# پاسخ تمرین شماره ۲ درس معماری کامپیوتر

امیر حسین عاصم یوسفی ۹۶۱۱۰۳۲۳

۲۸ اسفند ۱۳۹۷

### سوال ١:

برای حل این سوال باید به این توجه کرد که هر بلاک M در واقع یک  $Carry\ Select\ Adder\ (CSA)$  می باشد که اندازه تاخیر آن برابر با مجموع تاخیر مالتی پلکسر ها و تاخیر CRA ها می باشد که برای PF بیت به صورت زیر می باشد :

$$D_{64\;bit\;adder} = D_{CRA} + \left( \left( \frac{64}{M} - 1 \right) \times D_{MUX} \right)$$

همچنین باید این را در نظر گرفت که هر مالتی پلکسر یک تاخیر اضافه می کند (  $Fanout\ delay$  ) و اضافه می کند که این موضوع در تحلیل ما بسیار مهم است و به آن بر خواهیم گشت .

حال اگر پیچیدگی های پیاده سازی واقعی صرف نظر کنیم و فرض کنیم که تاخیر هر FA ۱ بیتی N برابر تاخیر یک مالتی پلکسر می باشد . بنابراین با توجه به تاخیر CRA داریم :

$$D_{CRA} = N \times M \times D_{MUX}$$
  
$$D_{CRA} = \left(\frac{64}{M} - 1\right) \times D_{MUX}$$

با توجه به عبارت های بالا به معادله زیر می رسیم :

$$NM^2 + M = 64$$
 (1)

که اگر در معادله بالا داشته باشیم N=2 (تاخیر هر جمع کننده ۱ بیتی ۲ برابر تاخیر یک مالتی پلکسر می باشد ) آن گاه برای M مقدار ۵.۴۱ به دست می آید که چون M باید یک مقدار صحیح باشد می توان دو مقدار را برای آن در نظر گرفت که به شرح زیر می باشد :

$$M = 5$$
 ,  $M = 6$ 

که اگر مقدار ۵ را در نظر بگیریم این جمع در ۱۳ مرحله انجام می شود پس داریم :

$$D_{adder} = 2 \times 5 + 12 = 22$$

بنابراین به ازای این مقدار تاخیر ما به اندازه ۲۲ مالتی پلکسر می باشد . به طور مشابه این اتفاق به ازای مقدار ۶ نیز می افتد .

با توجه به چیزی که در اول درباره تاخیر مالتی پلکسر گفته شده می توان فرض کرد که تاخیر یک جمع کننده ۱ بیتی برابر با تاخیر یک مالتی پلکسر با احتساب  $Fanout\ Delay$  می باشد بنابراین معادله ۱ به شکل زیر تغییر می کند:

$$1 \times M^2 + M = 64$$

که از معادله بالا مقدار ۷.۵۲ به دست می آید . که با توجه به این که M باید یک مقدار صحیح داشته باشد مقدار ۸ را برای آن در نظر می گیریم که در آن صورت تاخیر ما به اندازه 15 + 7 = 1 مالتی پلکسر می باشد که مقدار مینیمم می باشد .

. بنابراین به ازای M=8 به تاخیر مینیم می رسیم

## سوال ٢:

#### 

### سوال ٣:

نتدسنيتبدنستيدبنتدسيبنتسد

### سوال ۴:

با توجه به این که هر چه قدر فرکانس کاری بالاتر رود زمان بین دو کلاک کمتر می شود بنابراین اگر فرکانس از حدی بالاتر رود طول کلاک ها برای انجام عملیات ها کافی نمی باشد .

اگر بخواهیم حالتی از ضرب را در نظر بگیریم که به طولانی ترین کلاک نیاز دارد می توان حالتی را در نظر گرفت که یک عدد که هر ۳۲ بیت آن یک می باشد را در خودش ضرب کرد بنابراین در هر کلاک به عملیات ها زیر نیاز داریم:

۱. به یک عملیات از سوی واحد کنترلی نیاز داریم که چک می کند ضرب شونده صفر شده است یا نه

۲. بیت کم ارزش ضرب شونده را چک می کند

٣. يک دستور جمع صادر مي کند

۴. یک عملیات جمع انجام می شود

با توجه به این که تاخیر واحد کنترل برابر با  $\Lambda$  می باشد در هر کلاک  $\Upsilon$ ۴ میکروثانیه نیاز داریم تا دستورات واحد کنترلی درهر کلاک به درستی انجام شود .

حال برای جمع کردن چون از یک CSA ۸ بیتی استفاده کرده ایم که با این واحد های ۸ بیتی می توان دو عدد T بیتی را در ۴ مرحله جمع کرد و با توجه به فرمول تاخیر زیر :

$$D_{CSA} = \left(\frac{n}{k}\right) \times D_{FA} + (k-1) \times D_{MUX}$$

داريم :

$$k = 4$$
,  $n = 32$ ,  $D_{FA} = 5\mu s$ ,  $D_{MUX} = 4\mu s \Rightarrow D_{CSA} = (\frac{32}{4}) \times 5 + (4-1) \times 4 = 40 + 12 = 52\mu s$ 

بنابراین ۵۲میکروثانیه هم برای انجام شدن دستور جمع در هر کلاک نیاز داریم پس:

$$D_{Total} = 52 + 24 = 76\mu s$$

بنابراین طول هر کلاک باید ۷۶ میکروثانیه باشد تا عملیات ضرب به درستی انجام شود . ولی چون هم ضرب شونده و هم ضرب کننده ۳۲ بیتی هستند باید این مدار با فرکانسی کار کند که در ۳۲ کلاک بتواند کار را به اتمام برساند زیرا این مدار تا زمانی به کار خود ادامه می دهد که ضرب شونده صفر نشده باشد .

و با توجه به این که

Clock Cycle Time = 
$$76 \mu s = \frac{1}{Clock Rate}$$

بنابراین حداکثر فرکانسی که می تواند با آن کار کند برابر است با

$$Clock\ Rate = \frac{1}{76 \times 10^{-6}} \cong 13157 Hz$$

بنابراین با حداکثر فرکانس بالا کار کند .

### سوال ۵:

اگر هر کدام از اعداد A و B را به فرم زیر تعریف کنیم :

$$A = X_N X_{N-1} X_{N-2} \dots X_1 X_0$$
  
$$B = Y_N Y_{N-1} Y_{N-2} \dots Y_1 Y_0$$

حال با توجه به این که در الگوریتم ضرب ترکیبی هر بیت عدد A را با هر بیت عدد AND ، B می کنیم پس به حال با توجه به این که در الگوریتم AND نیاز داریم .

FA ) بعد از AND کردن این اعداد آن ها را با یک دیگر جمع می کنیم که بنابراین به N(N-1) جمع کننده ( N(N-1) بغد از N(N-1) جمع کننده ( N(N-1) بنیاز داریم بنابراین تعداد N(N-1) ها برابر است با

$$N(N-1) \times 2$$

بنابراین با توجه به این محاسبات برای یک ضرب کننده ترکیبی ۴ بیتی مقادیر به شکل زیر می باشد :

$$FA = 12 \Rightarrow HA = 12 \times 2 = 24$$
 
$$AND\ Gate = 4 \times 4 = 16$$

بنابراین با توجه به این تاخیر گیت AND برابر با ۳ واحد می باشد و این که تاخیر یک HA برابر با ۴ واحد می باشد بنابراین تاخیر یک ضرب کننده ترکیبی ۴ بیتی برابر است با :

$$D_{4bit\ Combinational\ Multiplier} = 24 \times 3 + 16 \times 4 = 72 + 64 = 136$$

بنابراین تاخیر برابر با ۱۳۶ واحد زمانی می باشد .

### سوال ۶:

با توجه به فرمول تاخیر  $Multi\ stage\ CSA$  که به شرح زیر می باشد :

$$D_{CSA} = (\frac{n}{k}) \times D_{FA} + (k-1) \times D_{Mux}$$

و همچنین با توجه به فرمول مساحت که به شرح زیر است :

$$A_{CSA} = n \times \left( \left( \frac{2k-1}{k} \right) \right) \times A_{FA} + (k-1) \times A_{Mux}$$

حال با ضرب دو مقدار به دست آمده در بالا به یک تابع دو متغیره بر حسب متغیرهای n و k می رسیم که به صورت زیر می باشد :

$$D_{CSA} \times A_{CSA} = ((\frac{n}{k}) \times D_{FA} + (k-1) \times D_{Mux}) \times (n \times (\frac{2k-1}{k}) \times A_{FA} + (k-1) \times A_{Mux})$$

که برای سادگی می توان (k-1) را k در نظر گرفت و مقدار 2k-1 را 2k در نظر گرفت پس تابع دو متغیره ما به صورت زیر ساده می شود :

$$D_{CSA} \times A_{CSA} = \left( \left( \frac{n}{k} \right) \times D_{FA} + (k) \times D_{Mux} \right) \times \left( n \times \left( \frac{2k}{k} \right) \times A_{FA} + (k) \times A_{Mux} \right)$$

که با انجام دادن این ضرب داریم:

$$H(n,k) = (\frac{2n^2}{k})D_{FA}A_{FA} + nD_{FA}A_{Mux} + 2nkD_{Mux}A_{FA} + K^2A_{Mux}D_{Mux}$$

اگر تابع دو متغیره بالا را بر حسب k کمینه کنیم به تعداد مراحلی می رسیم که area \* delay کمینه می شود . پس بر حسب k مشتق می گیریم برابر با صفر قرار میدهیم :

$$\frac{\partial H}{\partial k} = (-\frac{2n^2}{k^2})D_{FA}A_{FA} + 2nD_{Mux}A_{FA} + 2kA_{Mux}D_{Mux} = 0$$

که از این معادله مقدار k برابر است با k ۲.۲۵ که چون باید مقدار آن یک عدد صحیح باشد بنابراین مقدار k برای آن در نظر می گیریم .

بنابراین تاخیر یک جمع کننده ۸ بیتی برابر است با :

$$D_{CSA}=\left(\frac{n}{k}\right)\times D_{FA}+(k-1)\times D_{Mux}\;,\;n=8,\;k=2\;,D_{Mux}=3,D_{FA}=3\Rightarrow D_{CSA}=15$$

## سوال ٧:

برای CLA داریم :

$$P_i = X_i \oplus Y_i$$
 
$$G_i = X_i.Y_i$$
 
$$C_{i+1} = G_i + P_iC_i$$
 
$$S_i = P_i \oplus C_i$$

و با توجه به دوعدد داده شده داریم:

$$x_0 = 0, x_1 = 1, x_2 = 1, x_3 = 0$$
  
 $y_0 = 1, y_1 = 0, y_2 = 0, y_3 = 1$ 

: بنابراین برای  $P_i$  ها داریم

$$P_0 = x_0 \oplus y_0 = 1$$
  
 $P_1 = x_1 \oplus y_1 = 1$   
 $P_2 = x_2 \oplus y_2 = 1$   
 $P_3 = x_3 \oplus y_3 = 1$ 

: برای  $G_i$  ها داریم

$$G_0 = x_0.y_0 = 0$$
  
 $G_1 = x_1.y_1 = 0$   
 $G_2 = x_2.y_2 = 0$   
 $G_3 = x_3.y_3 = 0$ 

: برای  $C_i$  ها داریم

$$C_0 = 1 \Rightarrow C_1 = G_0 + P_0C_0 = 1$$
  
 $C_2 = G_1 + P_1C_1 = 1$   
 $C_3 = G_2 + P_2C_2 = 1$ 

: برای  $S_i$  ها داریم

$$S_0 = P_0 \oplus C_0 = 0$$
  
 $S_1 = P_1 \oplus C_1 = 0$   
 $S_2 = P_2 \oplus C_2 = 0$   
 $S_3 = P_3 \oplus C_3 = 0$ 

بنابراين

$$C_3 = 1$$
$$S_3 = 0$$

### سوال ۸:

### قسمت اول:

مراحل کار هر ضرب کننده را می توان به صورت زیر تقسیم بندی کرد:

۱. تولید کردن ضرب های جزئی (Partial Products)

۲. جمع کردن ضرب های جزئی

۳. جمع نهایی و به دست آوردن حاصل ضرب

Divide & اما تفاوت و برتری روش Wallace Tree در قسمت دوم نهفته است که ایده ی آن بر اساس Wallace Tree اما تمام تفاوت و برتری روش Conqure می باشد که به جای این که ضرب های جزئی را تک به تک با یک دیگر جمع کند از واحد های Conqure (Carry Save Adder) استفاده می کند .

مزیت های این روش به شرح زیر می باشد:

١. سرعت بالا

۲. توان مصرفی پایین

۳. تاخیر کمتر

۴. تعداد سطوح منطقی کمتری برای جمع کردن در آن وجود دارد

هدف استفاده از این روش رسیدن به سرعت بیشتر و در عین حال داشتن توان کمتر می باشد و امروزه در بسیاری از موارد مانند محاسبات ممیز شناور و تحلیل و پردازش سیگنال کاربرد بسیار فراوانی دارد .

#### قسمت دوم:

روش انجام آن به این شکل است که ابتدا بیت های Multipicant را با بیت های Multiplier تک به تک And کنیم تا به ضرب های جزئی برسیم و در مرحله با استفاده از CSA ها ضرب های جزئی را جمع می کنیم و در نهایت برای جمع آخر از یک CPA (Carry Propagate Adder) استفاده می کنیم .

بنابراین برای طراحی چنین مداری که دو عدد ۴ بیتی را در یک ضرب کند به اجزای زیر نیاز داریم:

۱. گیت ۱۶ AND عدد

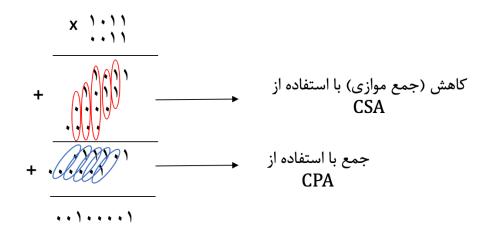
۲. جمع کننده CSA ۵ عدد

۳. جمع کننده CPA ا عدد

#### حاصل ضرب خواسته شده :

با توجه به شماره دانشجویی داریم :  $Multiplier=323~\%~32=3=(0011)_2~$  و با توجه به صورت سوال داریم :  $Multiplicant=(1011)_2~$ 

پس :



همان طور که می توان دید حاصل ضرب برابر با عدد ۳۳ (در مبنای ۱۰ ) می باشد .