



دانشگاه تهران

دانشکده ریاضی، آمار و علوم کامپیوتر

نیم سال دوم تحصیلی سال ۱۴۰۱-۱۴۰۰

مطالب تکمیلی اصول سیستم‌های کامپیوتری

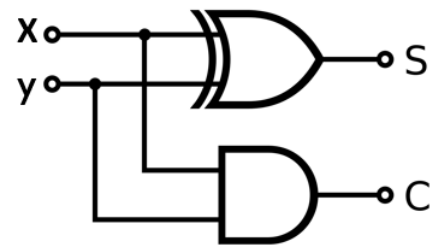
مطالب فصل چهارم مدار منطقی:

- **Half – Adder**: دو بیت را با هم جمع می‌کند بدین صورت که حاصل را در S و مقدار $Carry$ را در C قرار می‌دهد.

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$S = x'y + xy' = x \oplus y$$

$$C = xy$$

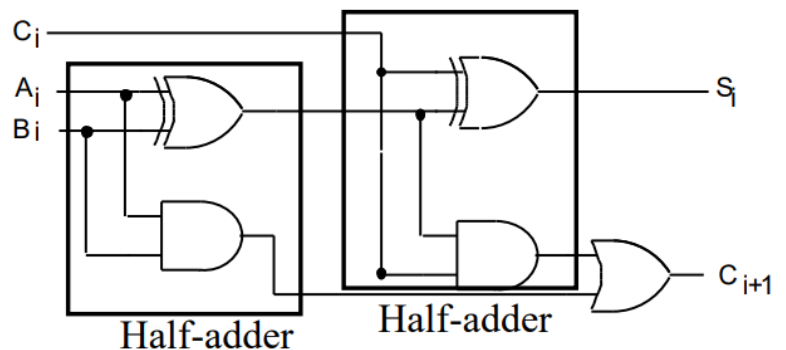


- **Full – Adder**: سه بیت را با هم جمع می‌کند بدین صورت که حاصل را در S و مقدار $Carry$ را در C قرار می‌دهد.

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$$S = x'y'z + x'yz' + xy'z' + xyz = (x \oplus y) \oplus z$$

$$C = xy + xz + yz$$



- **Half - sub**: مقدار $x - y$ را محاسبه می‌کند. بدین صورت که در D حاصل تفریق و در B قرض گرفتن را نگه می‌دارد. اگر $x - y$

نامنفی باشد، مقدار B برابر با ۰ و اگر $x - y$ منفی باشد، مقدار B برابر با ۱ است.

x	y	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

$$D = x'y + xy' = x \oplus y$$

$$B = x'y$$

- **Full - sub**: مقدار $x - y - z$ را محاسبه می‌کند. بدین صورت که در D حاصل تفریق و در B قرض گرفتن را نگه می‌دارد.

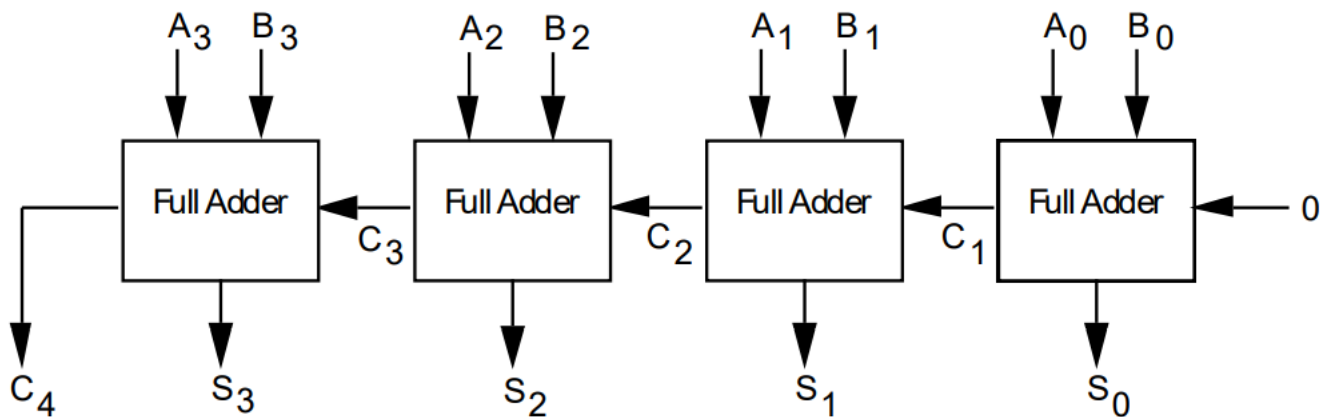
اگر $(x - y) - z$ یا $(x - y)$ نامنفی باشد، مقدار B برابر با ۰ و اگر $(x - y) - z$ یا $(x - y)$ منفی باشد، مقدار B برابر با ۱ است.

x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$D = x'y'z + x'yz' + xy'z' + xyz$$

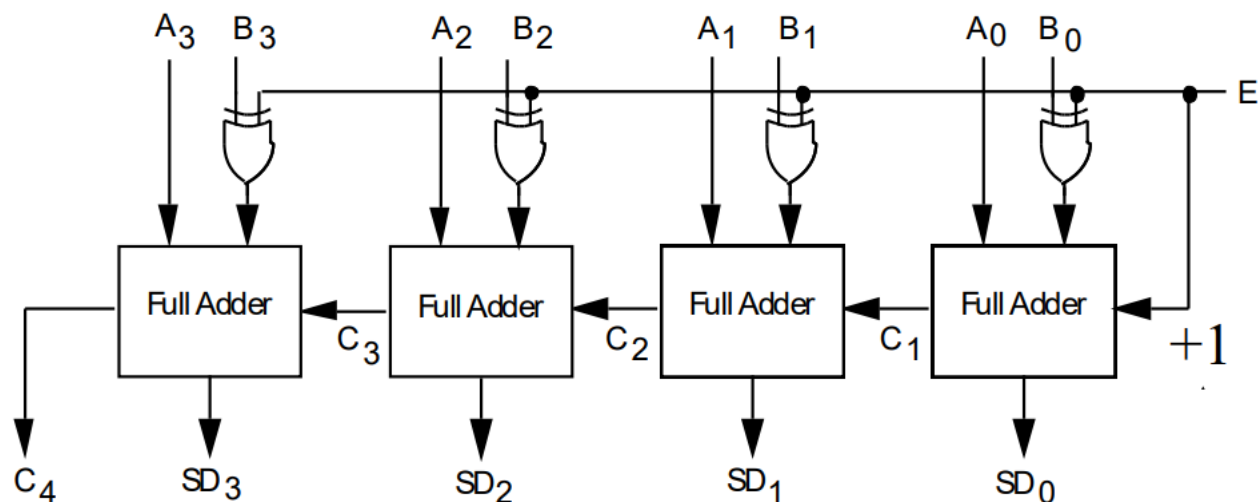
$$B = x'y + x'z + yz$$

- **4bit - Adder**: دو عدد چهار بیتی را با هم جمع می‌کند.



مقدار $carry$ در مرحله اول برابر ۰ است.

- **4bit – adder – subtractor**: در حالتی که $E = 1$ است، عدد ۴ بیتی B را از عدد ۴ بیتی A کم می‌کند. برای این کار، عدد B را 1 's complement می‌کند و سپس حاصل را با 1 جمع می‌کند. در اصل، A را با 2 's complement عدد B جمع می‌کند. در حالتی که $E = 0$ است، دو عدد را با هم جمع می‌کند.



- **overflow**: برای فهمیدن اینکه چه زمانی **overflow** رخ می‌دهد، کافیست **carry** دو مرحله آخر جمع را با یکدیگر مقایسه کنیم، اگر دو **carry** یکسان نباشند، **overflow** رخ می‌دهد. مثال:

00	01	11	10	00	11
0010	0011	1110	1101	0010	1110
0011	0110	1101	1010	1100	0100
<hr/>					
0101	1001	1011	0111	1110	0010
2	3	-2	-3	2	-2
3	6	-3	-6	-4	4
5	-7	-5	7	-2	2
	OFL		OFL		

همانطور که در تصویر مشخص است، در جمع‌هایی که دو **carry** مرحله آخر متفاوت است، **overflow** رخ داده است.