integer whole number

signed number

$(70)_{10} \rightarrow (\ ?\ )_2$

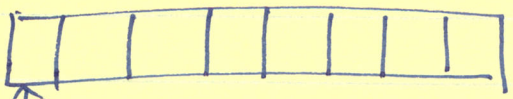
Remainders

2 | 70		
2 | 35	–	0
2 | 17	–	1
2 | 8	–	1
2 | 4	–	0
2 | 2	–	0
1	–	0

(70)

$(1000110)_2$

+70 −70



+70 0 1 0 0 0 1 1 0 8bits ⇒ 1byte

sign binary digit

Sign magnitude method

"value"

0 1 0 0 0 1 1 0 +70

1 1 0 0 0 1 1 0 −70

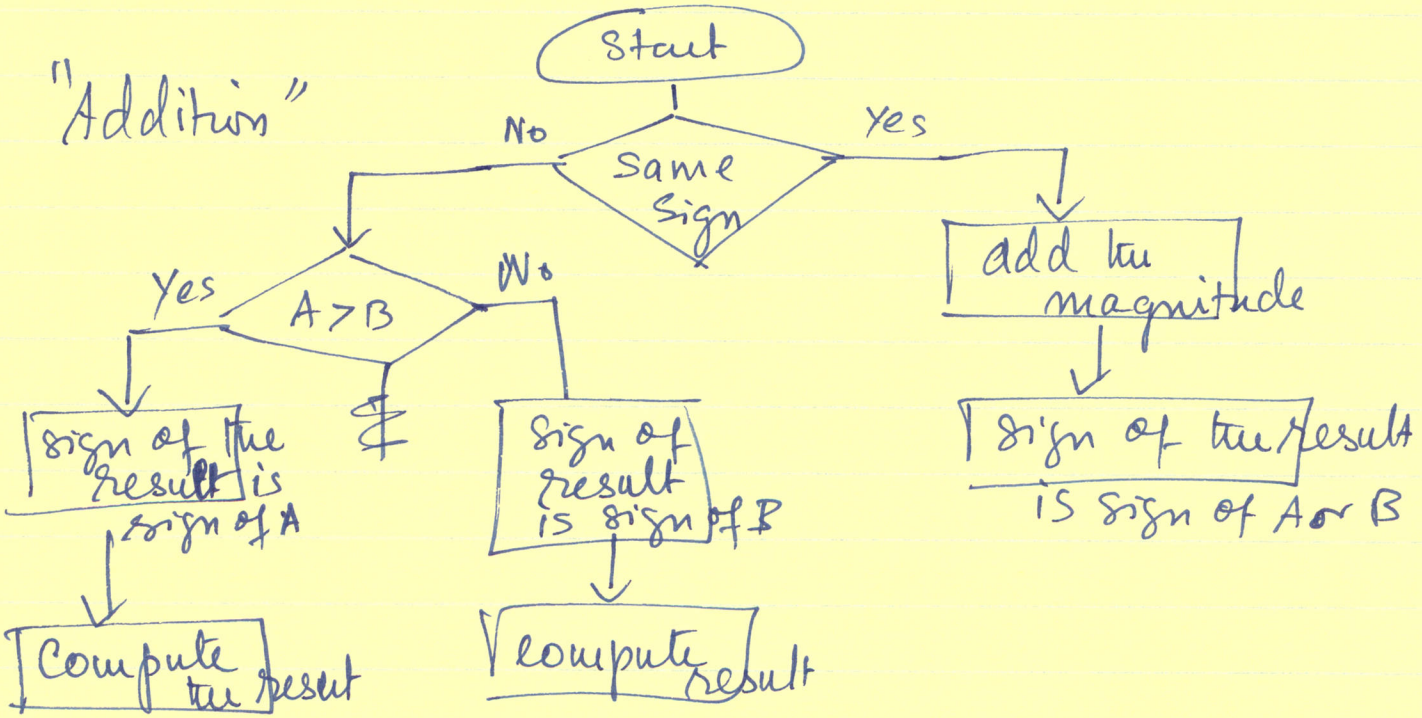
signed ~~arithmetic~~ arithmetic

Signed arithmetic

How am I going perform (i) addition (ii) subtraction with sign-magnitude representation ?

A + B A − B

"Addition"



"Subtraction" of signed integers that is another algorithm.

Is there an alternative way to represent signed numbers so that "arithmetic" (addition, Sub, mult, div) are simpler

what is that data representation?

positive # ⟹ Sign magnitude

negative # ⟹ 2's complement

radix = 2 radix complement

Example: $(+70)_{10}$

2 | 70
2 | 35 – 0
2 | 17 – 1
2 | 8 – 1
2 | 4 – 0
2 | 2 – 0
2 | 1 – 0
0

$(1000110)_2$

⇓

8 bit binary digit container

| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

+ −

1 1

{+, −}

0 1

Sign magnitude

MSB

| | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

? magnitude

Sign

1 negative 0 positive

↑ How Am I going to Compute ?

2's complement

1. convert decimal → binary ✓
2. find 1's complement
   → flip 1's → 0's
   → flip 0's → 1's
3. add 1

8 bit container −70

1	1	0	0	0	1	1	0

0 0 1 1 1 0 0 1. 1's complement

\+ 1

0 0 1 1 1 0 1 0 2's complement for −70

positive # sign mag

negative # 2's complement