



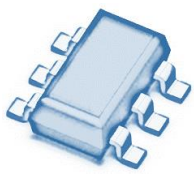
دانشکده مهندسی
کامپیوتر و فناوری اطلاعات

۱۳۹۸/۲/۲۸

پاسخ تکلیف ۴

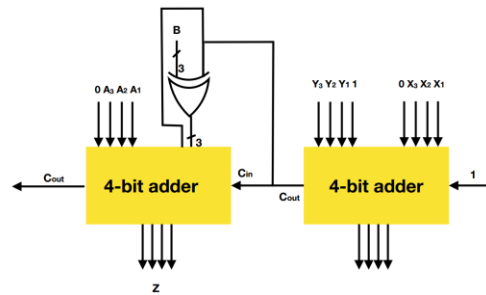
جمع کننده، ضرب کننده، تقسیم کننده

معماری کامپیوتر



(۱) مدار زیر را در نظر بگیرید. به جای علامت‌های سوال، عبارات‌های مناسب قرار دهید. A, B, X, Y سه بیتی هستند)

$if X + 2Y > ? \text{ then } Z = ? \text{ else } Z = ?$



If $x + 2y > 13$ then $z = A - B$ else $z = A + B$

(۲) بر روی روش تقسیم به روش غیربازیافتی (Non-Restoring) تحقیق کنید و به طور خلاصه روش را توضیح دهید. سپس تقسیم ۱۳۰ بر ۱۱ را به روش غیربازیافتی انجام دهید. (اعداد را هشت بیتی و به صورت بی‌علامت در نظر بگیرید)

در این روش دیگر مرحله بازیافتی وجود ندارد و به جای آن با مقدار منفی باقیمانده کار می‌کند. روش کار آن به صورت زیر است: (فرض کنید Q مقسوم و M مقسوم علیه باشد. (رجیستر A را در ابتدا 0 قرار دهید))

اگر A منفی باشد:

جفت رجیسترهای AQ را یکبار به چپ شیفت می‌دهد.

مقدار M را با A جمع می‌کند.

اگر A مثبت باشد:

جفت رجیسترهای AQ را یکبار به چپ شیفت می‌دهد.

مقدار M را از A کم می‌کند.

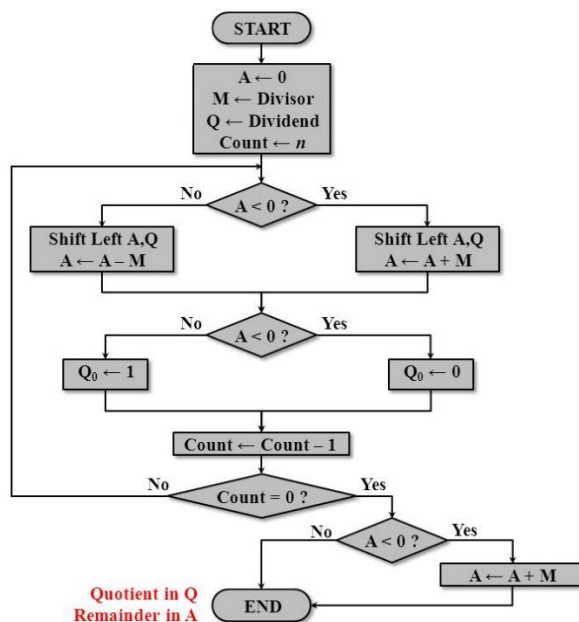
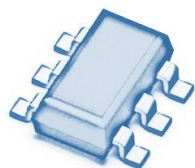
حال اگر مقدار جدید A منفی باشد، کم ارزش ترین بیت Q را 0 قرار می‌دهد. در غیر اینصورت (A مثبت)

کم ارزش ترین بیت Q 1 می‌شود.

این مراحل n بار تکرار می‌شود. (n تعداد بیت‌های مقسوم و مقسوم علیه)

بعد از مرحله n مقدار خارج قسمت در Q قرار دارد. اگر A مثبت باشد، مقدار باقیمانده است. اما اگر

منفی باید مقدار M را به آن اضافه کنیم تا مقدار صحیح باقیمانده را نشان دهد.



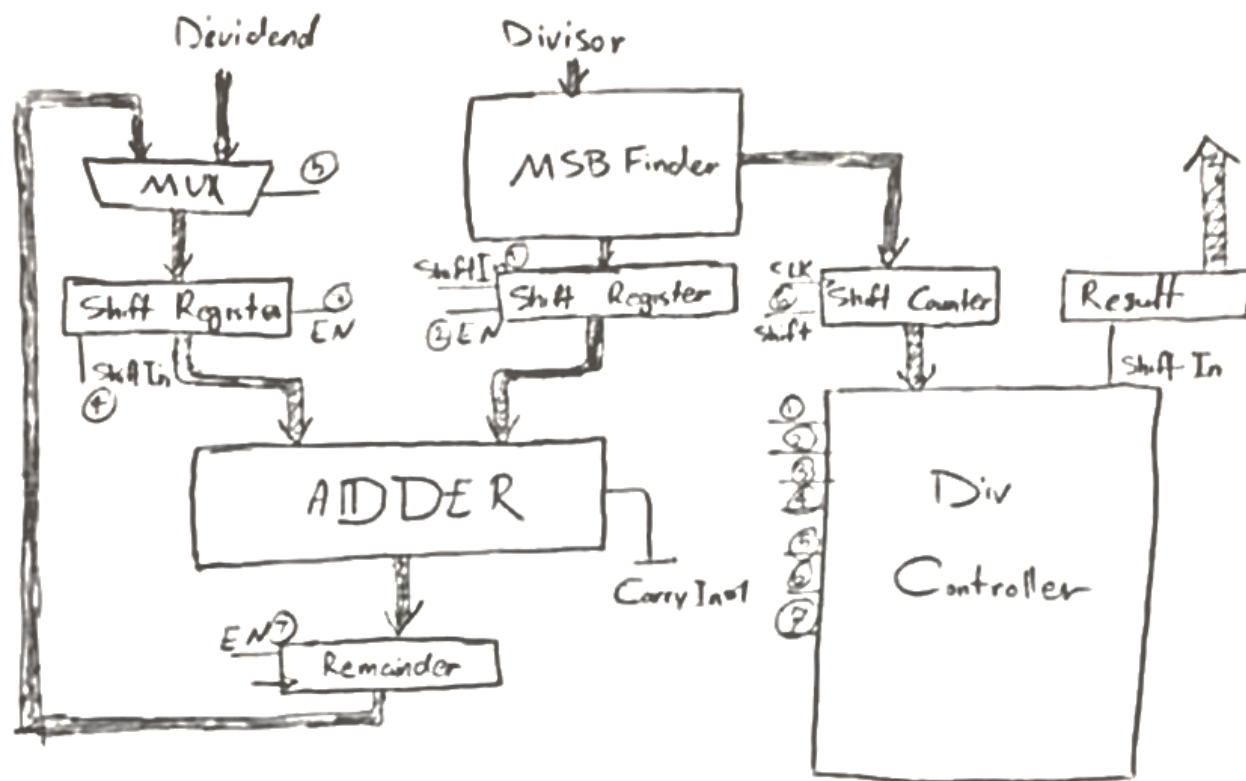
iteration	M	Steps	A	Q
0	000001011	Initial value	00000000	10000010
	000001011	$A > 0 : \text{shl}(AQ)$	00000001	0000010
		$A = A - M$	11110110	0000010
		$A < 0, Q_0=1$	11110110	00000100
2	000001011	$A < 0 : \text{shl}(AQ)$	11101100	0000100
		$A = A + M$	11110111	0000100
		$A < 0, Q_0=0$	11110111	00001000
3	000001011	$A < 0 : \text{shl}(AQ)$	11101110	0001000
		$A = A + M$	11111001	0001000
		$A < 0, Q_0=0$	11111001	00010000
4	000001011	$A < 0 : \text{shl}(AQ)$	11110010	0010000
		$A = A + M$	11111101	0010000
		$A < 0, Q_0=0$	11111101	00100000
5	000001011	$A < 0 : \text{shl}(AQ)$	11111010	0100000
		$A = A + M$	00000101	0100000
		$A > 0, Q_0=1$	00000101	01000001
6	000001011	$A > 0 : \text{shl}(AQ)$	00001010	1000001
		$A = A - M$	11111111	1000001
		$A < 0, Q_0=0$	11111111	10000010
7	000001011	$A < 0 : \text{shl}(AQ)$	11111111	0000010
		$A = A + M$	00001010	0000010
		$A > 0, Q_0=1$	00001010	00000101
8	000001011	$A > 0 : \text{shl}(AQ)$	00010100	0000101
		$A = A - M$	00001001	0000101
		$A > 0, Q_0=1$	00001001	00001011

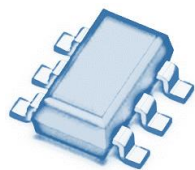
$$Q = 00001011 = (11)_{10}$$

$$R = 00001001 = (9)_{10}$$

(۳) به کمک یک جمع کننده ۴ بیتی، مدار یک تقسیم کننده ۱۶ بیتی را طراحی کنید.

ابتدا ماژولی طراحی می‌کنیم که یک عدد ۱۶ بیتی را گرفته و آن را تا جایی شیفت به چپ می‌دهد که MSB آن برابر ۱ شود. اینکار به کمک یک شیفت رجیستر و یک ماشین حالت امکان پذیر است. نام این وسیله را MSBFinder می‌نامیم. همانطور که در مدار مشاهده می‌کنید، مقسوم توسط MUX وارد یک شیفت رجیستر شده و مقسوم علیه نیز بعد از عبور از MSBFinder وارد یک شیفت رجیستر دیگر می‌شود. وظیفه MSBFinder این است که به تعدادی صفر سمت راست مقسوم علیه بگذارد تا تعداد ارقام آن با مقسوم برابر شود. این تعداد در رجیستر ShiftCounter نگهداری می‌شود. در هر مرحله مقسوم علیه از مقسوم کم شده و حاصل در رجیستر Remainder قرار می‌گیرد و هر بار از ShiftCounter کاسته می‌شود. این کار تا زمانی انجام می‌شود که ShiftCounter صفر شده باشد. هر بار که مقسوم از مقسوم علیه کم می‌شود، اگر مقدار ShiftCounter مثبت بود، به متغیر Result یک بیت ۱ وارد می‌شود و در نهایت جواب تقسیم در متغیر Result خواهد بود و باقیمانده آن نیز در Remainder قابل دسترسی است.





۴) اگر فرض کنیم تأخیر هر گیت and به ازای هر تغییر سیگنال، ۱۰ پیکوثانیه باشد و تأخیر هر جمع کننده ۲۰ پیکوثانیه باشد، تأخیر مدار یک ضرب کننده ۴ بیتی به روش شیفت و جمع به ازای هر کدام از عمل‌های ضرب زیر را حساب کنید.

a) 12×3

اگر فرض کنیم سیگنال تمام گیت‌های AND در اولین جمع تغییر کرده اند (برای مثال قبل از این جمع مقدار 'Z' داشته اند) پس برای تمام آنها زمان ۱۰ پیکوثانیه را می‌توان در نظر گرفت. چون نتایج آنها مستقل از هم است و به صورت موازی بدست می‌آید. بعد از آن زمان تأخیر جمع کننده‌ها باید محاسبه شود. اما از آنجایی که نتیجه هر جمع کننده به جمع کننده‌های قبلی خود وابسته است، تأخیرها به صورت مرحله به مرحله محاسبه می‌شود.

$$12_{10} = 1100_2$$

$$3_{10} = 0011_2$$

					۱	۱	۰	۰
					۰	۰	۱	۱
					۱	۱	۰	۰
				۱	۱	۰	۰	
				
			
.	.	.	۱	.	.	۱	.	.

پس تأخیر در مرحله جمع کننده‌ها نیز به تعداد آنها می‌شود که در نتیجه داریم:

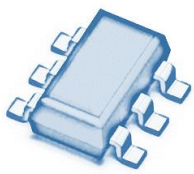
$$T = 10 \text{ ps} + 3 \times 20 \text{ ps} = 70 \text{ ps}$$

b) 15×9

$$9_{10} = 1001_2$$

$$5_{10} = 0101_2$$

					۱	۰	۰	۱
					۰	۱	۰	۱
					۱	۰	۰	۱
				
			۱	.	.	۱		
			
.	.	.	۱	.	۱	۱	.	۱



همانطور که در اینجا مشاهده می‌شود، بیتی که حاصل اولین AND است، نسبت به ضرب قبل تغییر کرده؛ پس باز هم باید به صورت ترتیبی زمان تأخیر را حساب کنیم و حاصل مانند قسمت قبل بدست می‌آید.

$$T = 10\text{ps} + 3 \times 20\text{ps} = 70\text{ps}$$

c) 15×15

$$15_{10} = 1111_2$$

					۱	۱	۱	۱
					۱	۱	۱	۱
				۱	۱	۱	۱	
		۱	۱	۱	۱	۱		
					۱	۱	۱	
۱	۱	۱	۰	۰	۰	۰	۰	۱

$$T = 10\text{ps} + 3 \times 20\text{ps} = 70\text{ps}$$

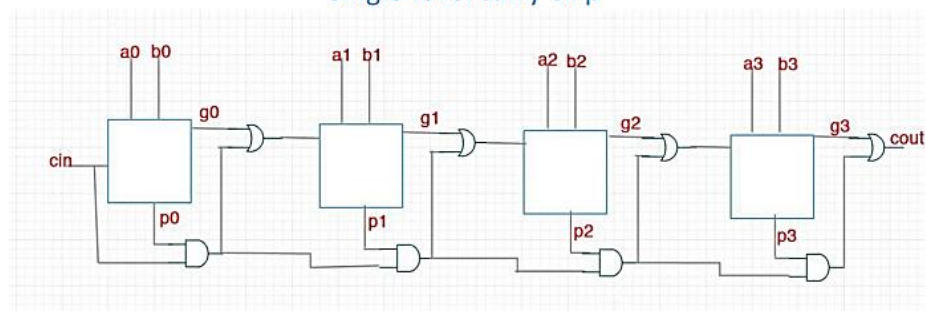
(۵) جمع کننده بهینه‌ای طراحی کنید که :

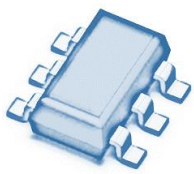
Single-level carry-skip (الف)

Two-level carry-skip (ب)

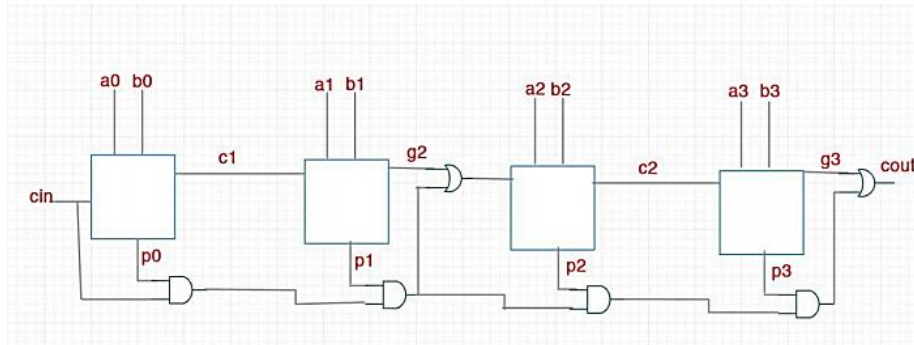
قسمت الف تنها با استفاده از p و carry قبلی قسمتی از g را پیش محاسبه می‌کرد که چون با بقیه اعضا در g or می‌شود اگر ۱ شود carry بعدی هم ۱ شده و نیاز برای انتظار برای محاسبه ی g نیست. در قسمت ب باید دو بیت دو بیت carry را با استفاده از p and های قبلی و cin (مانند شکل) پیش محاسبه کنیم و ببینیم نیاز برای انتظار برای g هست یا جداگانه محاسبه می‌شود.

Single-level carry-skip

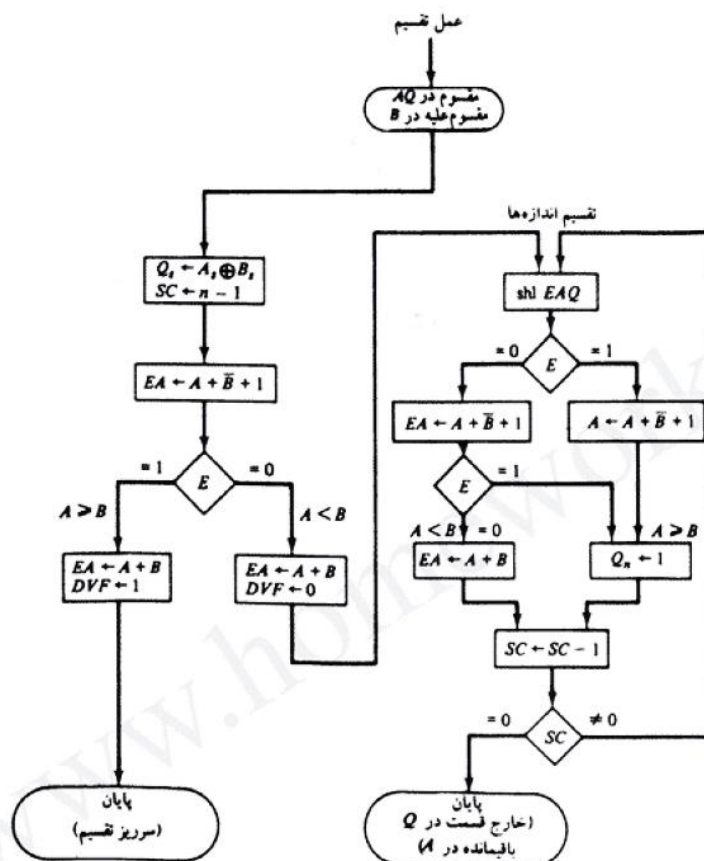




Two-level carry-skip



۶) فلوچارت عمل تقسیم به روش بازیافتی (Restoring) را با در نظر گرفتن سرریز در تقسیم رسم کرده و سپس تقسیم ۷ بر ۳ را با استفاده از این روش انجام دهید. (اعداد را چهار بیتی و به صورت بی علامت در نظر گرفته و تمام مراحل را به طور دقیق بنویسید)





استاد درس : دکتر فربه

معماری کامپیوتر

پاسخ تکلیف شماره ۴



دانشکده مهندسی
کامپیوتر و فناوری اطلاعات

موفق باشید

B=0011			
Comment	A	Q	SC
	0000	0111	4
Shl	0000	1110	
A=A-B	1101		

	1101		
A=A+B	0011		

	0000	1110	3
Shl	0001	1100	
A=A-B	1101		

	1110		
A=A+B	0011		

	0001	1100	2
Shl	0011	1000	
A=A-B	1101		

	0000	1001	1
Shl	0001	0010	
A=A-B	1101		

	1110		
A=A+B	0011		

	0001	0010	0
	Reminder	Quotient	