Lesson 02: Arithmetic Operations—Addition and subtraction
Objective

- Understand sign extension of 2’s complement number
- Negation
- Addition
- Subtraction
Sign Extension
Sign Extension

- Used in order to equalize the number of bits in two operands for addition and subtraction.
- Sign extension of 8-bit integer in 2’s complement number representation becomes 16-bit number 2’s complement number representation by sign extension.
Sign Extension

- When m-bit number sign extends to get n-bit number then $b_{m-1}$ copies into extended places upto $b_{n-1}$.
- msb (b7) in an 8 bit number copies into b15, b14, b13, b12, b11, b10, b9 and b8 to get 16-bit sign extended number in 2’s complement representation.
Examples

- $01000011_b$ becomes $00000000100 0011_b$
- $1100 0011_b$ becomes $111111111100 0011_b$
Negation
Two's-complement

- Original value: $0b00001100$ (12)
- Negate each bit: $0b11110011$
- Add 1: $0b11110100$ (Two's-complement representation of $-12$)
Perform Two’s complement for negation

- The 8-bit representation of +12 is 0b00001100
- The 8-bit two's-complement representation of –12 is 0b11110100
- Add the 8-bit two's-complement representation of –12, and +12
  
  \[
  \begin{array}{c}
  0b00001100 \\
  + 0b11110100 \\
  \hline
  0b00000000
  \end{array}
  \]
Addition
Carry

carry out of
low bit during addition

\[ 1 \leftarrow \]

0b 1 0 0 1
0b 0 1 0 1
----------
0b 1 1 1 0
Example + 3 – 4 using in 4-bit two's-complement notation

• The 4-bit two's-complement representations of +3 and −4
  0b0011 and
  0b1100

Adding— 0b1111

Answer— Two's-complement representation of −1
Example $-3 - 4$ in 4-bit two's-complement notation

- 4-bit 2’s complement numbers can only be between $+7$ and $-8$
- To perform subtraction—Negate the second operand and add
- Thus, actual computation—perform $-3 + (-4)$
- The two's-complement representations of $-3$ and $-4$ are $0b1101$ and $0b1100$
Example $-3 - 4$ in 4-bit two's-complement notation

- Adding $0b1101$ and $0b1100$
- Get $0b11001$ (a 5-bit result, counting the overflow)
- Discarding the fifth bit when fourth bit = 1, we get $0b1001$
- Answer—, the two’s complement representation of $-7$
Compute $-7 - 4$ using Sign Extension

- 4-bit 2’s complement numbers can only be between $+7$ and $-8$
- To perform subtraction—Negate the second operand and add
- Thus, actual computation—performed $-7 + (-4)$ after sign extension
- The two's-complement representations of $-7$ and $-4$ after sign extension = 0b11111001 and 0b11111100
Compute $-7 - 4$ using Sign Extension

- Adding $0b11111001$ and $0b11111100$
- Get $0b111110101$ (a 9-bit result, counting the overflow)
- Don't discard the ninth bit when eight bit = 0
- Taking the ninth bit as sign of the result we get $0b100001011$, the result is $-11_d$
Subtraction
Subtraction of two positive numbers

First step: find two’s complement
- Number 16: 0b00010000
- One's complement by inverting bits: 0b11101111
  - Add 1: 0b1

Two's complement of 16 bit: 0b11110000

Carry from low bits on addition: 1 1 1

Second step: addition of first number with two’s complement: 0b 0 0 0 0 0 0 1 0
Subtraction + 5 with + 3

Borrow out of high bit of subtraction

\[ \begin{array}{c}
\text{0b 1 0 0 1} \\
\text{0b 0 1 0 1} \\
\hline
\text{0b 0 1 0 0}
\end{array} \]
Borrow

Borrow out of high bit of subtraction

\[ \leftarrow 1 \]

\[ \begin{array}{cccc}
0b & 0 & 0 & 0 & 1 \\
\end{array} \quad (+1) \]

\[ \begin{array}{cccc}
0b & 0 & 0 & 1 & 1 \\
\end{array} \quad \text{Subtract} \quad (+3) \]

\[ \begin{array}{cccc}
0b & 1 & 1 & 1 & 0 \\
\end{array} \quad \text{Answer} \quad (+2) \]
Add the quantities +5 and –4 in 4-bit two's-complement notation

- 4-bit two's-complement representations of +5 and –4 are 0b0101 and 0b1100
- Adding these together gives 0b10001, which is the two's-complement representation of +1
Subtract 4 from –3 in 4-bit two's-complement notation

- 4-bit 2’s complement numbers can only be between +7 and −8
- To perform subtraction, we negate the second operand and add
- Thus, actual computation we want to perform is −3 + (−4)
- The two's-complement representations of −3 and −4 are 0b1101 and 0b1100
Subtract 4 from – 3

- Adding these quantities, we get 0b11001 (a 5-bit result, counting the overflow)
- Discarding the fifth bit when fourth bit = 1, we get 0b 1001, the two’s complement representation of –7
Subtract 4 from – 11

- 4-bit 2’s complement numbers can only be between +7 and –8. To perform subtraction, we use 8-bit numbers.
Example

- Find \(0b00000100 - 0b11110101\)
- Get \(0b100000111\) (a 9-bit result, counting the overflow)
- We don't discard the ninth bit when eight bit = 0
- Taking the ninth bit as sign of the result we get \(0b1\ 0000\ 0111\), the result is \(-7\)
Summary
We learnt

- Sign Extension generates a higher bit two’s complement representation of a number
- Addition uses carry to left
- Implement by a circuit as negation followed by addition is subtraction
- Subtraction needs borrow to right and therefore, it is easier to design circuit which does negation of second operand and then performs add operation
End of Lesson 2 on
Arithmetic Operations—
Addition and subtraction