CSC 220: Computer Organization

Unit 1
Number Systems

Prepared by:

Md Saiful Islam, PhD

Department of Computer Science

College of Computer and Information Sciences

Overview

- Common Number Systems
- Conversion Among Bases
- Binary Coded Decimal (BCD)

Chapter-1

M. Morris Mano, Charles R. Kime and Tom Martin, **Logic and Computer Design Fundamentals**, Global (5th) Edition, Pearson Education Limited, 2016. ISBN: 9781292096124

Decimal review

- Decimal numbers consist of digits from 0 to 9, each with a weight.

1	6	2	.	3	7	5	digits
100	10	1		1/10	1/100	1/1000	weights

- Notice that the weights are all powers of the base, which is 10.

1	6	2	.	3	7	5	digits
10^2	10^1	10^0		10^{-1}	10^{-2}	10^{-3}	weights

- To find the decimal value of a number, you can multiply each digit by its weight and sum the products:

$$(1 \times 10^2) + (6 \times 10^1) + (2 \times 10^0) + (3 \times 10^{-1}) + (7 \times 10^{-2}) + (5 \times 10^{-3}) = 162.375$$

Common Number Systems

System	Base	Symbols	Used by humans?	Used in computers?
Decimal	10	0, 1, … 9	Yes	No
Binary	2	0, 1	No	Yes
Octal	8	0, 1, … 7	No	No
Hexa-decimal	16	0, 1, … 9, A, B, … F	No	No

Quantities/Counting

Decimal	Binary	Octal	Hexa-decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11

Binary Numbers

- Binary, or base 2, numbers consist of only the digits 0 and 1. The weights are now powers of 2.
- For example, consider the binary number 1101.01:

1	1	0	1	.	0	1	binary digits, or bits
2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	weights in decimal

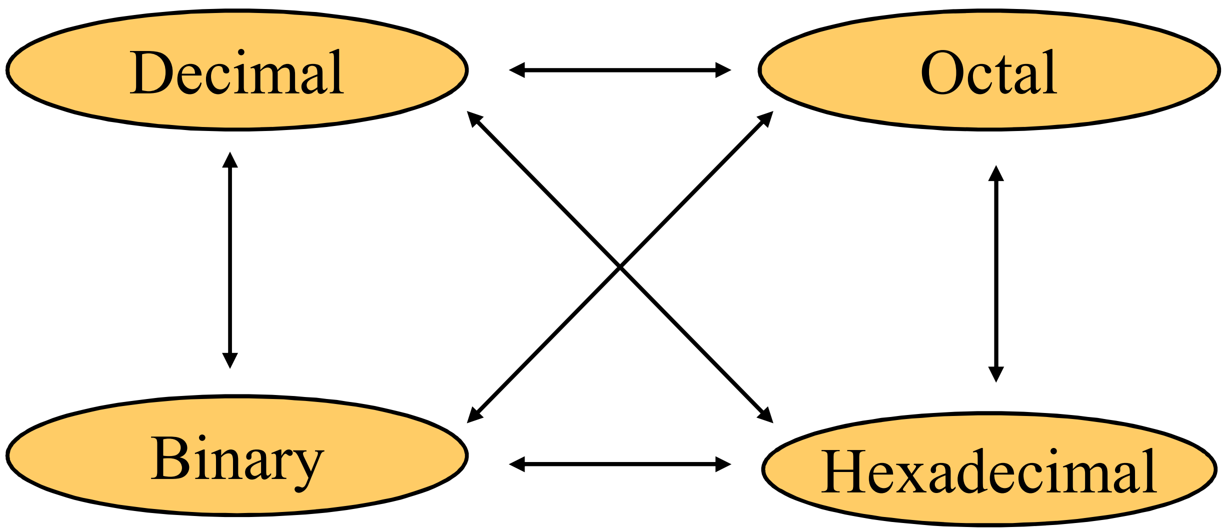
- The decimal value of 1101.01 is computed just like before:

$$(1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) + (0 \times 2^{-1}) + (1 \times 2^{-2}) =$$
$$8 + 4 + 0 + 1 + 0 + 0.25 = 13.25$$

Some powers of 2		
$2^0 = 1$	$2^4 = 16$	$2^8 = 256$
$2^1 = 2$	$2^5 = 32$	$2^9 = 512$
$2^2 = 4$	$2^6 = 64$	$2^{10} = 1024$
$2^3 = 8$	$2^7 = 128$	

Conversion Among Bases

- The possibilities:



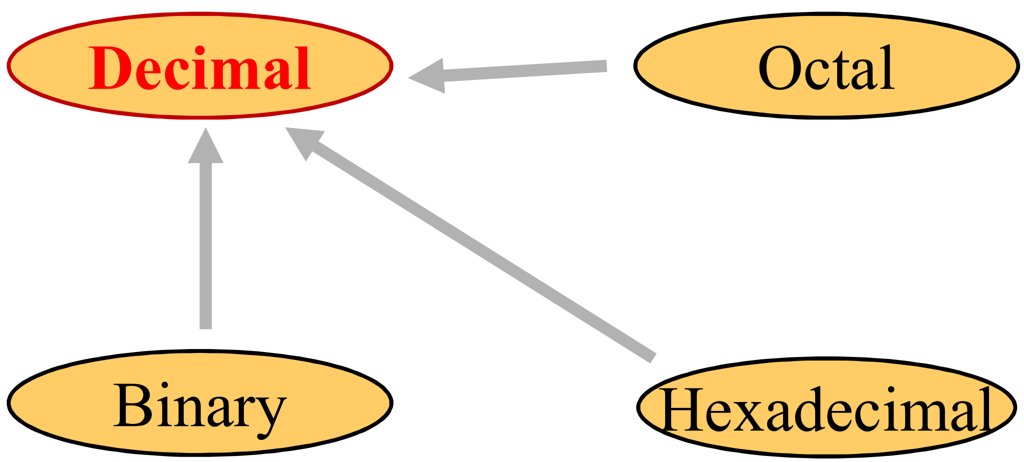
Quick Example

$$25_{10} = 11001_2 = 31_8 = 19_{16}$$

Base

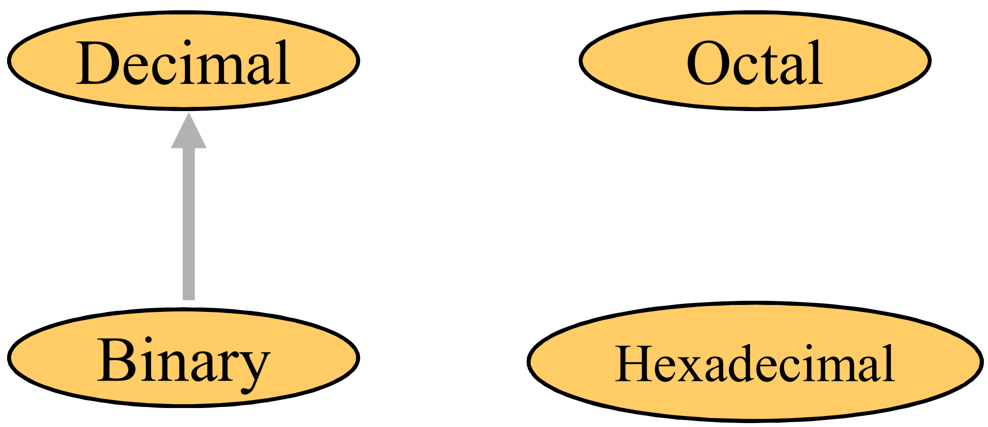
Group 1: To Decimal

- Technique
  - Multiply each digit by b^n, where b is the "base"
  - n is the position of the bit, starting from 0 on the right
  - Add the results



Binary to Decimal

- Technique
  - Multiply each bit by 2^n, where 2^n is the “weight” of the bit
  - n is the position of the bit, starting from 0 on the right
  - Add the results



Example

Bit “0”

$$101011_2 \Rightarrow \begin{array}{rcl} 1 \times 2^0 & = & 1 \\ 1 \times 2^1 & = & 2 \\ 0 \times 2^2 & = & 0 \\ 1 \times 2^3 & = & 8 \\ 0 \times 2^4 & = & 0 \\ 1 \times 2^5 & = & 32 \\ \hline & & 43_{10} \end{array}$$

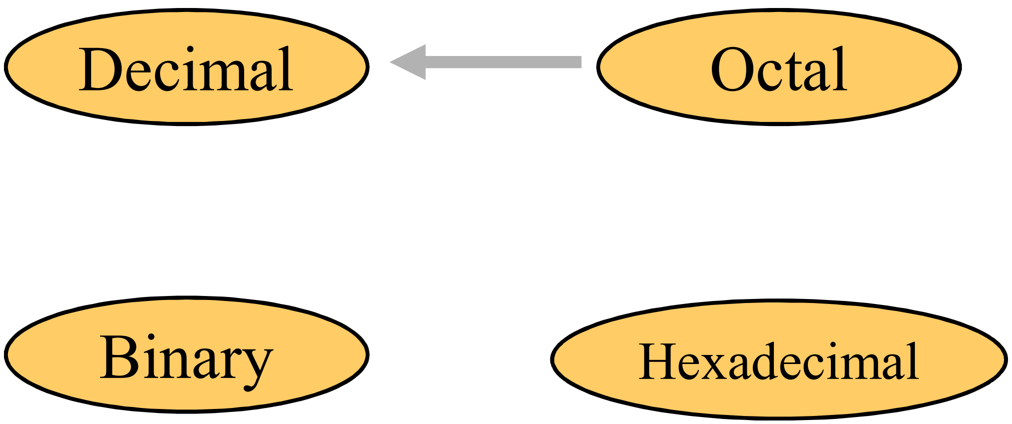
Converting Binary to Decimal

What is the decimal equivalent of the binary number 1101110?

$$
\begin{aligned}
& 1 \times 2^6 = 1 \times 64 = 64 \\
+\ & 1 \times 2^5 = 1 \times 32 = 32 \\
+\ & 0 \times 2^4 = 0 \times 16 = 0 \\
+\ & 1 \times 2^3 = 1 \times 8 = 8 \\
+\ & 1 \times 2^2 = 1 \times 4 = 4 \\
+\ & 1 \times 2^1 = 1 \times 2 = 2 \\
+\ & 0 \times 2^0 = 0 \times 1 = 0 \\
& = 110 \text{ in base } 10
\end{aligned}
$$

Octal to Decimal

- Technique
  - Multiply each digit by 8^n, where 8^n is the “weight” of the bit
  - n is the position of the bit, starting from 0 on the right
  - Add the results

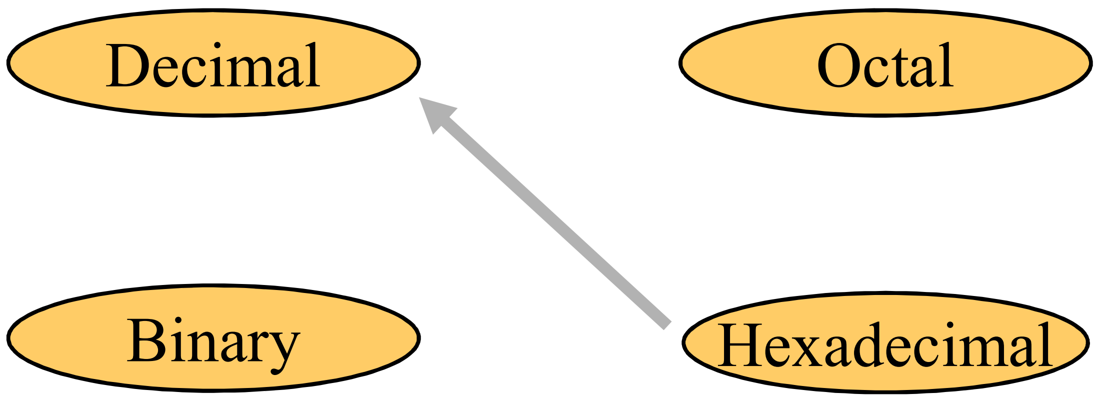


Example

$$
\begin{array}{llr}
724_8 \Rightarrow & 4 \times 8^0 = & 4 \\
 & 2 \times 8^1 = & 16 \\
 & 7 \times 8^2 = & 448 \\
\hline
 & & 468_{10}
\end{array}
$$

Hexadecimal to Decimal

- Technique
  - Multiply each digit by 16^n, where 16^n is the “weight” of the bit
  - n is the position of the bit, starting from 0 on the right
  - Add the results



Base 16 is useful too

- The hexadecimal system uses 16 digits:

  0 1 2 3 4 5 6 7 8 9 A B C D E F

- Hexadecimal is useful as a shorthand for binary numbers.
  - Since $16 = 2^4$, one hex digit is equivalent to four bits (including leading 0s).
  - It's often easier to work with numbers like "B4" instead of "10110100".
- Hex shows up in many different contexts.
  - IP addresses, such as "80.AE.05.27".
  - RGB color triplets, like "C0C0FF".
- You can convert between base 10 and base 16 using the same method as for converting from decimal to binary.

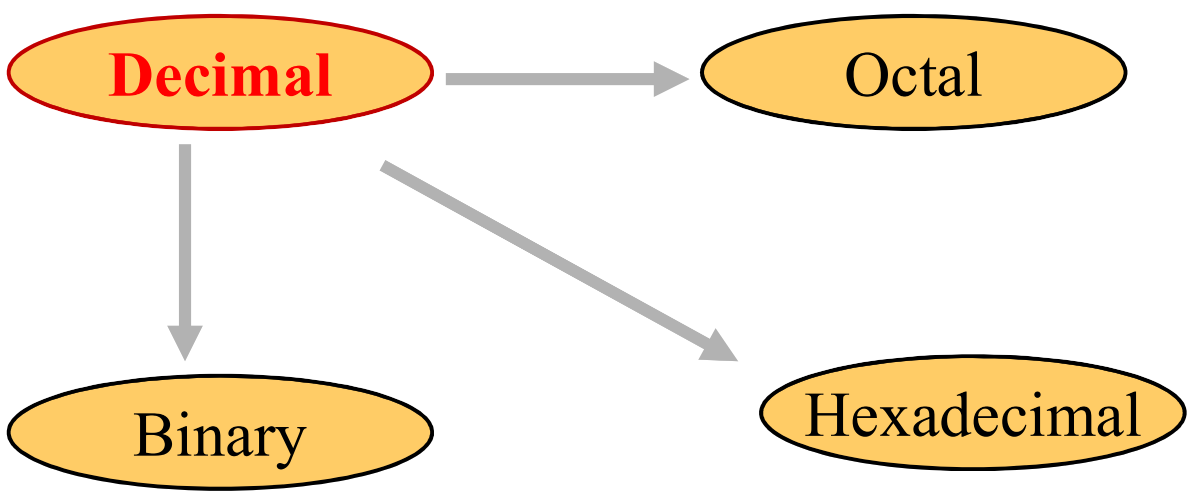
Decimal	Binary	Hex
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Example

$$
\begin{array}{llrrrrl}
ABC_{16} \Rightarrow & C \times 16^0 & = 12 \times & 1 & = & 12 \\
& B \times 16^1 & = 11 \times & 16 & = & 176 \\
& A \times 16^2 & = 10 \times & 256 & = & 2560 \\
\hline
& & & & & 2748_{10}
\end{array}
$$

Group 2: From Decimal

- Technique
  - Divide by the **base**, keep track of the remainder
  - First remainder is bit 0 (LSB, least-significant bit)
  - Second remainder is bit 1
  - Etc.



Why does this work?

- This same idea works for converting from decimal to any other base.
- Think about "converting" 162 from decimal to decimal:

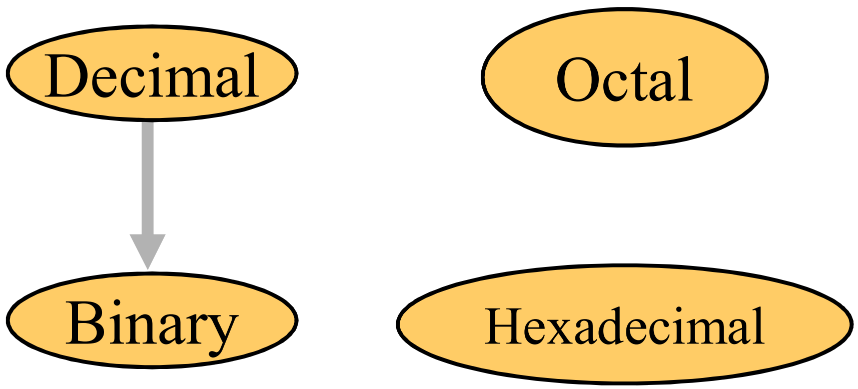
$$
\begin{aligned}
162 / 10 &= 16 \quad \text{rem } 2 \\
16 / 10 &= 1 \quad \text{rem } 6 \\
1 / 10 &= 0 \quad \text{rem } 1
\end{aligned}
$$

- After each division, the remainder contains the rightmost digit of the dividend, while the quotient holds the remaining digits.
- Similarly when converting fractions, each multiplication strips off the leftmost digit as the integer result, leaving the remaining digits in the fractional part.

$$
\begin{aligned}
0.375 \times 10 &= 3.750 \\
0.750 \times 10 &= 7.500 \\
0.500 \times 10 &= 5.000
\end{aligned}
$$

Decimal to Binary

- Technique
  - Divide by two, keep track of the remainder
  - First remainder is bit 0 (LSB, least-significant bit)
  - Second remainder is bit 1
  - Etc.



Example

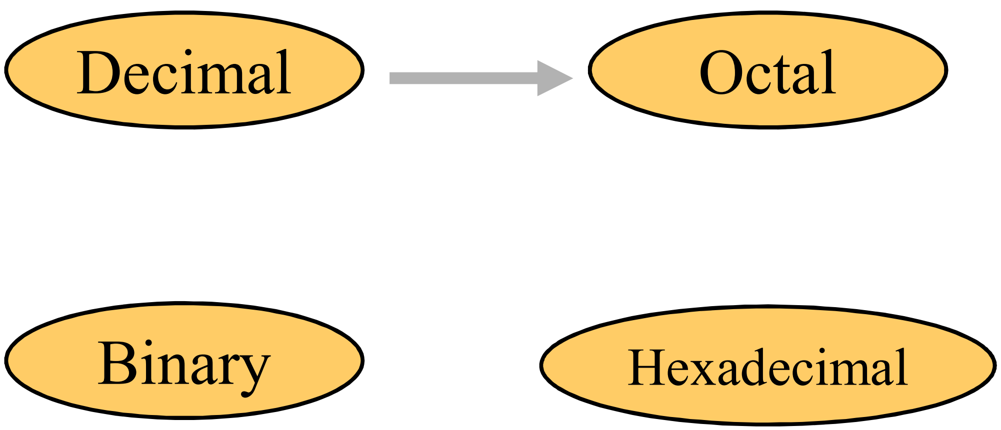
$125_{10} = ?_2$

2	125	
2	62	1
2	31	0
2	15	1
2	7	1
2	3	1
2	1	1
	0	1

$125_{10} = 1111101_2$

Decimal to Octal

- Technique
  - Divide by 8
  - Keep track of the remainder



Example

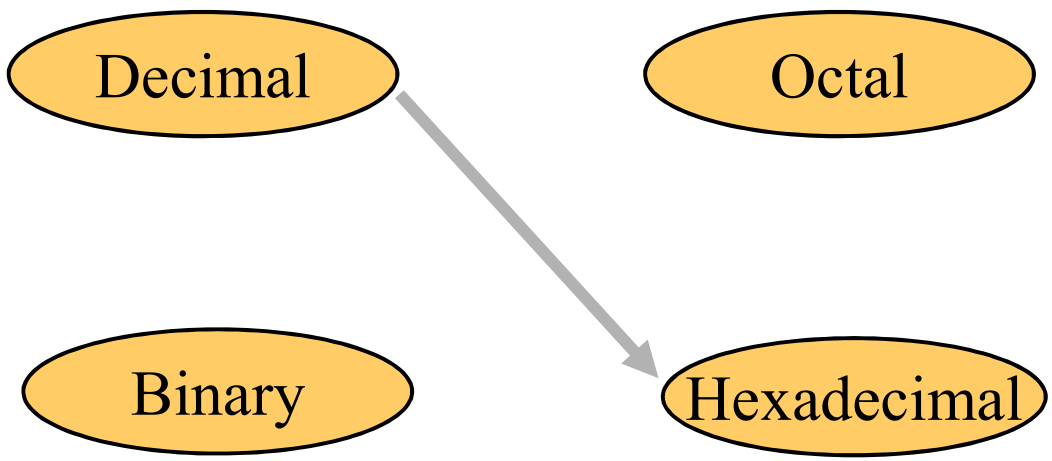
$$1234_{10} = ?_8$$

8	1234	
8	154	2
8	19	2
8	2	3
	0	2

$$1234_{10} = 2322_8$$

Decimal to Hexadecimal

- Technique
  - Divide by 16
  - Keep track of the remainder



Example

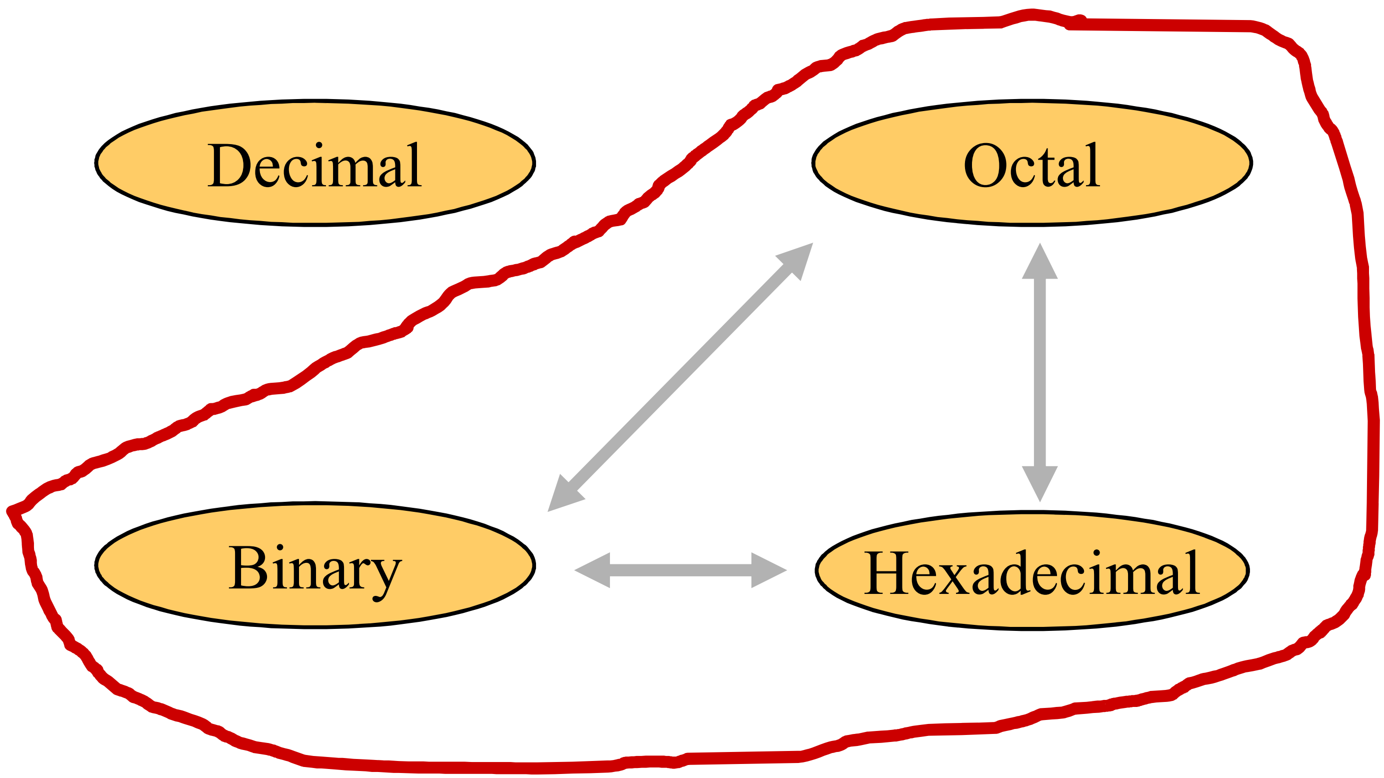
$1234_{10} = ?_{16}$

```
16 | 1234
16 |   77     2
16 |    4    13 = D
        0     4
```

$1234_{10} = 4D2_{16}$

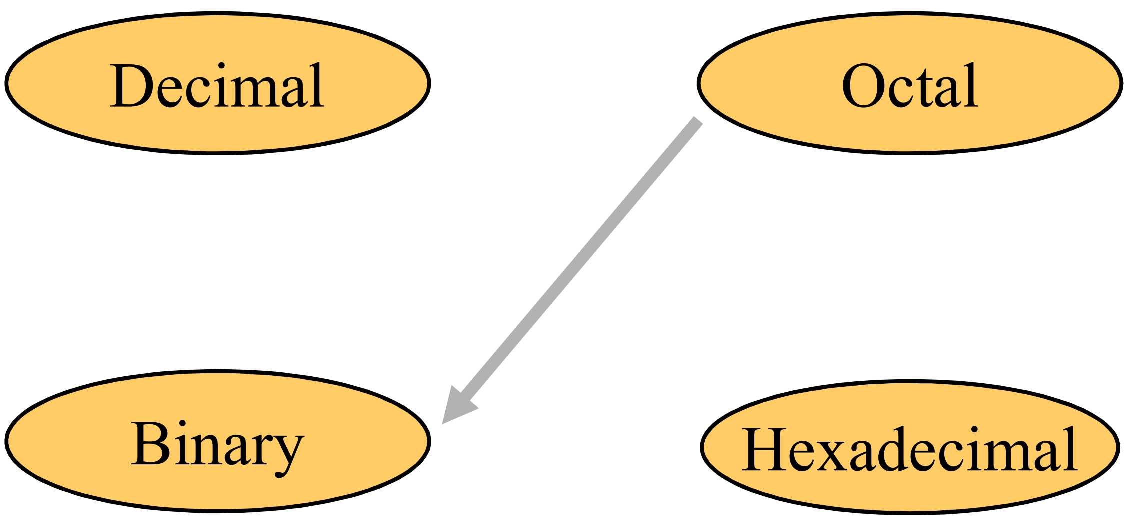
Group-3: Except Decimal

- Technique
  - Convert each digit to a equivalent binary representation



Octal to Binary

- Technique
  - Convert each octal digit to a 3-bit equivalent binary representation



Example

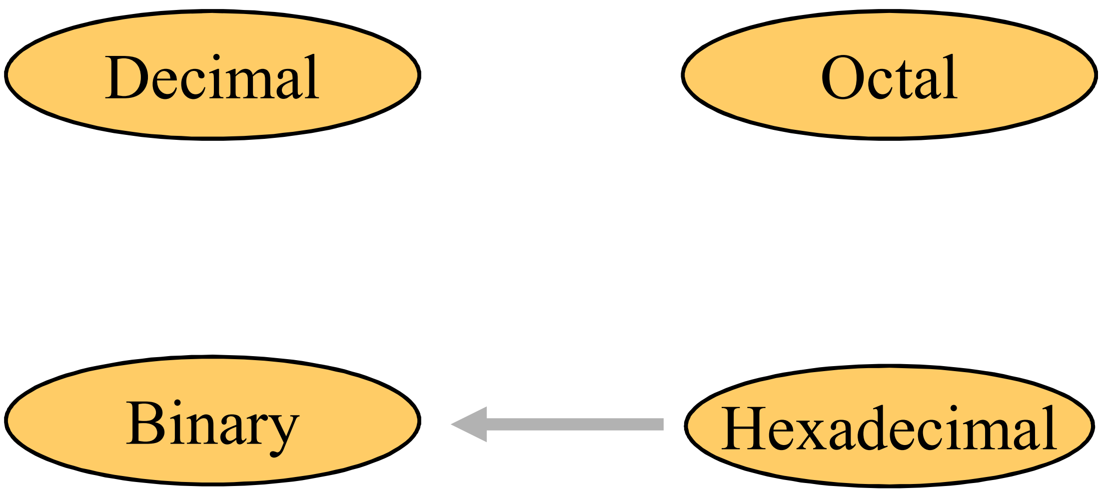
$705_8 = ?_2$

$$\begin{array}{ccc} 7 & 0 & 5 \\ \downarrow & \downarrow & \downarrow \\ 111 & 000 & 101 \end{array}$$

$705_8 = 111000101_2$

Hexadecimal to Binary

- Technique
  - Convert each hexadecimal digit to a 4-bit equivalent binary representation



Binary and hexadecimal conversions

- Converting from hexadecimal to binary is easy: replace each hex digit with its equivalent four-bit binary value.

$$
\begin{array}{lcccccccc}
261.A5_{16} & = & 2 & 6 & 1 & . & A & 5_{16} \\
 & = & 0010 & 0110 & 0001 & . & 1010 & 0101_2
\end{array}
$$

- To convert from binary to hexadecimal, partition the binary number into groups of four bits, starting from the point. (Add 0s to the ends if needed.) Then replace each four-bit group by the corresponding hex digit.

$$
\begin{array}{lcccccc}
10110100.001011_2 & = & 1011 & 0100 & . & 0010 & 1100_2 \\
 & & B & 4 & . & 2 & C_{16}
\end{array}
$$

Binary	Hex
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Example

$10AF_{16} = ?_2$

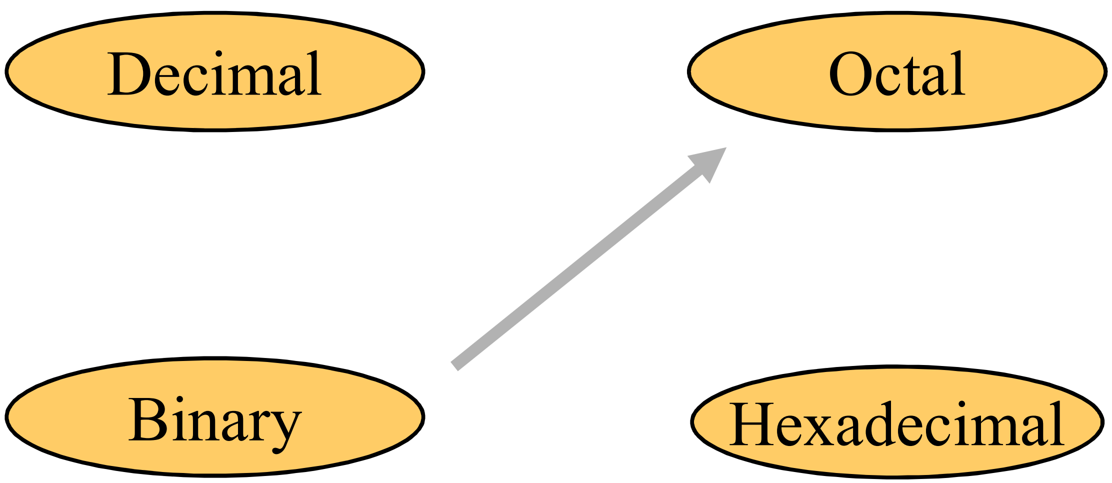
1 0 A F

0001 0000 1010 1111

$10AF_{16} = 0001000010101111_2$

Binary to Octal

- Technique
  - Group bits in threes, starting on right
  - Convert to octal digits



Example

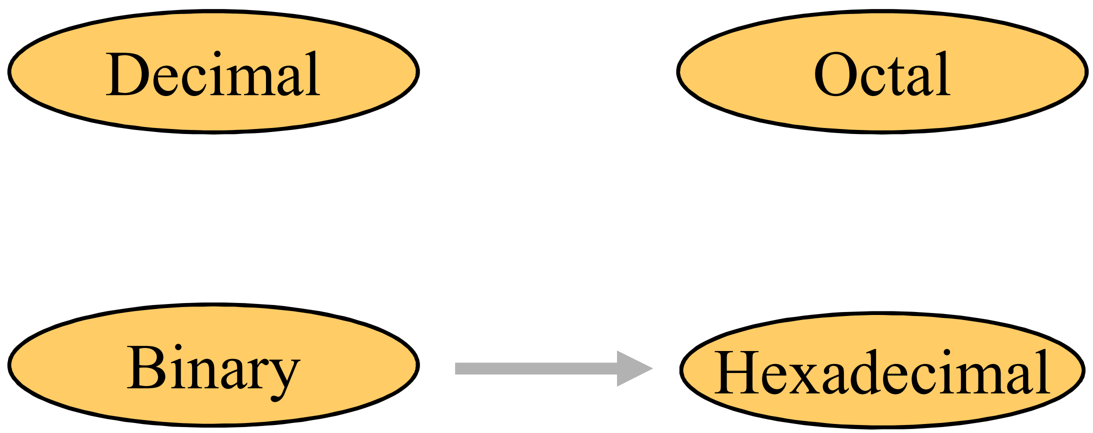
$1011010111_2 = ?_8$

$$\begin{array}{cccc} 1 & 011 & 010 & 111 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 3 & 2 & 7 \end{array}$$

$1011010111_2 = 1327_8$

Binary to Hexadecimal

- Technique
  - Group bits in fours, starting on right
  - Convert to hexadecimal digits



Example

$1010111011_2 = ?_{16}$

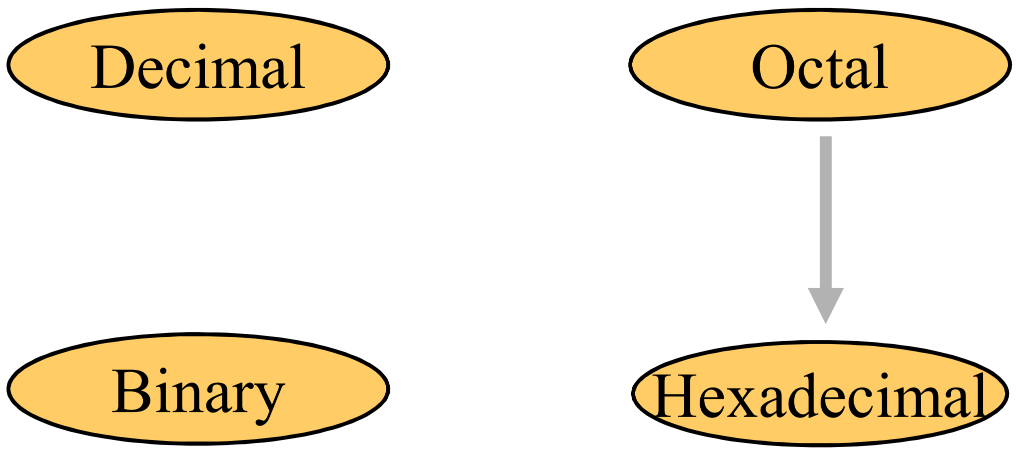
10 1011 1011

2 B B

$1010111011_2 = 2BB_{16}$

Octal to Hexadecimal

- Technique
  - Use binary as an intermediary



Example

$1076_8 = ?_{16}$

1 0 7 6

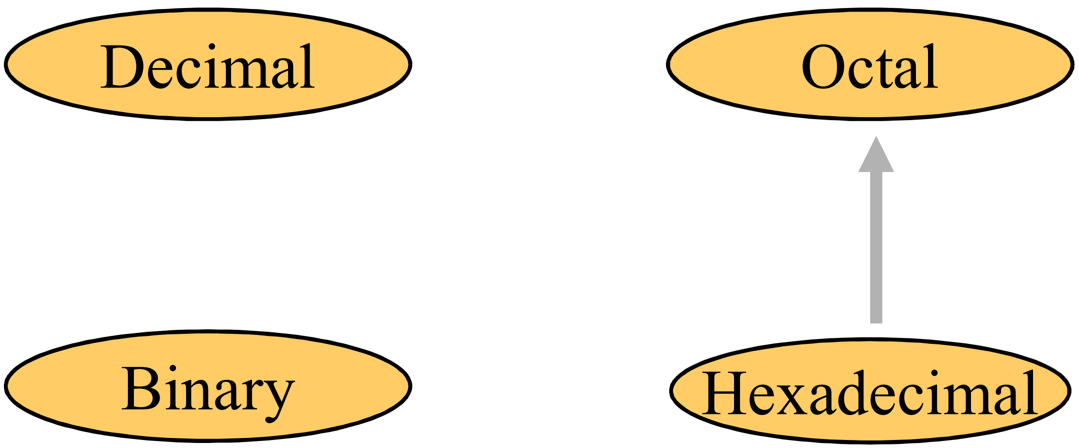
001 000 111 110

2 3 E

$1076_8 = 23E_{16}$

Hexadecimal to Octal

- Technique
  - Use binary as an intermediary



Example

$1F0C_{16} = ?_8$

1	F	0	C
0001	1111	0000	1100

0|001|111|1 00|00 1|100

1	7	4	1	4

$1F0C_{16} = 17414_8$

Exercise – Convert ...

Decimal	Binary	Octal	Hexa-decimal
33			
	1110101		
		703	
			1AF

Don't use a calculator!

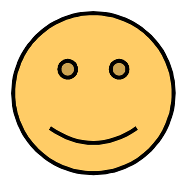
Skip answer

Answer

Exercise – Convert …

Answer

Decimal	Binary	Octal	Hexa-decimal
33	100001	41	21
117	1110101	165	75
451	111000011	703	1C3
431	110101111	657	1AF



Fractions

- Same technique for the 1^{st} group (To Decimal)
- Different technique for the 2^{nd} group (From Decimal)
- Same technique for the 3^{rd} group (Except Decimal)

- Decimal to decimal (just for fun)

$$
\begin{array}{lrl}
3.14 \Rightarrow & 4 \times 10^{-2} & = 0.04 \\
& 1 \times 10^{-1} & = 0.1 \\
& 3 \times 10^{0} & = \underline{3\qquad} \\
& & \ 3.14
\end{array}
$$

Fractions

- Group-1 (To Decimal)

Ex: Binary to decimal

10.1011 =>

$$
\begin{array}{l}
1 \times 2^{-4} = 0.0625 \\
1 \times 2^{-3} = 0.125 \\
0 \times 2^{-2} = 0.0 \\
1 \times 2^{-1} = 0.5 \\
0 \times 2^{0} = 0.0 \\
1 \times 2^{1} = \underline{2.0} \\
\qquad\qquad\quad 2.6875
\end{array}
$$

Fractions

- Group-2 (From Decimal)

▪ Similarly when converting fractions, each multiplication strips off the leftmost digit as the integer result, leaving the remaining digits in the fractional part.

$$0.375 \times 10 = 3.750$$
$$0.750 \times 10 = 7.500$$
$$0.500 \times 10 = 5.000$$

Fractions

Ex: Decimal to binary

3.14579

$$
\begin{array}{r}
.14579 \\
\times \quad 2 \\
\hline
\boxed{0}.29158 \\
\times \quad 2 \\
\hline
\boxed{0}.58316 \\
\times \quad 2 \\
\hline
\boxed{1}.16632 \\
\times \quad 2 \\
\hline
\boxed{0}.33264 \\
\times \quad 2 \\
\hline
\boxed{0}.66528 \\
\times \quad 2 \\
\hline
\boxed{1}.33056
\end{array}
$$

etc.

11.001001...

Exercise – Convert ...

Decimal	Binary	Octal	Hexa-decimal
29.8			
	101.1101		
		3.07	
			C.82

Don’t use a calculator!

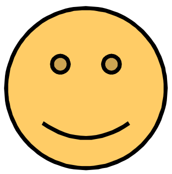
Skip answer

Answer

Exercise – Convert …

Answer

Decimal	Binary	Octal	Hexa-decimal
29.8	11101.110011…	35.63…	1D.CC…
5.8125	101.1101	5.64	5.D
3.109375	11.000111	3.07	3.1C
12.5078125	1100.10000010	14.404	C.82



4-Bit Binary Coded Decimal (BCD) Systems

- The 4-bit BCD system is usually employed by the computer systems to represent and process numerical data only.
- In the 4-bit BCD system, each digit of the decimal number is encoded to its corresponding 4-bit binary sequence.

Decimal digits	Weighted 4-bit BCD code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

4-Bit BCD Code

- Represent the decimal number 5327 in BCD code.

  4-bit BCD representation of decimal digit 5 is 0101
  4-bit BCD representation of decimal digit 3 is 0011
  4-bit BCD representation of decimal digit 2 is 0010
  4-bit BCD representation of decimal digit 7 is 0111

  Therefore, the BCD representation of decimal number 5327 is 0101001100100111.