دانشکده مهندسی کامپیوتر

معماری کامپیوتر

نیم‌سال اول ۱۴۰۲-۱۴۰۱

حل تمرین پنج — **آخرین مهلت تحویل ساعت ۲۴ روز دوشنبه ۲۶ دی**

مهلت تحویل امتیازی ساعت ۲۴ روز دوشنبه ۵ دی

به موارد زیر توجه کنید:

۱- حتما نام و شماره دانشجویی خود را روی پاسخ‌نامه بنویسید.

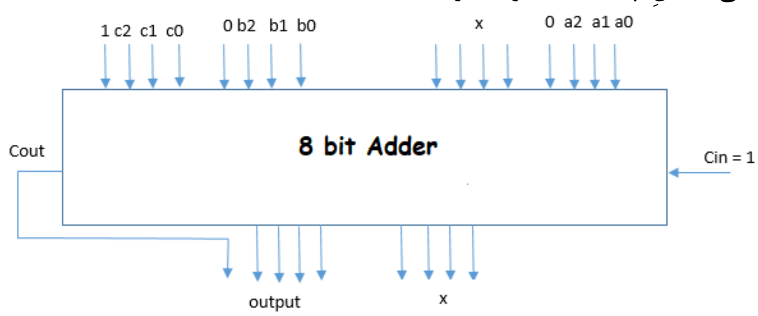
۲- کل پاسخ تمرینات را در قالب یک فایل pdf با شماره دانشجویی خود نام‌گذاری کرده در سامانه CW بارگذاری کنید.

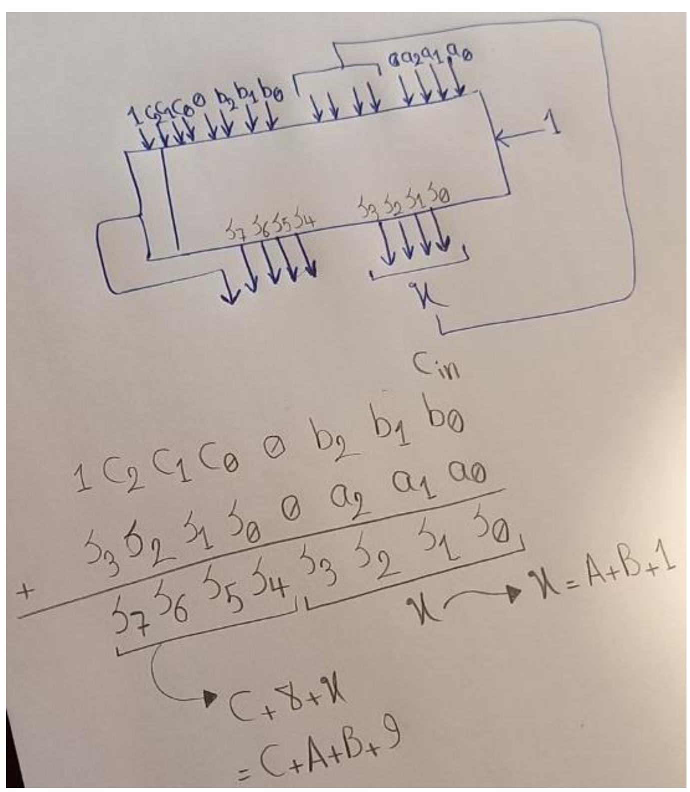
۳- این تمرین ۱۰۰ نمره دارد که معادل ۰,۵ نمره از نمره کلی درس است.

۴- تحویل این تمرین در تاریخ «تحویل امتیازی» ۰,۱ نمرهٔ اضافه دارد.

۵- در صورت مشاهده هر گونه مشابهت نامتعارف هر دو (یا چند) نفر کل نمره این تمرین را از دست خواهند داد.

۱- (۱۰ نمره) فرض کنید سه عدد سه بیتی $A = A_2A_1A_0$ و $B = B_2B_1B_0$ و $C = C_2C_1C_0$ را مطابق شکلِ زیر به یک جمع‌کنندهٔ ۸ بیتی می‌دهیم. نتیجهٔ نهایی معادلِ چه عددی خواهد بود؟

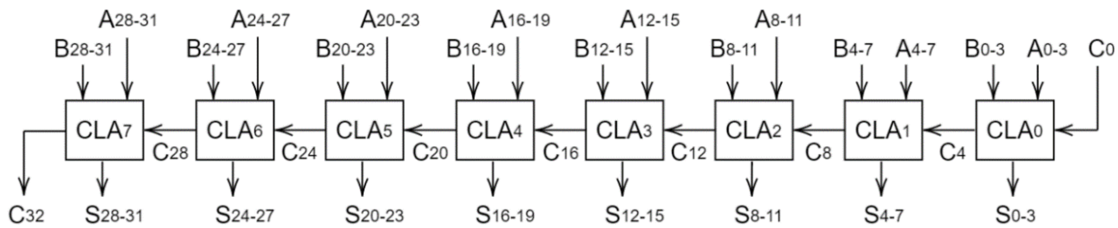




۲- (۴۰ نمره) می‌خواهیم جمع دو عدد ۳۲ بیتی را با کنار هم قراردادنِ پیمانه‌های جمع‌کنندۀ ۴ بیتی با پیش‌بینی رقم نقلی (CLA) انجام دهیم. (منظور این است که بیتِ نقلیِ خروجیِ هر پیمانۀ ۴ بیتی به بیتِ نقلیِ ورودیِ پیمانۀ بعدی وارد شود.)

الف- شکل جمع‌کننده را رسم کنید و تاخیر آن‌را برحسب تاخیر گیت‌ها حساب کنید. فرض کنید تاخیر گیت‌های پایه (and-or-not) یکسان و برابر باشد و نیز فرض کنید xor را با استفاده از گیت‌های پایه ساخته‌ایم.

پاسخ:



$p_i = a_i \oplus b_i,\quad g_i = a_i b_i,\quad C_4 = g_3 + p_3 g_2 + p_3 p_2 g_1 + p_3 p_2 p_1 g_0 + p_3 p_2 p_1 p_0 C_0$

$t_{g_i} = \delta$

$t_{p_i} = 3\delta$

در لحظه $t = 3\delta$ همه p_i و g_i ها حاضر می‌شوند. بنابراین با در نظر گرفتن زمان لازم برای and کردن p_i و g_i ها، تاخیر بیت‌های نقلی خروجی به صورت زیر است:

$t_{c_4} = 3\delta + 2\delta = 5\delta$

$t_{c_8} = 5\delta + 2\delta = 7\delta$

...

$t_{c_{28}} = 5\delta + 6(2\delta) = 17\delta$

$t_{c_{32}} = 5\delta + 7(2\delta) = 19\delta = t_{c_{31}}$

$t_{S_{28-31}} = t_{c_{31}} + t_{xor} = 19\delta + 3\delta = 22\delta$

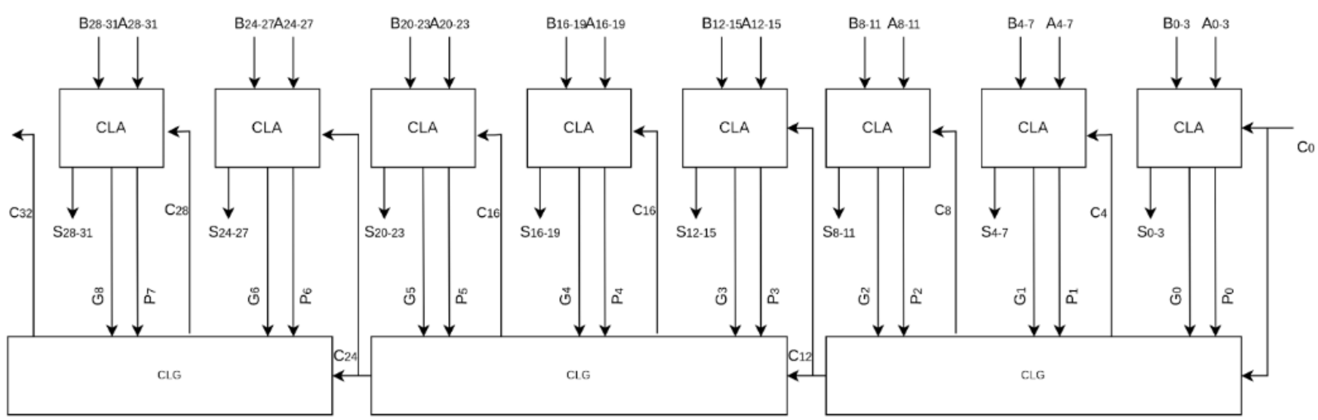
ب- زمان به‌دست آمده را با حالتی که از جمع‌کنندۀ ripple carry استفاده کنیم مقایسه کنید.

پاسخ: در استفاده از ripple carry adder نیاز به ۳۲ تمام‌افزا خواهیم داشت که تاخیر محاسبه در یک جمع‌کننده برابر 2δ است و از کنار هم قرار دادن ۳۲ تا از آنها، خواهیم داشت:

$t_{C_{32}} = 32(2\delta) = 64\delta$

$t_{S_{31}} = t_{c_{31}} + t_{xor} = 31(2\delta) + 3\delta = 65\delta$

ج- این بار فرض کنید بیت‌های نقلی را مطابق شکل زیر در دو لایه تولید کنیم. تاخیر آخرین بیت نقلی را حساب کنید.



$t_{p_i} = 3\delta,\quad t_{g_i} = \delta,\quad P_2 = p_2.p_1.p_0,\quad G_2 = g_2 + p_2.g_1 + p_2.p_1.g_0 \quad \Longrightarrow \quad t_{P_2} = 4\delta,\quad t_{G_2} = 5\delta \quad \Longrightarrow \quad t_{C_{12}} = 7\delta$

$t_{c_{12}} = 7\delta \quad \Longrightarrow \quad t_{C_{24}} = 9\delta \quad \Longrightarrow \quad t_{C_{32}} = 11\delta,\quad t_{S_{31}} = t_{C_{24}} + t_{XOR} = 12\delta$

۳- (۱۰ نمره) ثابت کنید حاصل‌ضرب یک عدد n رقمی در یک عدد m رقمی در مبنای r حداکثر n+m رقم دارد.

پاسخ: حداکثر مقدار قابل نمایش n رقمی در مبنای r برابر $r^n - 1$ است بنابراین حداکثر حاصل‌ضرب به صورت زیر است:

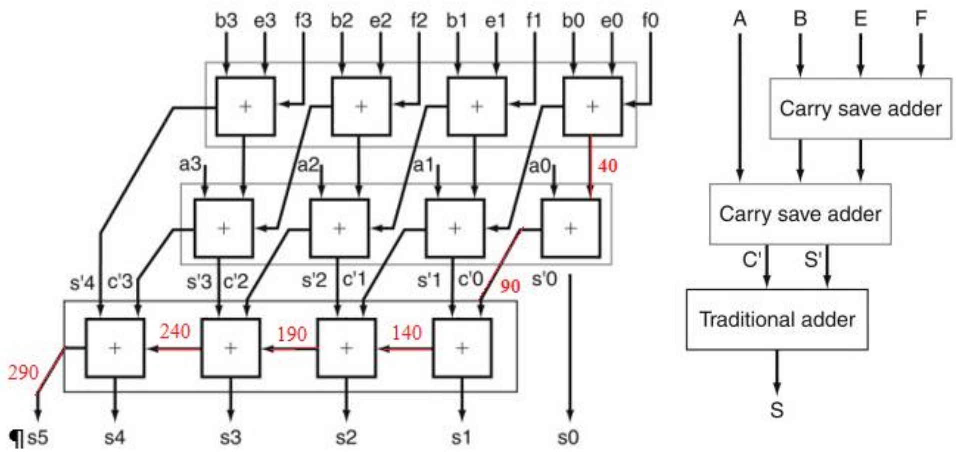
$$(r^n - 1)(r^n - 1) = r^{2n} - 2r^n + 1 \le r^{2n} - 1$$

که یعنی نتیجه در حداکثر ۲n بیت قابل نمایش است.

این موضوع را با استفاده از لگاریتم در مبنای r هم می‌توانستیم نشان دهیم. دو عدد b_r و a_r را که هر کدام حداکثر n رقم دارند در هم ضرب می‌کنیم. تعداد بیت حاصلضرب عبارت خواهد بود از:

$$log_r(a_r * b_r) = log_r(a_r) + log_r(b_r) \le 2n$$

۴- (۱۰ نمره) می‌خواهیم ضرب دو عدد ۴ بیتی را به کمک مدار carry save adder و مطابق با شکلِ زیر (اسلاید ۳۵) انجام دهیم. اگر رقم نقلی در هر جمع‌کننده پس از ۵۰ نانوثانیه و بیت حاصل‌جمع در ۴۰ نانوثانیه تولید شود، مدت زمان لازم برای محاسبهٔ نتیجهٔ نهایی را به دست آورید.



پاسخ: مسیر بحرانی و تاخیر نهایی روی شکل دیده می‌شود.

۵- (۱۰ نمره) نشان دهید شرطِ لازم و کافی برای سرریز شدنِ تقسیمِ دو عددِ صحیحِ بدونِ علامت A و B در خارج قسمتِ n بیتی، این است که $A \ge 2^n B$

پاسخ: اگر خارج قسمت بیش از n بیت داشته باشد سرریز رخ می‌دهد. فرض کنیم سر ریز رخ نداده؛ از رابطه تقسیم کمک می‌گیریم:

$$A = BQ + r : r \le B - 1$$
$$Q < 2^n \Rightarrow Q \le 2^n - 1 \Rightarrow A \le B(2^n - 1) + B - 1 \Rightarrow A \le 2^n B - 1 \Rightarrow A < 2^n B$$

بنابراین اگر سرریز رخ ندهد، در بدترین حالت (وقتی Q و r بیشینه باشند و سرریز رخ ندهد) $A = 2^n B - 1$ و در حالت‌هایی که Q بزرگ‌تر باشد، سرریز رخ می‌دهد.

گزارهٔ «اگر سرریز رخ ندهد، آنگاه $A < 2^n B$ » معادل است با گزارهٔ «اگر $A \ge 2^n B$، آنگاه سرریز رخ می‌دهد»، پس یک طرف قضیه ثابت شد.

حالا برای نشان دادن اینکه اگر سرریز رخ داده، حتما $A \ge 2^n B$، از برهان خلف استفاده می‌کنیم. فرض می‌کنیم سرریز رخ نداده داریم:

$$Q \le 2^n - 1 \Rightarrow A = (2^n - 1)B + r \ge 2^n B \Rightarrow r \ge B$$

که تناقض است. بنابراین حکم درست است.

۶- (۱۰ نمره) فرض کنید قالب نمایش اعداد ممیز شناور ۳۲ بیتی به صورت زیر باشد.

Sign (1bit)	E for Exponent (e bits)	F for Fraction (f bits)

$(-1)^{Sign} \times (1.F) \times 2^{(E-bias)}$

اگر عدد $-53\frac{7}{9}$ در این نمایش به صورت $(CB5C71C7)_{16}$ نمایش داده شود، اندازه هر فیلد و مقدار bias را در این نمایش مشخص کنید.

پاسخ: ابتدا عدد داده شده را بدون در نظر گرفتن علامت به باینری تبدیل می‌کنیم و آنرا نرمال می‌کنیم:

$53 = (110101)_2 \quad \frac{7}{9} = (0.\overline{110001})_2$

$53\frac{7}{9} = (110101.11000111000111\ldots)_2 = (1.10101110001110001110000\ldots)_2 \times 2^5$

در نتیجه بخش کسری به صورت ..1010111000111000111000 خواهد بود که با تطبیق دادن با نماش داده شده می‌بینیم:

$CB5C71C7_{hex} = (1100101"1010111000111000111000111")_2$

نتیجه می‌گیریم که ۲۵ بیت برای نمایش بخش کسری است. یک بیت هم طبق دادهٔ سوال برای علامت است و درنتیجه ۶ بیت برای نمایش نما داریم. همچنین ۱۰۰۱۰۱ باید نمایش biased برای عدد ۵ باشد، بنابراین:

$(100101)_2 = 37 - bias = 5 => bias = 32$