1 Summary: Binary and Logic

• Binary Unsigned Representation : each 1-bit is a power of two, the right-most is for 2^0:
$0110101_2 = 2^5 + 2^4 + 2^2 + 2^0 = 32 + 16 + 4 + 1 = 53_{10}$

• Unsigned Range on n bits is $[0...(2^n - 1)]$. For n=4 we have the range
[0=0000, 1=0001, ..., 15=1111]

• Binary Two's Complement Representation : same as unsigned, except the power of two for the first bit (the sign bit) is with negative sign. When the signed bit is 0 its positive identical with unsigned transformation; when its 1 that power of two is negative:
$0100101_2 = \quad +2^5 + 2^2 + 2^0 = 32 + 4 + 1 = 37_{10}$
$1100101_2 = -2^6 + 2^5 + 2^2 + 2^0 = -64 + 32 + 4 + 1 = -64 + 37 = -27_{10}$

• it is essential to know how many bits to use, to know which bit is the sign bit for negatives:
7 bits two's complement : -27 = 1100101
10 bits two's complement: -27 = 1111100101

• Two's Complement Range on n bits is $[-2^{n-1}... + (2^{n-1} - 1)]$. For $n = 4$ we have the range
[-8=1000, -7=1001, -6=1010,... -1=1111, 0=0000, 1=0001, ..., 7=0111]

• 3-Step rule to convert from base 10 to two's complement negatives
- write in binary 27 = 0000011011
- flip the bits 1111100100
- add one 1111100101

• addition works like in base 10 with carry bit 1+1 = write 0, carry 1 ;
1+1+1 = write 1 carry 1 etc
• subtraction base2 : use addition of the negative value

• base $4 = 2^2$ use every two bits in base 2 to make a bit in base 4:
$0100101_2 = 0\_10\_01\_01 = 0\_(val = 2)\_(val = 1)\_(val = 1)_4 = 211_4 = 37_{10}$

• base $8 = 2^3$ use every three bits in base 2 to make a bit in base 8:
$0100101_2 = 0\_100\_101 = 0\_(val = 4)\_(val = 5)_8 = 45_8 = 37_{10}$

• base $16 = 2^4$ use every four bits in base 2 to make a bit in base 16:
$0100101_2 = 010\_0101 = (val = 2)\_(val = 5)_{16} = 25_{16} = 37_{10}$
$0011011_2 = 001\_1011 = (val = 1)\_(val = 11)_{16} = 1B_{16} = 16 + 11_{10} = 27_{10}$

• any base, like base=3 : break the number into powers of 3, with coefficients {0,1,2}
$35_{10} = 27 + 6 + 2 = 3^3 + 0 * 3^2 + 2 * 3^1 + 2 * 3^0 = 1022_3$

4-bit Signed Binary Number Comparison

1posrep flip(posrep)

Decimal	Signed Magnitude	Signed One's Complement	Signed Two's Complement
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
-0	1000	1111	–
-1	1001	1110	1111
-2	1010	1101	1110
-3	1011	1100	1101
-4	1100	1011	1100
-5	1101	1010	1011
-6	1110	1001	1010
-7	1111	1000	1001

• LOGIC TABLE

A	B	$A \wedge B$ (AND)	$A \vee B$ (OR)	$A \oplus B$ (XOR)	$\neg(A \wedge B)$ (NAND)
1	1	1	1	0	0
1	0	0	1	1	1
0	1	0	1	1	1
0	0	0	0	0	1

• DNF (OR between clauses for out=1); CNF (OR between negated clauses for out=0). EXAMPLE:

A	B	C	output	DNF	CNF
0	0	0	0		$A \vee B \vee C$
0	0	1	1	$\neg A \wedge \neg B \wedge C$	
0	1	0	0		$A \vee \neg B \vee C$
0	1	1	1	$\neg A \wedge B \wedge C$	
1	0	0	1	$A \wedge \neg B \wedge \neg C$	
1	0	1	0		$\neg A \vee B \vee \neg C$
1	1	0	1	$A \wedge B \wedge \neg C$	
1	1	1	0		$\neg A \vee \neg B \vee \neg C$

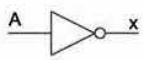
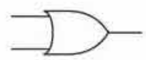
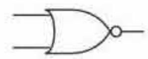
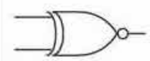
$DNF = (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge B \wedge C) \vee (A \wedge \neg B \wedge \neg C) \vee (A \wedge B \wedge \neg C)$

$CNF = (A \vee B \vee C) \wedge (A \vee \neg B \vee C) \wedge (\neg A \vee B \vee \neg C) \wedge (\neg A \vee \neg B \vee \neg C)$

• BOOLEAN ALGEBRA LAWS (1=true=T; 0=false=F)

Commutative laws	$p \wedge q \equiv q \wedge p$ $p \vee q \equiv q \vee p$
Associative laws	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$ $(p \vee q) \vee r \equiv p \vee (q \vee r)$
Distributive laws	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$ $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
Identity laws	$p \wedge T \equiv p$ $p \vee F \equiv p$
Complement laws	$p \wedge \neg p \equiv F$ $p \vee \neg p \equiv T$
Annihilator laws	$p \wedge F \equiv F$
	$p \vee T \equiv T$
Idempotence laws	$p \wedge p \equiv p$
	$p \vee p \equiv p$
Absorption laws	$p \wedge (p \vee q) \equiv p$
	$p \vee (p \wedge q) \equiv p$
Double negation law	$\neg(\neg p) \equiv p$
De Morgan's laws	$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$

- LOGIC GATES

Name	NOT	AND	NAND	OR	NOR	XOR	XNOR
Alg. Expr.	$\overline{A}$	AB	$\overline{AB}$	$A+B$	$\overline{A+B}$	$A\oplus B$	$\overline{A\oplus B}$
Symbol							
Truth Table	A X 0 **1** 1 **0**	B A X 0 0 **0** 0 1 **0** 1 0 **0** 1 1 **1**	B A X 0 0 **1** 0 1 **1** 1 0 **1** 1 1 **0**	B A X 0 0 **0** 0 1 **1** 1 0 **1** 1 1 **1**	B A X 0 0 **1** 0 1 **0** 1 0 **0** 1 1 **0**	B A X 0 0 **0** 0 1 **1** 1 0 **1** 1 1 **0**	B A X 0 0 **1** 0 1 **0** 1 0 **0** 1 1 **1**