

((معاری کامپیوتر))

۱۴۰۳/۷/۷

سیستم نمایش اعداد

$$N_v = \sum_{i=-m}^{n-1} a_i r^i, \quad 0 \leq N$$

$$0 \leq a_i \leq r-1$$

$$(a_{n-1} \dots a_0 \cdot a_{-1} a_{-2} \dots a_{-m})_r$$

\uparrow MSD \downarrow LSD

سیستم نمایش اعداد سنتی - غیر سنتی

اعداد hex ($r=16$) ← {F و B و A و 9 و ... و 0 و 1}

تبدیل مبنای

۱) تبدیل از مبنای ۲ به ۱۰: محاسبه N_v

۲) تبدیل از مبنای ۱۰ به ۲: صیغ ← تقسیم متوالی ← کسری ← ضرب متوالی

۳) تبدیل از مبنای r_1 به r_2 : تعداد اعداد ← r_2^n و $0 \leq N \leq r_1^n - 1$

$$(\underbrace{\quad}_{\text{رقم } n_1})_{r_1} \rightarrow (\underbrace{\quad}_{\text{رقم } n_2})_{r_2} \Rightarrow r_1^{n_1} - 1 \leq r_2^{n_2} - 1 \Rightarrow n_2 = \left\lceil \frac{n_1 \log r_1}{\log r_2} \right\rceil$$

جمع و تفریق

می خواهیم یک مدار واحد برای جمع و تفریق ارائه دهیم $A - B = A + (-B)$

نمایش اعداد علامت دار (signed Digits)

$$16 = 1 \times 10 + (-6) \times 1 = 4 \quad -5 = (-1) \times 10 + 5 \times 1 = -5$$

یک نمایش غیر سنتی

* sign-Magnitude *

$$N = (s \ a_{n-2} \ a_{n-3} \ \dots \ a_0)_r \Rightarrow s=0 \rightarrow \text{مثبت}, \ s=r-1 \rightarrow \text{منفی}$$

$$r=10 \Rightarrow 012 \rightarrow +12 \text{ و } 912 \rightarrow -12 \quad r=2 \Rightarrow 0101 \rightarrow 5 \text{ و } 1101 \rightarrow -5$$

$$-(r^{n-1} - 1) \leq N \leq (r^{n-1} - 1)$$

* خواسته می مدار محاسباتی واحد را به نتیجه نمی رساند

* دارای مشکل +0 و -0 می باشد * بیت علامت جزئی از عدد نیست (بدون محاسبه)

** جمع و تفریق

① در صورت هم علامت بودن قدر مطلق هاجم می شوند و علامت قبلی گذاشته می شود

② در صورت ناهم علامت بودن قدر مطلق کوچکتر از قدر مطلق بزرگتر کم می شود و علامت قدر مطلق بزرگتر گذاشته می شود

* Diminished Radix Comp (r-1)'s Comp

$$N = (0 \ a_{n-2} \ a_{n-3} \ \dots \ a_0)_r \quad -N = r^n - 1 - N \quad -(r^{n-1} - 1) \leq N \leq r^{n-1} - 1$$

* در تمامی نهائش ها اعداد مثبت با صفر آغاز می شوند. * بیت علامت جزئی از عدد (بامعاسبه)

* مشکلات: ۱- مشکل +0 و -0. ۲- بار محاسباتی اضافه برای Carry برای

شیوهی تبدیل به قدر مطلق: تمام بیت ها به جز بیت علامت را Reverse کنید.

* Carry برای مجددا با عدد جمع می شود $\leftarrow +5 - 3 = +2$

* Radix Comp (r's Comp)

$$-N = r^n - N \quad -r^{n-1} \leq N \leq r^{n-1} - 1$$

شیوهی تبدیل به قدر مطلق تمام صفرها تا اولین یک نگه داشته می شوند و باقی Reverse می شوند.

$$\begin{array}{cccc} 2^3 & 2^2 & 2^1 & 2^0 \\ - & & & \\ 1 & 0 & 1 & 1 \end{array}$$

$$= (-2^3) + 2^1 + 2^0 = -8 + 3 = -5$$

* Carry برای دور ریخته می شود.

* یک شیوهی دیگر برای تبدیل به ۱۰:

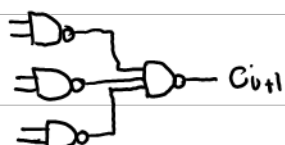
* مفهوم Overflow: حاصل عملیات بزرگتر از بازهی نهائش اولیه شود \leftarrow شیوه تشخیص: علامت دو عدد یکسان

باشد و علامت حاصل باقی ها متفاوت باشد. ۲- Carry ای که به رقم آخر وارد می شود و با Carry ای که خارج می شود یکسان نباشد.

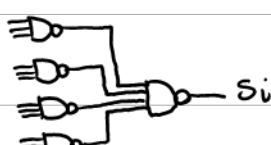
(مدار جمع کننده)

$$C_{i+1} = a_i b_i + a_i C_i + b_i C_i \quad S_i = a_i \oplus b_i \oplus C_i = \bar{a} \bar{b} C + \bar{a} b \bar{C} + a \bar{b} \bar{C} + a b C$$

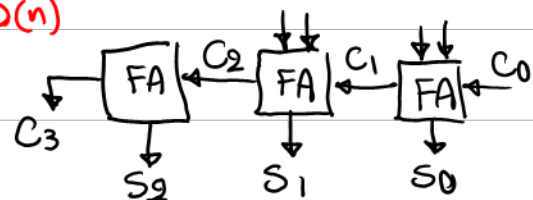
$$\Rightarrow T = n \times 2\Delta = O(n)$$

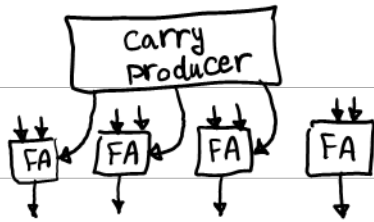


$$T = 2\Delta$$



$$T = 2\Delta \Leftrightarrow \Delta = \text{تاخیر هر گیت}$$

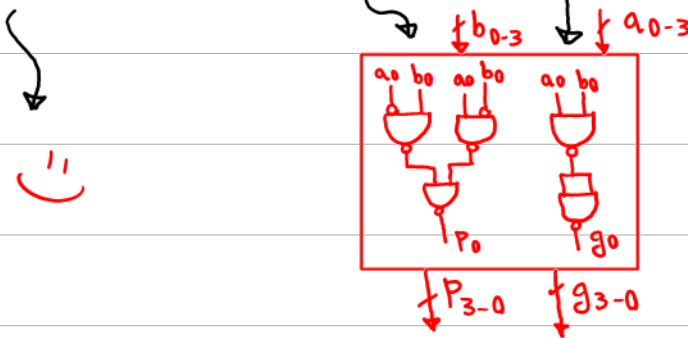




a_i	b_i	C_{i+1}	
0	0	0	→ Kill
0	1	C_i	→ propagate
1	0	C_i	
1	1	1	→ generate

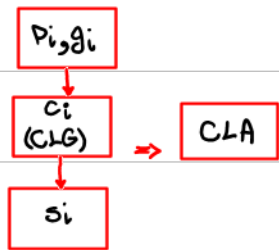
(CLA) Carry Look-Ahead Adder

$$C_{i+1} = g_i + P_i C_i \quad , \quad g_i = a_i \cdot b_i \quad , \quad P_i = a_i \oplus b_i \quad , \quad S_i = a_i \oplus b_i \oplus C_i = P_i \oplus C_i$$



$$T_{CLA}(4) = 6\Delta \quad T_{CPA}(4) = 8\Delta$$

$$T_{CLA}(16) = 12\Delta \quad T_{CPA}(16) = 32\Delta$$



مشکل این Adder (fan-out, fan-in) در جمع کسده های بزرگ است (قبل از Cascade)

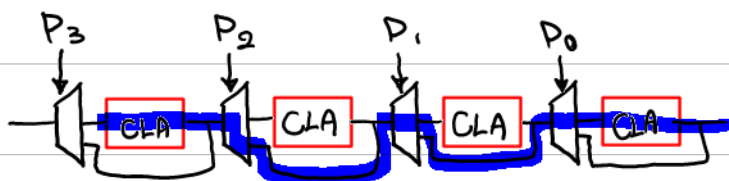
راه حل: Cascading \Leftarrow CLA(16) \leftarrow CLA \leftarrow CLA \leftarrow CLA

* در این حالت خود چهار قطعه CLA مجدا به حالت CLA تبدیل می شوند که G_i و P_i تعریف می شوند.

حالات مختلف که یک قطعه Carry تولید کند و بقیه آن را Propagate کند.

$$G_4 = g_3 + P_3 g_2 + P_3 P_2 g_1 + P_3 P_2 P_1 g_0 \quad , \quad P_4 = P_3 P_2 P_1 P_0 C_0$$

$$C_4 = G_0 + P_0 C_0 \quad , \quad C_8 = G_1 + P_1 C_4 \Rightarrow T_{CLA} = O(\log_2^n)$$



(CSA) Carry Skip Adder

Critical Pass \rightarrow

$$T_{CSK}(n, m) = m \cdot t_c + t_{mux} + \left(\frac{n}{m} - 2\right) t_{mux} + (m-1) t_c + t_s$$

تعداد قطعات هر CLA
تعداد سیگنال ها

* در این جمع کسده اگر قرار باشد تمام Adder های یک دسه Carry را انشمار دهند در همان ابتدا Carry توسط mux انتقال می یابد.

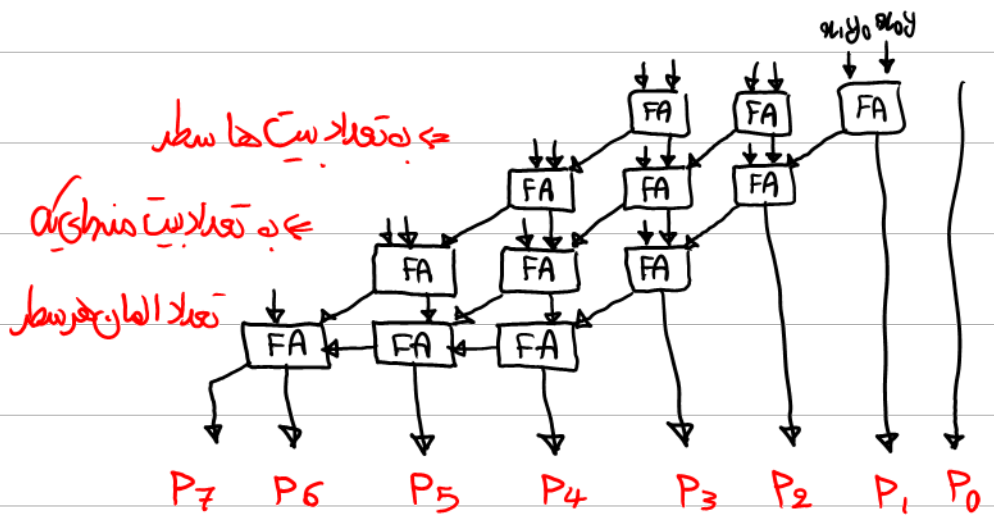
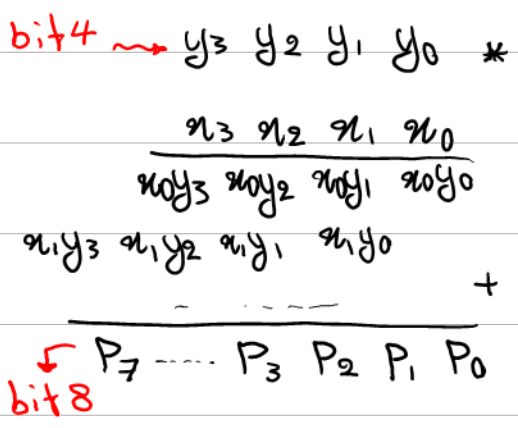
$$\frac{dT(n,m)}{dm} = m_{\text{optimum}} = \sqrt{\frac{t_{\text{max}} \times n}{2tc}}$$

$$\Rightarrow T_{\text{CSK}}(n, m) = O(\sqrt{n})$$

* می توانیم ساینز گروه ها (m) را متفاوت در نظر بگیریم که با توجه به مسیر بحرانی باید CLA های اول و آخر را با m کمتر در نظر بگیریم و CLA های میانی را با m بیشتر در نظر بگیریم

۱۴۰۳ / ۷ / ۱۴

ضرب کننده (Multiplier) - بدون علامت



$2\Delta = \text{تعداد FA}$ \Rightarrow $12\Delta = \text{Array Multiplier}$

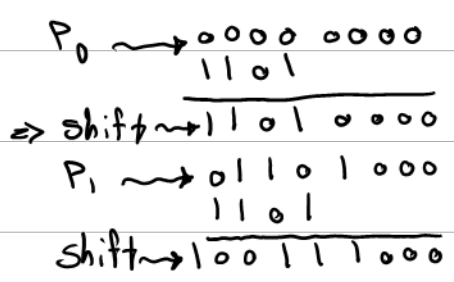
* Carry ها را به صورت قطری منتقل می کنیم. (Carry save Adder) - (به جز ردیف آخر)

ضرب کننده (Add & shift Multiplier)

$P=0$
for($i=0, i<4, i++$)
 $P_{i+1} = P_i + x_i 2^i y_i$

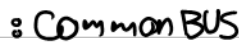
الگوریتم
بهره

$P=0$
for($i=0, i<4, i++$)
 $P_i = P_i + x_i y_i$
 $P_{i+1} = 2^{-1} \times P_i$

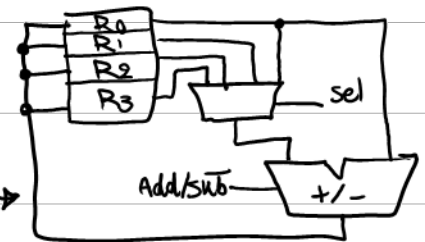


سیستم دیجیتال سکه ها

تمام امان های حافظه با یک لبه ی CLK کاری کسه. * به دو بخش مسیر داده و کنترل تقسیم می شود. مسیر داده، مجموعه ای از امان های محاسباتی و امان های حافظه که توسط BUS به یکدیگر متصل شده اند. گذرگاه (BUS): مجموعه ای از سیم ها که برای انتقال اطلاعات از یک نقطه به نقطه دیگر استفاده می شود.

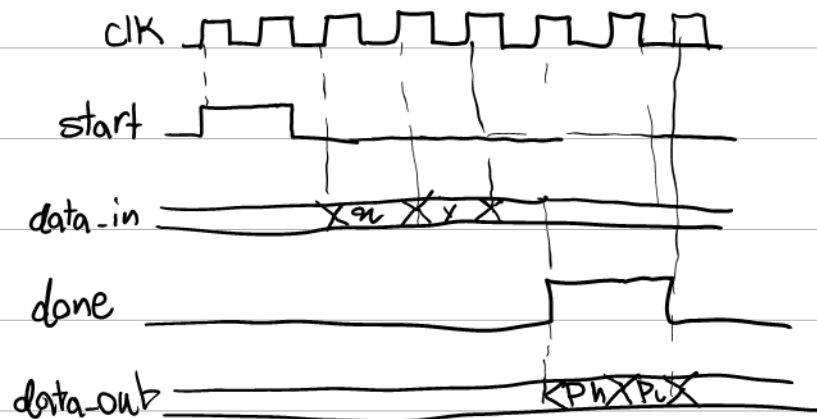


Data Path \rightarrow



```

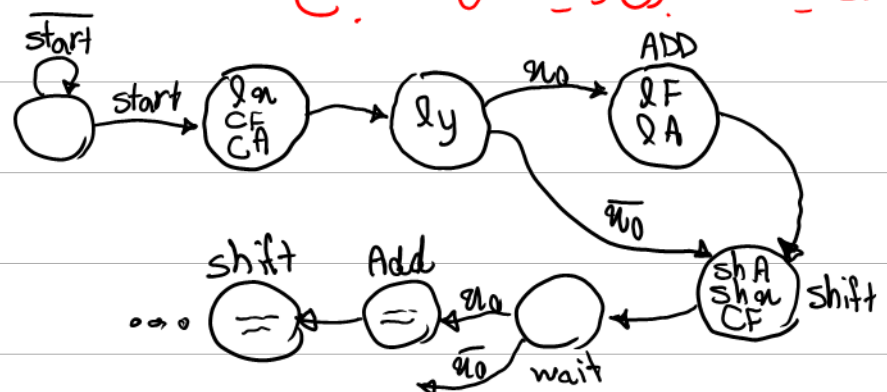
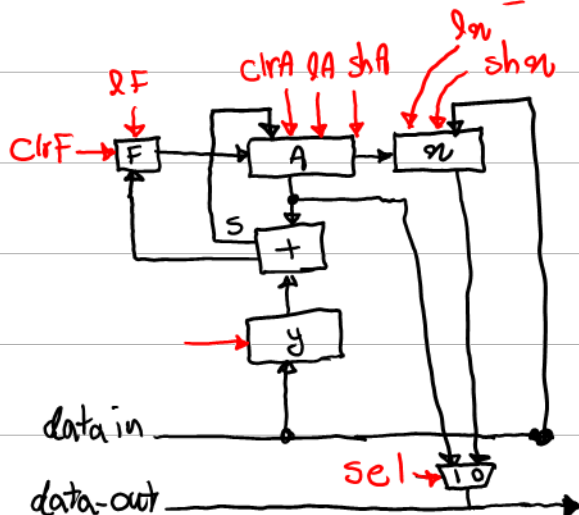
graph LR
    data_in[4-bit data-in] --> X((X))
    start[start] --> X
    X --> data_out[4-bit data-out]
    X --> done[done]
    Rst[Rst] --> X
    clk[clk] --> X
  
```



اجزای مورد نیاز

① جمع کسره ی ۴ بی ۲ ② سه رحسستر ۴ بی برای ۷۰۰ با قابلیت خود

For carry into FF C_1 (3)



$$30 \times y = 011110 \times y = 32y - 2y = 30y$$

* از راست به چپ، اولین یک را منفی کنیم سپس اهارا نادیده می گیریم و مجددا اولین صفر را جمع می کنیم

$x_i x_{i-1}$	
NOP	0
ADD	1
SUB	-1 (1)
NOP	0

$$-7 \times -9 = 011001 \downarrow \Rightarrow \begin{array}{r} 11001 \\ \hline 11001 \\ \hline 00111 \\ 00000 \\ 00000 \\ 00000 \\ \hline 11001 \\ 00111 \\ \hline \end{array}$$