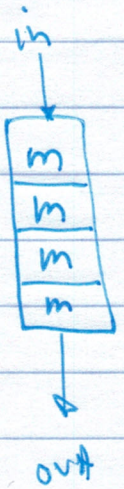
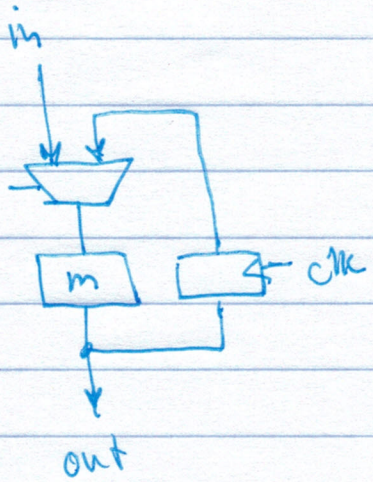
EEC 480

April 20, 2021

Adders & Subtractors



"full view"

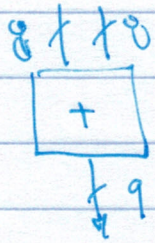


"iterative"

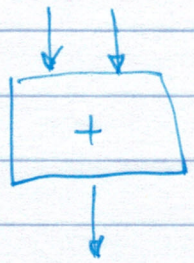
Adders

- 2-input adder
- both inputs same width

→ width of sum = width of inputs + 1

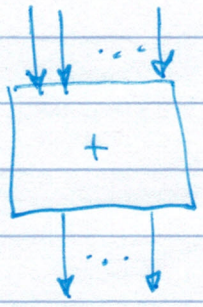


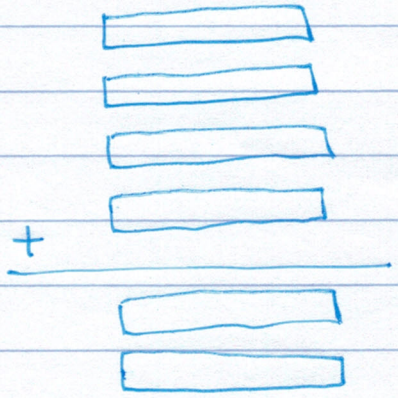
1) Carry-Propagate Adders (CPA)
   - output is a single word
   - carry must propagate
   - normal addition



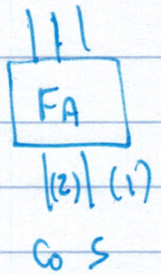
2) Carry-Save Adders (CSA)

- output is not one word
- " is 2+ words
- input is 3+ words





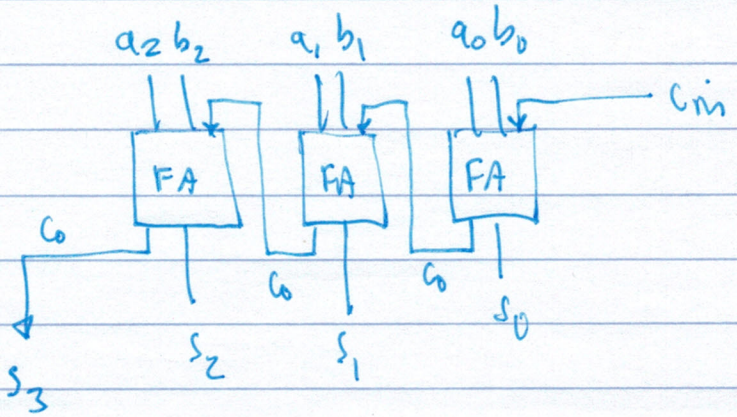
Full Adder




```
  a
  b
+ c
-----
Co S
```

1) Ripple-Carry Adder

- simplest CPA
- smallest CPA
- slowest CPA

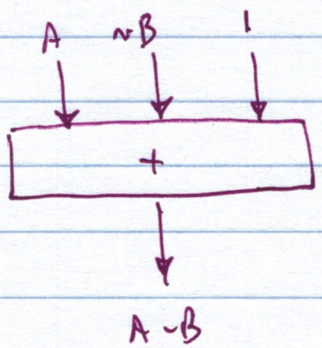


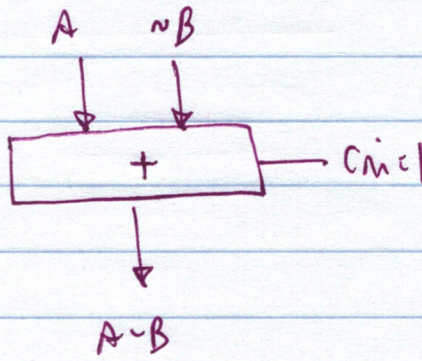
Subtractors

- requires a signed format Ex: 1 - 1 = -1
- we will use 2's complement
- Use a 2's compl. adder

$$A - B = A + (-B)$$

$$= A + (\sim B) + 1$$



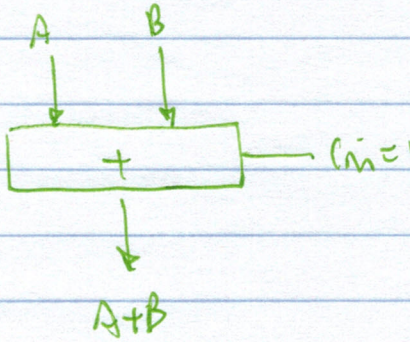


add/sub | B

A	X	Y	XOR
0	0	0	0
0	0	1	1
1	1	0	1
1	1	1	0

$B = B \text{ XOR } (0)$ ← X

$\bar{B} = B \text{ XOR } (1)$



$\sim B = (B) \text{ XOR } (\overline{Add/Sub})$

Sign extension

- needed for 2's complement addition

Ex: $-5 + 18$

$-8\ 4\ 2\ 1$

$-32\ 16\ 8\ 4\ 2\ 1$

$$\begin{array}{rr} 0\,0\,1\,0\,1\,1 & = -5 \\ +\ 0\,1\,0\,0\,1\,0 & = +18 \\ \hline 0\,1\,1\,1\,0\,1 & = +29 \end{array}$$

Rule #1: input + output word widths are the same!

input #1: $8 + 2 + 1 = 11$

$$\begin{array}{r} 18 \\ \hline 29 \end{array}$$

Rule #2: the value of a 2's complement # will never change due to sign extension

$1\,0\,1\,1 = -8 + 2 + 1 = -5$

$1\,1\,0\,1\,1 = -16 + 8 + 2 + 1 = -5$

$1\,1\,1\,0\,1\,1 = -32 + 16 + 8 + 2 + 1 = -5$

Procedure:

1) Calc min width of sum

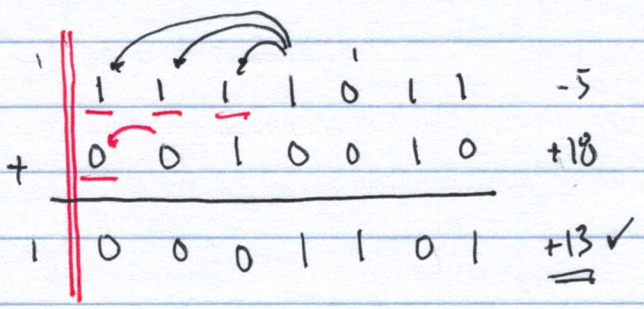
2) Sign extend inputs to width as answer (sum)

Ex. a 8 bits

{a[7], a[7], a}

3) Add as usual

4) Ignore bits that ripple to the left of answer's MSB



[-8, +7] 4-bit
[-32, +31] 6-bit
[-40, +38]

↓

7 bits

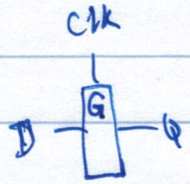
Single-Bit Memories

Memory

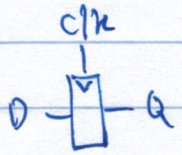
1) Single-Bit - have data input + data output

a) Clock-less latch

b) Transparent level-sensitive latches



* c) Edge-triggered FFs



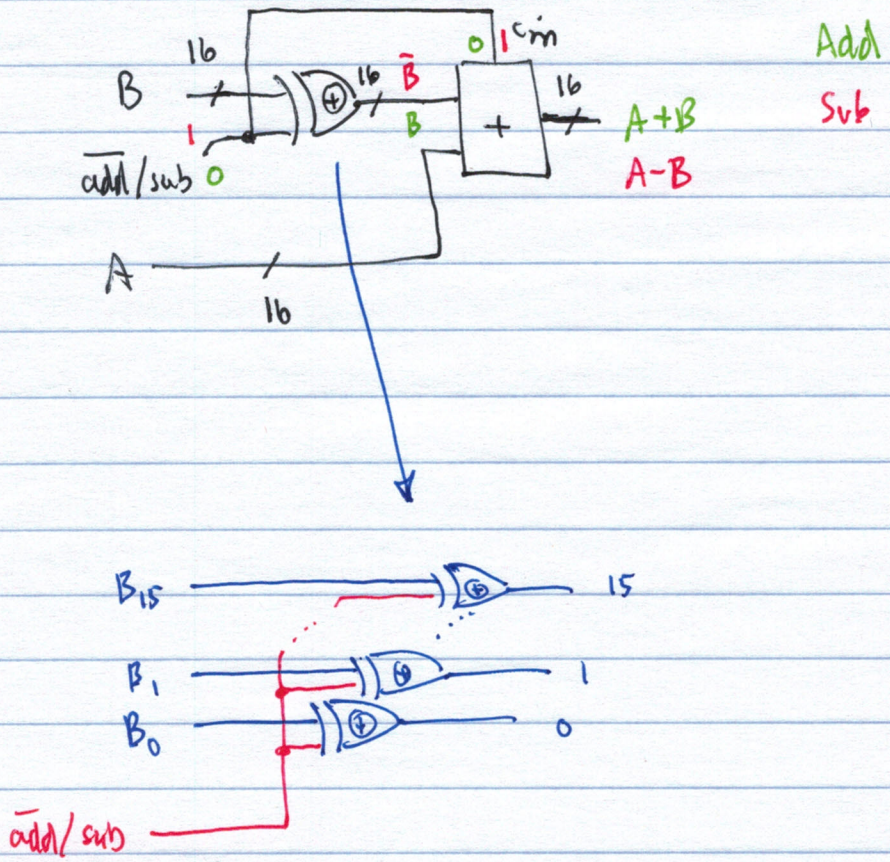
2) Array - have an address

a) SRAM

b) DRAM

c) Flash

d) etc...


16
B
16
$\bar{B}$
B
1
add/sub
0
0
1
C_{in}
16
A+B
A-B
A
16
Add
Sub
B_{15}
15
B_1
1
B_0
0
add/sub