Binary Subtraction
Binary Codes

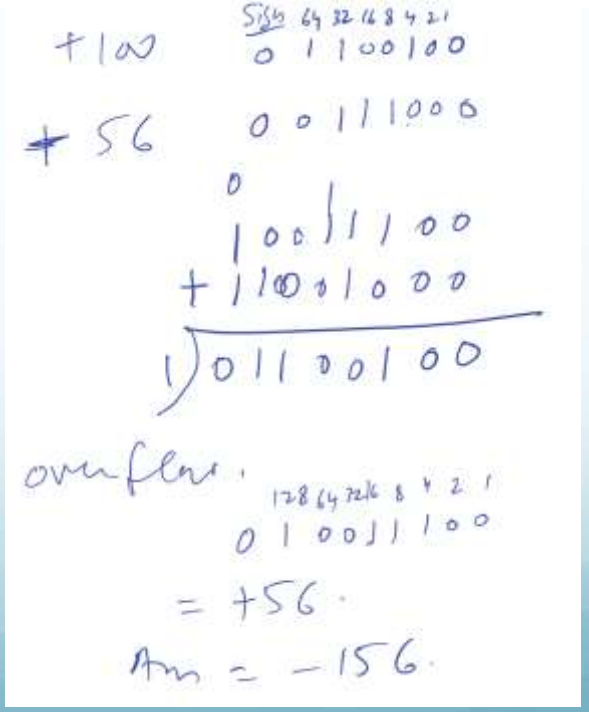
Instructor: Engr. Rimsha

Example Arithmetic (Signed 2's Complement)

Arithmetic 9 and 11

+ 9		00001001		- 9	11110111
+11		00001011		+11	00001011
+20		00010100		+ 2	00000010
+ 9		00001001		- 9	11110111
-11		11110101		-11	11110101
- 2		11111110		-20	11101100

Add -100 and -56



Arithmetic Subtraction

- Subtraction can be performed by simply converting the equation into an addition formula.
  - Take the 2's complement of the subtrahend (including the sign bit) and add it to the minuend (including the sign bit)
  - A carry out of the sign bit position is discarded
  - Note: Subtraction operation can be changed to an addition operation if the sign of the subtrahend is changed. This is easily done by taking it's 2's complement

Example

- Consider the subtraction (–6) –(–13) = +7
- In binary with eight bits the same is written as (11111010 – 11110011)
- This subtraction is changed to addition by taking 2's complement of the subtrahend (– 13) to give (+ 13)
- In binary this is 11111010 + 00001101 = 100000111
- Removing the end carry, we obtain the correct answer: 00000111(+7)

Binary Codes

- All symbols in a computer must be represented by a binary code (binary representation).
- An n-bit binary code is a group of n bits that can represent up to 2^n distinct combinations of 1's and 0's.
  - Each distinct combination represents a single symbol in the computer.

BCD Code (8 4 2 1)

- The most common representation for binary digits is the binary coded decimal (BCD) form which is a binary assignment of the decimal numbers.
  - This code is the simplest, most intuitive binary code for decimal digits and uses the same weights as a binary number, but only encodes the first ten values from 0 to 9 (6 out of 16 possible combinations remains unassigned).
  - A number with k distinct decimal digits will require 4k bits in BCD.
  - Each digit of a decimal value is converted to its respective binary representation.
  - BCD number needs more bits than its equivalent binary value?

Decimal Symbol	BCD Digit
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Multi-Digit BCD

Decimal Symbol	BCD Representation
10	0001 0000
11	0001 0001
212	0010 0001 0010
213	0010 0001 0011
5673	0101 0110 0111 0011
5684	0101 0110 1000 0100

BCD Addition

- BCD only represents each of the decimal digitals 0 through 9 as a single 4-bit binary value.
- When adding two BCD values, if the sum is equal to or less than 1001 (9), the corresponding BCD value is correct.
- However, when the binary sum is greater or equal to 1010 (10), the result is an invalid BCD value.
  - To overcome the invalid BCD value add 0110 (6) to the result to obtain the BCD representation and also produces a carry as required.
    - The use of 0110 (6) works because the difference between a carry in the most significant bit position of the binary sum and a decimal carry differ by 16-10 = 6.

BCD Addition Examples

4		0100		3		0011		9		1001
+5		0101		+7		0111		+9		1001
9		1001		10		1010		18		10010
						+0110				+0110
						0001 0000				0001 1000

Multi-Digit BCD Addition

Add 295 and 635 in BCD

BCD Carry	1	1		
	0010	1001	0101	295
	0110	0011	0101	+635
Binary Sum	1001	1101	1010	
Add 6		0110	0110	
BCD Sum	1001	0011	0000	930

BCD Arithmetic

- BCD arithmetic involving negative numbers uses the 10's complement for representing the negative numbers including the sign digit.
  - 0 (0000) represents a positive sign and 9 (1001) represents a negative sign
- As an example, imagine we want to add

  (+257) + (-160) = +97

	1
0 257	0000 0010 0101 0111
9 840	1001 1000 0100 0000
	1010 1010 1001 0111
	0110 0110
0 097	0000 0000 1001 0111

- Note: To obtain 10's complement of a BCD number, we first take the 9's complement (by subtraction of each digit from 9) and then add one to least significant digit

End of Lecture