\overline{0110}$

$/4 * 2 = 0/8$

$/8 * 2 = 1/6$

$/6 * 2 = 1/2$

$/2 * 2 = 0/4$

۲ واحد به سمت چپ شیفت می‌دهیم

$/10\overline{0110} * 2^{+2} = /10011001100110$

S	E	M
0	10010 ($+16$)	10011001100110

$\xrightarrow{HEX} (4A666)_{16}$

مثال: با فرضیات مثال قبل، عدد $-\frac{4}{7}$ را نمایش بدهید.

$-\frac{4}{7} * 2 = 1\,\frac{1}{7}$

$\frac{1}{7} * 2 = 0\,\frac{2}{7}$

$\frac{2}{7} * 2 = 0\,\frac{4}{7}$

$-\frac{4}{7} = 0/\overline{100} * 2^0$

به فرم مانتیس نرمال است.

S	E	M
1	$0+16 = 16 = 10000$	10010010010010

$= (C2492)_{16}$

مثال: با فرضیات مثال قبل، عدد $-\frac{1}{6}$ را نمایش بدهید.

$-\frac{1}{6} * 2 = 0\,\frac{2}{6}$

$\frac{2}{6} * 2 = 0\,\frac{4}{6}$

$\frac{4}{6} * 2 = 1\,\frac{2}{6}$

$-\frac{1}{6} = 0/0\overline{01} = 0/0010101 \ldots$

دو واحد به سمت چپ shift می‌دهیم تا مانتیس نرمال شود.

$0/\overline{10} * 2^{-2}$

S	E	M
1	$-2+16 = 14 = 01110$	10101010101010

$= (BAAAA)_{16}$

مثال: با فرض اینکه فضای ذخیره ۲۰ بیت، نما ۵ بیت بایاس شده و مانتیس نرمال باشد.

الف) max و min مثبت را بیابید.

← عدد بلافاصله بعد از صفر

min مثبت:

S	E_{min}	M_{min}
0	00000	10000000000000

$$+\ 0/10000000000000 * 2^{0-16} = 2^{-1} * 2^{-16} = 2^{-17}$$

max مثبت:

S	E_{max}	M_{max}
0	11111	11111111111111

$$\longrightarrow 0/11111111111111 * 2^{31-16} = (1-2^{-14}) * 2^{15} = 2^{15} - 2$$

$$0/11111111111111 = 1 - 2^{-14}$$

ب) دقت را بیابید. دقت: فاصله بین اعداد متوالی قابل نمایش ← در سیستم ممیز شناور هرچه به سمت max مثبت پیش میرویم اختلاف بین اعداد بیشتر میشه چون توان بزرگ‌تر میشه.

ریزترین دقت: فاصله بین min مثبت و عدد بعد

بزرگ‌ترین (بدترین) دقت: فاصله بین max مثبت و عدد قبلی

$$\text{min مثبت} = 0/1000\ldots0 * 2^{-16}$$

$$\text{عدد بعدی} = 0/1000\ldots1 * 2^{-16}$$

$$\xrightarrow{\text{اختلاف}} \underbrace{0/00\ldots1}_{2^{-14}} * 2^{-16} = 2^{-30}$$

$$\text{max مثبت} = 0/111\ldots1 * 2^{+15}$$

$$\text{عدد قبلی} = 0/111\ldots0 * 2^{+15}$$

$$\xrightarrow{\text{اختلاف}} \underbrace{0/000\ldots1}_{2^{-14}} * 2^{+15} = 2$$

- عملیات ممیز شناور:

جمع و تفریق: $0/M_1 * 2^{E_1} \pm 0/M_2 * 2^{E_2}$ ← ابتدا باید توان (نما) را یکسان کنیم ← نمای کوچک را بزرگ می‌کنیم و مانتیس آن را به سمت راست shift میدهیم. بعد که نماها یکسان شد، یکی از نماها را می‌نویسیم و مانتیس ها را جمع و تفریق می‌کنیم. حال اگر مانتیس حاصل نرمال نبود با shift آن را نرمال می‌کنیم.

مثال: $0/1110 * 2^{1001} + 0/1100 * 2^{1010}$

با فرض مانتیس و توان ۴ بیتی هستند.

$$0/0111 * 2^{1010} + 0/1100 * 2^{1010}$$

$$= 1/0011 * 2^{1010} \xrightarrow[\text{مانتیس}]{\text{نرمال کردن}} 0/10011 * 2^{1011}$$

$$\xrightarrow{\text{قطع کنیم}} 0/1001 * 2^{1011}$$

$$\xrightarrow{\text{گرد کنیم}} 0/1010 * 2^{1011}$$

0/1

← باید شیفت راست بدیم ← معایب مانتو: سرریز در

مثال: $0.1111 * 2^{1001} - 0.1100 * 2^{1001} = \underbrace{0.0011}_{} * 2^{1001} \xrightarrow[\text{چپ بریم}]{\text{۲ بار شیفت}} 0.1100 * 2^{0111}$

نرمال نیست ← زیرسرریز در مانتیس ← جلو بیت‌ها با ارزش مانتیس صفر شده‌اند.
underflow

- ضرب ممیز شناور: $(0.M_1 * 2^{E_1}) * (0.M_2 * 2^{E_2})$

$$(0.M_1 * 0.M_2) * 2^{E_1+E_2-Bias}$$

$bias = 2^{e-1} = 2^{k-1} = 8$

$3-8=-5 \qquad 12-8=+4$

مثال: $(0.1001 * 2^{0011}) * (0.1000 * 2^{1100}) = 0.01001000 * 2^{1111-1000} = 0.1001000 * 2^{0111-\ldots}$ $\quad 7-8=-1$

$= 0.1001 * 2^{0110}$

← زیرسرریز مانتیس ← شیفت به سمت چپ

- تقسیم ممیز شناور: $(0.M_1 * 2^{E_1}) \div (0.M_2 * 2^{E_2}) = \frac{0.M_1}{0.M_2} * 2^{E_1-E_2+bias}$

مثال: $(0.1110 * 2^{1101}) \div (0.1001 * 2^{1001}) = \frac{0.1110}{0.1001} * 2^{0100+1000=1100} = 1.1000 * 2^{1100} = 0.1100 * 2^{1101}$

$$\begin{array}{r|l} 1110.0000 & 1001 \\ -1001 & \overline{1.1000} \\ \hline 1010 & \\ 1001 & \\ \hline 1000 & \end{array}$$

چون تا ۴ رقم اعشار می‌خواهیم انجام بدهیم ۴ تا صفر می‌گذاریم.

← در مانتیس سرریز پیش آمده که خطا نیست با شیفت به سمت راست درست می‌شود.

نکته: در جمع و تفریق اعداد ممیز شناور، امکان سرریز و زیرسرریز در مانتیس است.

نکته: در عملیات ضرب ممیز شناور، امکان زیرسرریز در مانتیس است.

نکته: در عملیات تقسیم ممیز شناور، امکان سرریز در مانتیس است.

نکته: در تمام عملیات ممیز شناور (جمع، تفریق، ضرب، تقسیم) هم امکان سرریز و هم امکان زیرسرریز در نما است.

← نمای حاصل از مینیمم نمای قابل ذخیره، کوچک‌تر شده است. چنین عدد بسیار کوچک به صفر است که می‌توان به صفر تقریب زد. البته حالتی است که در استانداردها این اعداد را به صورت denormal ذخیره می‌کنند.

← نمای حاصل از ماکزیمم نمای قابل ذخیره بیشتر شده است. چنین عددی قابل ذخیره نیست و $V = 1$

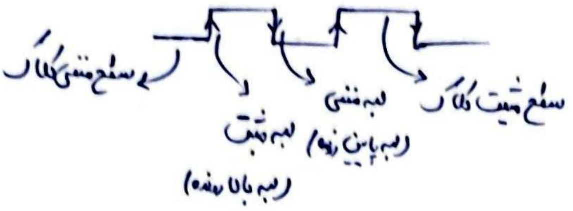
- استاندارد IEEE 754 برای نمایش اعداد ممیز شناور

		S	E	M
- فرمت تک‌دقتی (واحد) ← ۳۲ بیتی "single precision"	bias = 127 ←	1	8	23
- فرمت دقت مضاعف ← ۶۴ بیتی "double precision"	bias = 1023 ←	1	11	52

- بایاس در استاندارد IEEE ← $2^{e-1}-1$

- نرمال در استاندارد IEEE ← 1. ← بیت مخفی ← این بیت را ذخیره نمی‌کنیم ← implicit (ضمنی)
(hidden)

- کلاک پالس (کلاک سایکل): سیگنالی است که متناوباً صفر و یک می‌شود. — در ماشین‌های ترتیبی وجود دارد و وظیفه سنکرون کردن را بر عهده دارد.



⟹ از یک لبه مثبت تا لبه مثبت بعد را یک clock pulse می‌گویند.

واحد ثانیه. مدت زمان یک سیکل ساعت = clock cycle Time = Period = T

$1ms = 10^{-3}s$ $\quad 1\mu s = 10^{-6}s$ $\quad 1ns = 10^{-9}s$ $\quad 1ps = 10^{-12}s$

- فرکانس کلاک = نرخ کلاک = تعداد clock pulse در ثانیه $= \frac{1}{T} = \frac{1}{\text{clock cycle Time}} =$ clock Rate → واحد: هرتز (HZ)

$1KHZ = 10^{3} HZ$ (کیلو) $\quad 1MHZ = 10^{6} HZ$ (مگا) $\quad 1GHZ = 10^{9} HZ$ (گیگا) $\quad 1THZ = 10^{12} HZ$ (ترا)

- MIPS: Million Instruction Per Second = تعداد میلیون دستور در ثانیه

- CPI: clock per instruction = $\frac{\text{تعداد کلاک}}{\text{تعداد دستور}} = \sum \left(\text{تعداد پالس هر کلاس} * \text{درصد هر کلاس} \right)$

- IPC: instruction per clock

- یک دستور از تعداد کلاک درست شده.

- یک برنامه از تعداد دستور درست شده.

- متوسط مدت زمان اجرا یک دستور $= CPI * T$

- مدت زمان اجرا یک برنامه = تعداد دستور $* \; CPI * T$

Ic = instruction count

- $$MIPS = \frac{\text{تعداد میلیون دستور}}{\underset{(s)}{\text{زمان اجرای برنامه}}} = \frac{\text{تعداد دستور}}{\text{زمان اجرای برنامه}(\mu sec)} = \frac{\text{instruction count}}{(\text{instruction count}) * \underset{(\mu sec)}{CPI * T}} = \frac{1}{\underset{(\mu sec)}{CPI * T}} = \frac{\overset{(MHZ)}{\text{clock Rate}}}{CPI}$$

میکرو — میکرو — میکرو — مگا

- تسریع ماشین A نسبت به ماشین B (speed up) $= \frac{\text{سرعت ماشین } A}{\text{سرعت ماشین } B} = \frac{\text{زمان اجرای ماشین } B}{\text{زمان اجرای ماشین } A} = \frac{IC_B * CPI_B * T_B}{IC_A * CPI_A * T_A}$

- قانون امدال: اگر کسر f $(0 \le f \le 1)$ از یک برنامه P برابر سریع شود کل برنامه چند برابر سریع می‌شود؟

$$\text{speed up} = \frac{\text{زمان اولیه}}{\text{زمان پس از بهبود}} = \frac{t}{(1-f) * t + \frac{ft}{P}} = \frac{1}{(1-f) + \frac{f}{P}}$$

- FLOPS: Floating Point operations Per second → متوسط تعداد عمل ممیز شناور در ثانیه

MFLOPS $= 10^{6}$ $\quad$ GFLOPS $= 10^{9}$ $\quad$ TFLOPS $= 10^{12}$

مثال: یک برنامه حاوی ۲۰۰۰۰ دستور روی ۲ ماشین A و B اجرا شده است. ماشین A دارای CPI = 8 و ماشین B دارای CPI = 4 می‌باشد. نرخ کلاک ماشین A برابر ۲۰۰ MHZ است و سایکل تایم ماشین B برابر ۱۰ ns می‌باشد. کدام ماشین و چند برابر سریع‌تر است؟

$$\text{تسریع ماشین A نسبت به ماشین B} = S = \frac{\text{سرعت ماشین A}}{\text{سرعت ماشین B}} = \frac{\text{زمان اجرای ماشین B}}{\text{زمان اجرای ماشین A}} = \frac{\cancel{IC_B} * CPI_B * T_B}{\cancel{IC_A} * CPI_A * T_A} =$$

$$\frac{4 * 10 * 10^{-9}}{8 * \frac{1}{200} * 10^{-6}} = \frac{3}{2} = 1.5 > 1$$

سرعت ماشین A، 1.5 برابر سرعت ماشین B است.

مثال: اگر ۴۰ درصد زمان یک برنامه ۵ برابر سریع‌تر شود، کل برنامه چند برابر سریع‌تر می‌شود؟

t : زمان اجرای اولیه برنامه

$$\text{زمان اجرای برنامه پس از بهبود}: \ 0.6t + \frac{0.4t}{5} = 0.68t$$

$$\text{تسریع} = \frac{\text{زمان قبل از بهبود}}{\text{زمان بعد از بهبود}} = \frac{t}{0.68t} = \frac{100}{68}$$

مثال: یک پردازنده دارای ۴ کلاس دستور است. برنامه‌ای روی این پردازنده اجرا شده که در جدول زیر مشخصات آن آمده است. اگر نرخ کلاک پردازنده ۲۰۰ MHZ باشد MIPS را بیابید.

کلاس دستور	درصد هر کلاس در برنامه	تعداد پالس کلاک هر دستور
A	۲۰	۶
B	۳۰	۴
C	۱۰	۱۴
D	۴۰	۳

$$CPI = \text{هر دستور به طور متوسط چند کلاک می‌خواهد} = \frac{20}{100} * 6 + \frac{30}{100} * 4 + \frac{10}{100} * 14 + \frac{40}{100} * 3 = \underline{5}$$

$$T = \text{مدت زمان هر پالس} = \frac{1}{200}\ \mu s =$$

$$\text{متوسط زمان اجرای هر دستور} = 5 * \frac{1}{200} = \frac{1}{40}\ \mu s$$

۱ دستور $\frac{1}{40}\ \mu s$

? $1s = 10^6\ \mu s$

$$? = 40 * 10^6 = \underline{40}\ MIPS$$

مثال: فرض کنید در یک برنامه ۲۰۰ میلیون عمل ممیز شناور اجرا شده است. زمان اجرای برنامه ۱۰ ثانیه که در این زمان، ۵۰ درصد آن صرف اجرای دستورات ممیز شناور شده است و ۵۰ درصد دیگر زمان صرف سایر عملیات شده است. FLOPS را بیابید.

$$FLOPS = \frac{\text{تعداد دستورات ممیز شناور}}{\text{زمان اجرا (ثانیه)}} = \frac{200}{10} = 20\ M \Rightarrow MFLOPS = 20$$

مثال: فرض کنید ۵۰ درصد از زمان یک برنامه مربوط به محاسبات ممیز شناور و ۳۰ درصد زمان مربوط به ضرب اعداد صحیح است. برای بهبود این برنامه ۲ پیشنهاد وجود دارد:

t = زمان اولیه ، $0.5t$ = زمان ممیز شناور ، $0.3t$ = زمان ضرب صحیح ، $0.2t$ = زمان برنامه‌های دیگر

۱) سرعت ممیز شناور را دو برابر کنیم.

$$\text{زمان پیشنهاد اول} = 0.3t + 0.2t + \frac{0.5t}{2} = 0.75t$$

۲) سرعت ضرب صحیح را ۶ برابر کنیم.

$$\text{زمان پیشنهاد دوم} = 0.5t + 0.2t + \frac{0.3t}{6} = 0.75t$$

کدام پیشنهاد سرعت کلی برنامه را افزایش می‌دهد؟ هر دو یکسان است.