

Bit Manipulation


CPU \rightarrow i's o's

18 \rightarrow $(10010)_2$
16
Binary Ref

1 0 1 0 1 0
 \downarrow most significant bit
MSB
 \downarrow Least significant bit
LSB

$$57_{10} \rightarrow$$

2	57	R
2	28	1
2	14	0
2	7	0
2	3	1
2	1	
	0	1



$$(57)_{10} \rightarrow (11001)_2$$

Integer Family

um by

nibble

byte

short

int

long

4 bits

8 bits

16

32

64

2^4

2^8

2^{16}

2^{32}

2^{64}

16 numbers

nibble

2^4

$= 16$

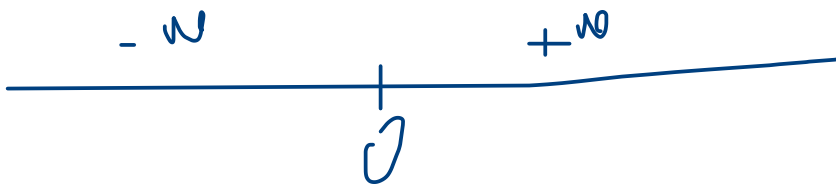
B.

D

0000	—	0
0001	—	1
0010	—	2
0011	—	3
0100	—	4
0101	—	5
0110	—	6
0111	—	7
1000	—	8
1001	—	9
1010	—	10
1011	—	11
1100	—	12
1101	—	13
1110	—	14
1111	—	15

— w 3.

A, X



nibble

2^4

$\boxed{=16}$

	B.	D	
1			
	0 0 0 0	0	+ve
	0 0 0 1	1	
	0 0 1 0	2	
	0 0 1 1	3	
	0 1 0 0	4	
	0 1 0 1	5	
	0 1 1 0	6	
	0 1 1 1	7	
	1 0 0 0	8	-ve
	1 0 0 1	9	
	1 0 1 0	10	
	1 0 1 1	11	
	1 1 0 0	12	
	1 1 0 1	13	
	1 1 1 0	14	
	1 1 1 1	15	

A 2 \rightarrow \pm $\boxed{0}$

1. MSB = 0 \rightarrow +ve number
2. MSB = 1 \rightarrow -ve number

remaining 3 bits used to make number in decimal

A3

nibble

2⁴

16

2's = 1's + 1
complement

1's comp =
toggle bits

B.	D
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	
1010	
1011	
1100	
1101	
1110	
1111	

+ 4

- 4

msb

0

look for
last 3 bit

2's

complement
of last 3 bit

1000

= (1000) + 0001

= 0111 + 0001 → + 0001
1000 = 8

A3 ✓

nibble

2⁴

= 16

	B.	D
0	0000	0
	0001	1
	0010	2
	0011	3
	0100	4
	0101	5
	0110	6
	0111	7
<hr/>		
1	1000	8
	1001	9
	1010	10
	1011	11
	1100	12
	1101	13
	1110	14
	1111	15

+ 4

- 4

$$(1001)_2 = (1001)_2 + 0001 \rightarrow$$

$$0110 + 0001$$

$$\begin{array}{r} 1010 \\ \downarrow \\ 0101 \\ + 0001 \\ \hline 0110 = 6 \end{array}$$

$$\begin{array}{r} 0110 \\ 0001 \\ \hline 0111 = 7 \end{array}$$

A3 ✓

nibble

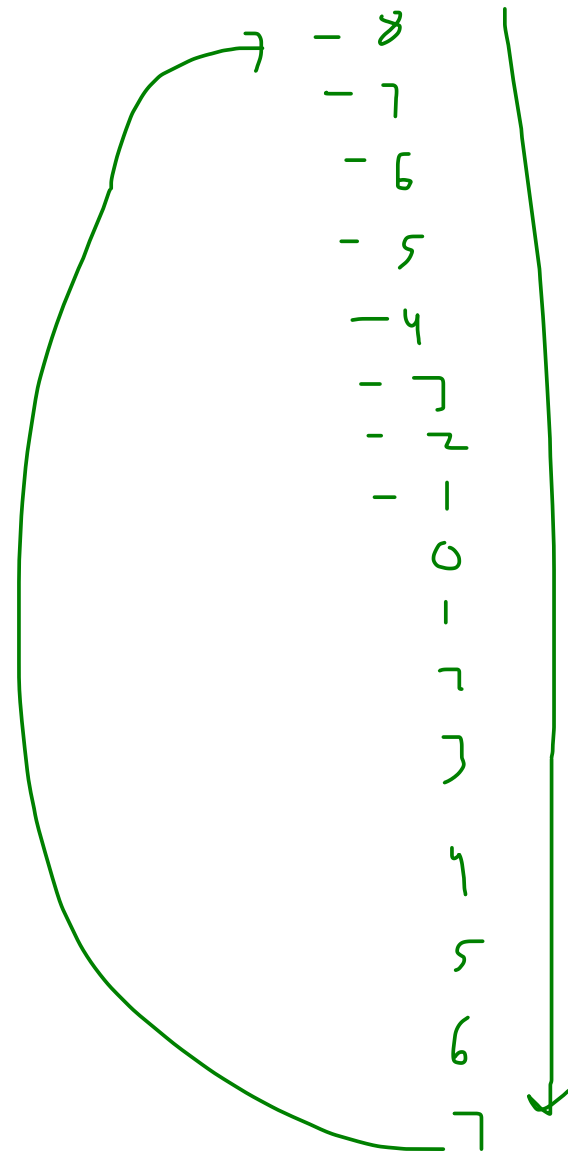
2⁴

16

	B.	D
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	10
11	1011	11
12	1100	12
13	1101	13
14	1110	14
15	1111	15

+ n

- n



msb
0

1. Convert in decimal
2. Add +ve sign

msb
1

1. Take 2s compliment of binary number
2. Add -ve sign

Correct
approach
to find
range

nibble

$$\rightarrow 4 \text{ bits} \rightarrow 2^4 \rightarrow -2^3 + 0 \cdot 2^3 - 1$$

16
numbers

$$-8 + 0 \cdot 7$$

byte

$$\rightarrow 8 \rightarrow 2^8 \rightarrow -2^7 + 0 \cdot 2^7 - 1$$

Short

$$\rightarrow 16 \rightarrow 2^{16} \rightarrow -2^{15} + 0 \cdot 2^{15} - 1$$

int

$$\rightarrow 32 \rightarrow 2^{32} \rightarrow -2^{31} + 0 \cdot 2^{31} - 1$$

long

$$\rightarrow 64 \rightarrow 2^{64} \rightarrow -2^{63} + 0 \cdot 2^{63} - 1$$

Nike 0 = 5

out of range ?

Nike a = 12

not \rightarrow 4
12

2	12	
2	0	0
2	3	0
2	1	1
	0	1

1 1 0 0
- - - -

1000	- 4
1001	- 7
1010	- 6
1011	- 5
1100	- 4
1101	- 3
1110	- 2
1111	- 1
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

out of range ?

With a = 16
x = 0

2	1	0
2	6	0
2	4	0
2	2	0
2	1	0
	0	1

X 0 0 0 0

1000	-6
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

16 \rightarrow (0001 0 0 0 0)

↓

Byte

ind

16 \rightarrow 0000 | 0000

27
015

B = D

msb
0

1. Convert in decimal
2. Add +ve sign

msb
1

1. Take 2s compliment of binary number
2. Add -ve sign

P = B

+ve

1. convert in binary
2. Fit in bits

$$(16)_{10} = \begin{array}{r} 16 \\ \times \quad 0000 \\ \hline \end{array}$$

$$(16) \rightarrow 0000 = 0$$

↓
N/A

-ve

1. Ignore sign
2. Convert in Binary
3. Fit in Bits
4. 2s comp

$$-7 \rightarrow \begin{array}{c|c|c} 7 & & \\ \hline 2 & 3 & 1 \\ \hline 2 & 1 & 1 \\ \hline & 0 & 1 \end{array} \uparrow$$

$$\begin{array}{c} 0111 \\ \hline \end{array} \downarrow = \begin{array}{r} 1000 \\ 0001 \\ \hline 1001 \end{array} \checkmark$$

Basic operators of Bit Manipulation

$ $ OR	\cdot AND	\wedge XOR	$<<$ left shift	$>>$ Right shift	$>>>$ Triple R S
\sim 1's Comp	$-X$ 2's Comp				

1. OR \rightarrow | \rightarrow | ~~0~~

a	b	Rs
0	0	0
0	1	1
1	0	1
1	1	1

$$\begin{array}{c|c} 0 & 0 = 0 \\ 1 & 0 = 1 \end{array}$$

2. AND \rightarrow &
0

a	b	Rs
0	0	0
0	1	0
1	0	0
1	1	1

$$\begin{array}{c} 0 \& 1 = 0 \\ 1 \& 1 = 1 \end{array}$$

3 XOR

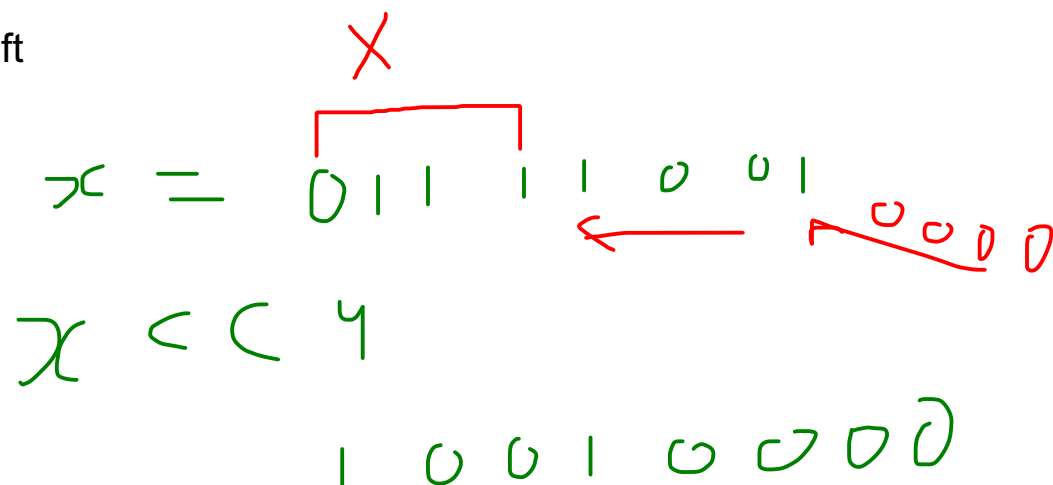
\wedge

\neg

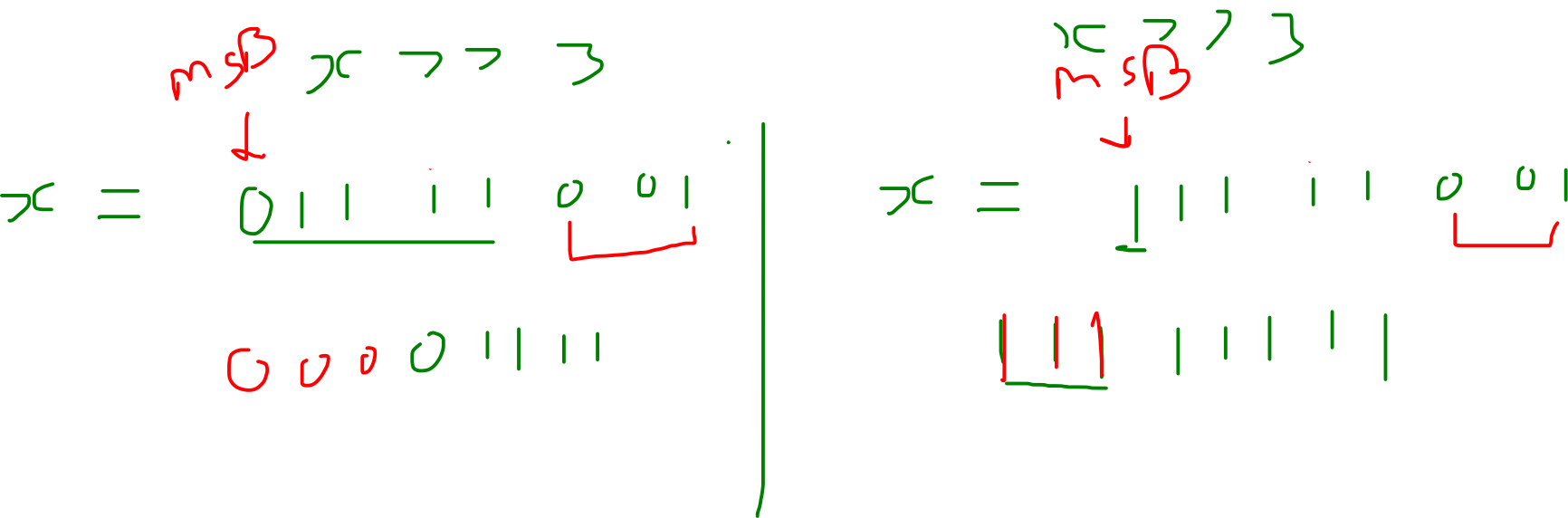
a	b	$a \wedge b$
0	0	0
0	1	0
1	0	0
1	1	1

a
 $\neg a$
 a
 $\neg a$

4. << left shift



5. >> right shift



כך



צ י י ד אק



Basic Knowledge

ON (OR)

$$x = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 0 & 1 \\ \hline \end{array}$$

$$\text{mask} = \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline \end{array}$$

$$x \mid \text{mask} = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ \hline \end{array}$$

Bit ON

$$\text{mask} = \begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline \end{array}$$

$$\text{mask} \ll 3$$

$$\begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline \end{array}$$

1 < 3

OFF (AND)

$$x = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 0 & 1 \\ \hline \end{array}$$

$$\text{mask} = \begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

$$x \& \text{mask}$$

$$\begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 0 & 0 & 1 \\ \hline \end{array}$$

Bit off

$$\text{mask} = 1$$

$$\begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline \end{array}$$

mask < 4

$$00010000 \rightarrow 15 \& 10 \Rightarrow \begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

Toggle (XOR)

$$\begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 0 & 1 \\ \hline \end{array}$$

$$\text{mask} = \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline \end{array}$$

$$x \oplus \text{mask}$$

$$\begin{array}{|c|c|c|c|c|c|} \hline 1 & 0 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

Bit Toggle

Check bit is on/off (AND)

$$x = \begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ \hline \end{array}$$

Check
bit is on/off (AND)

$$\begin{array}{ccccccc} & & & \downarrow & & & \\ x = & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ y = & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}$$

$$x \& y = 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0$$

$$\text{ans} == 0$$

→ Bit off

$$\begin{array}{ccccccc} & & & \downarrow & & & \\ x = & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ y = & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{array}$$

$$x \& y = 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0$$

$$\text{ans} \neq 0$$

Bit on

Right Set Bit Mask

$$x = 16 = \underbrace{256's}_{1001\underline{1}00} \quad \downarrow$$

$$xsbm = \underbrace{29 \ 0's}_{100}$$

$$\begin{aligned} xsbm &= x \& x \\ &= x \& -x \end{aligned}$$

$$x = \underbrace{A \text{ 1's \& 0's}}_{\text{A}} \mid \underbrace{B \text{ 0's}}_{\text{B}}$$

~~$$x' = \underbrace{A \text{ 0's \& 1's}}_{\text{A}} \mid \underbrace{B \text{ 1's}}_{\text{B}}$$~~

$$x'' = x' + 1 = \underbrace{A \text{ 0's \& 1's}}_{\text{A}} \mid \underbrace{B \text{ 0's}}_{\text{B}}$$

$$x \text{ \& } x'' = \underbrace{A \text{ 0's}}_{\text{A}} \mid \underbrace{B \text{ 0's}}_{\text{B}}$$

R5 Bm

