دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

تفریق کننده‌ها

طراحی واحد منطق و حساب

Arithmetic logic unit (ALU) design

نوع نمایش: بی‌علامت عمل محاسباتی: تفریق

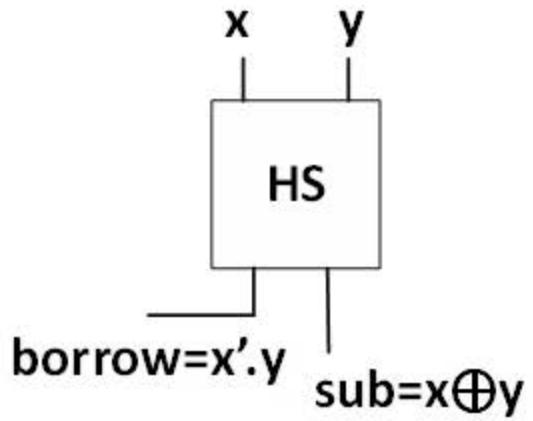
- تفریق ۲ عدد **n**-بیتی:
  - تفریق‌کننده آبشاری **(Ripple subtractor)**
  - تفریق‌کننده مکمل‌گیر **(Complement subtractor)**

تفریق کننده اعداد بی‌علامت

نوع نمایش: بی‌علامت عمل محاسباتی: تفریق

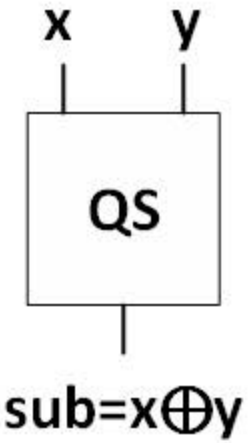
- **n=1** (تفریق دو عدد بی‌علامت تک بیتی)
  - ربع تفریق‌کننده **(Quarter subtractor)**
  - نیم تفریق‌کننده **(Half subtractor)**
  - تمام تفریق‌کننده **(Full subtractor)**

x	y	(تفریق ریاضی)	
		borrow	sub
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0



delay (sub) = d
delay (borrow) = 2d
Cost = 3 g

x	y	s=x-y (تفریق ریاضی)
0	0	0
0	1	1
1	0	1
1	1	0

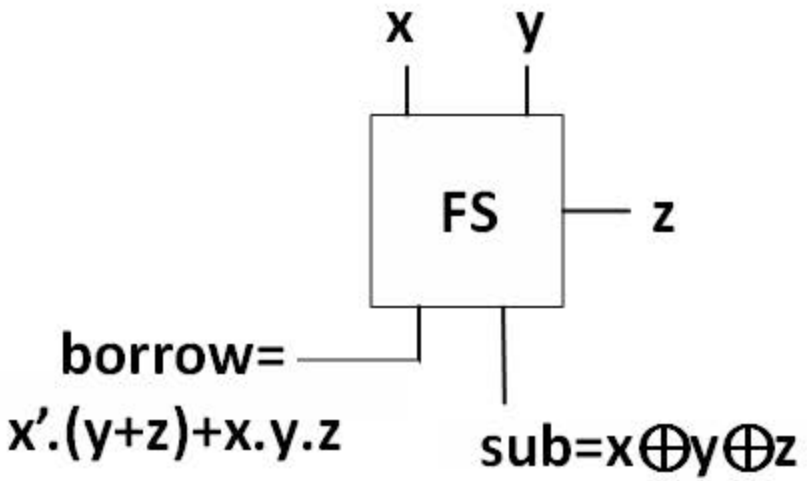


delay (sub) = d
Cost = 1 g

نوع نمایش: بی‌علامت عمل محاسباتی: تفریق

- **n=1** (تفریق دو عدد بی‌علامت تک بیتی)
  - تمام تفریق‌کننده **(Full subtractor)**

x	y	z	x-y-z (تفریق ریاضی)	
			borrow	sub
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

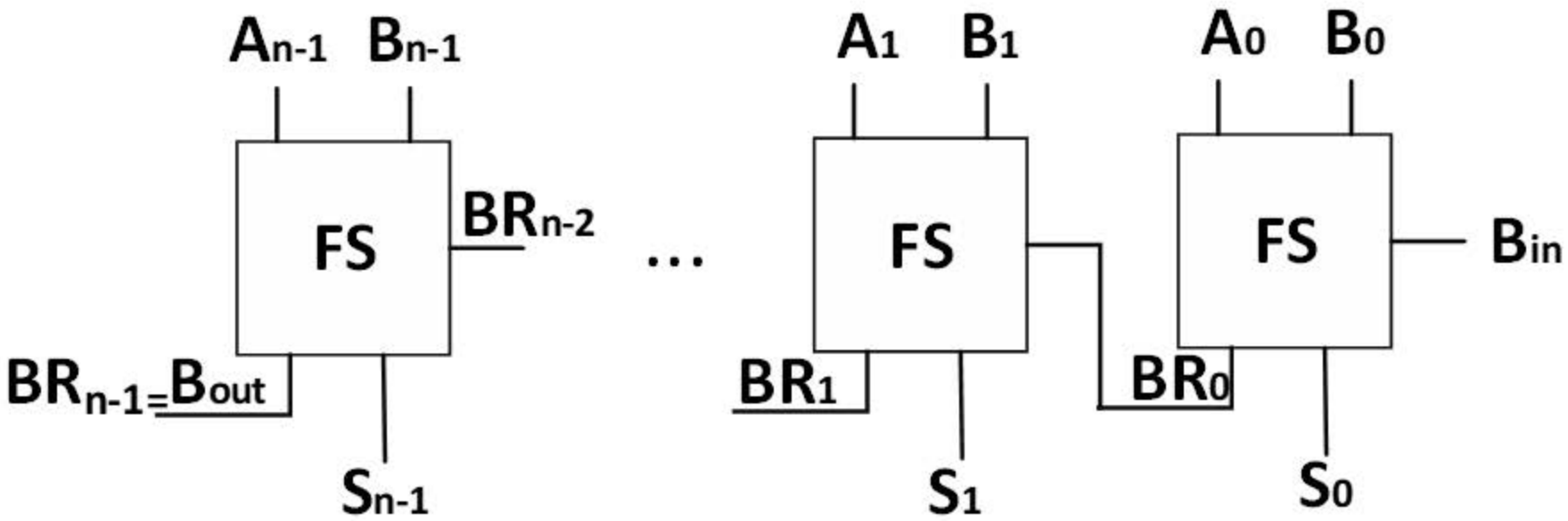


delay (sub) = d
delay (borrow) = 3d
cost = 6 g

FS delay = 3d

تفریق کننده آبشاری (Ripple subtractor)

تفریق‌کننده آبشاری (Ripple subtractor)



delay (subtract) = (3n-2)d
delay (borrow) = 3nd

cost = 6n g

تفریق کننده مکمل گیر (Complement subtractor)

تفریق کننده مکمل گیر

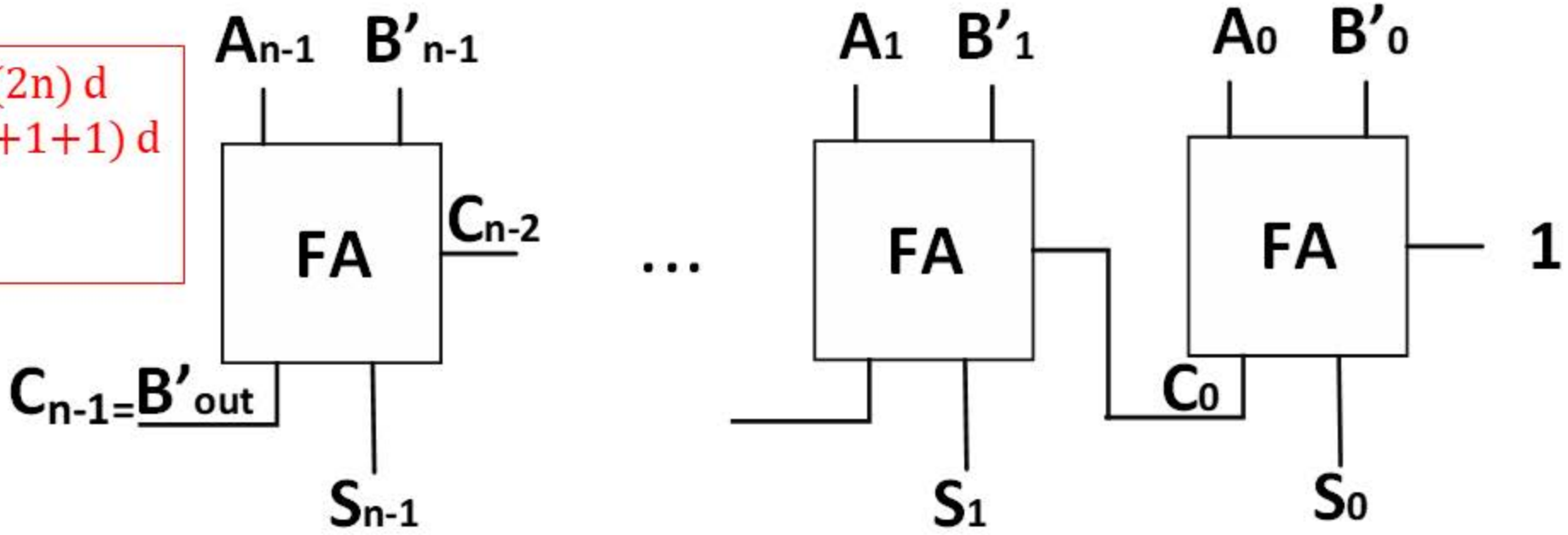
- برای محاسبه **A – B** در فضای **n** بیتی بصورت زیر عمل می‌کنیم:

$$A - B = (\text{in n bit}) = 2^n + A - B$$
$$= A + (2^n - B)$$
$$= (\text{provable}) = A + (B' + 1)$$

تفریق کننده مکمل گیر

delay (subtract) = (2n) d
delay (borrow) = (2n+1+1) d

cost = 6n g



در این روش، رقم نقلی تولید شده، برعکس رقم قرضی است. چرا؟

Bout = NOT (Cn-1)

سرریز یا Overflow

- در محاسبات **ALU** لازم است نتیجه محاسبات در **n** بیت قابل نمایش باشد (چرا؟)

- در محاسبات، گاهی اوقات شرایطی پیش می‌آید که پاسخ محاسبه دو عدد **n**-بیتی در فضای **n**-بیتی نادرست است (پاسخ غلط). برای اطلاع به کاربر، سیگنالی به نام **Overflow** (سرریز) در نظر گرفته می‌شود که از این طریق، اطلاع داده می‌شود. لذا قبل از برداشت پاسخ از مدار، لازم است ابتدا بیت سرریز چک شود که آیا یک است یا صفر.
  - اگر صفر باشد، پاسخ در **n** بیت، درست است و ارسال به مرحله بعدی است.
  - اگر یک باشد، پاسخ نادرست است و امکان انجام وجود ندارد. یعنی سخت افزار قادر به پاسخ دادن نیست (راه حل در این شرایط چیست؟)

شرایط سرریز شدن (پاسخ غلط) در محاسبات جمع و تفریق، بی‌علامت

- هنگام جمع دو عدد بی‌علامت
  - ایجاد رقم نقلی، به معنای سرریز شدن نتیجه است. (زیرا نتیجه **n+1** رقمی است و قابل نمایش در **n** بیت نیست)

- هنگام تفریق دو عدد بی‌علامت
  - ایجاد رقم قرضی، به معنای سرریز شدن نتیجه است. (زیرا نتیجه منفی است و چون قابل نمایش نیست بیت قرضی ایجاد شده است.)
    - توجه: چناچه از مدار تفریق‌گر مکمل‌گیر استفاده شود، نبود بیت نقلی، به معنای وجود رقم قرضی است. پس نبود بیت نقلی یعنی سرریز شدن نتیجه تفریق.

- مثال: مشخص کنید در فضای ۴-بیتی، کدام سرریز می‌شود و کدام سرریز نمی‌شود؟

0101 **0011 +** **----------**	**1011** **1101 +** **----------**	**0101** **0011 -** **----------**	**0011** **0101 -** **----------**

سوال؟