Systems and Devices 1
Lec 2 : Data types

Before we get started ...

- Before we can design a data processing system we need to understand what data it will be processing.
  - How will information processed by the computer be represented?
    - Range, resolution, standard, format, encoding ...
  - Also, useful to understand the technology used to implement the system.
    - We can design an architecture independent of the implementation technology, but …
      - How data is stored internally / externally (capacity), accessed and processed (time), all have a significant impact on system performance i.e. some design decisions are technology dependant.

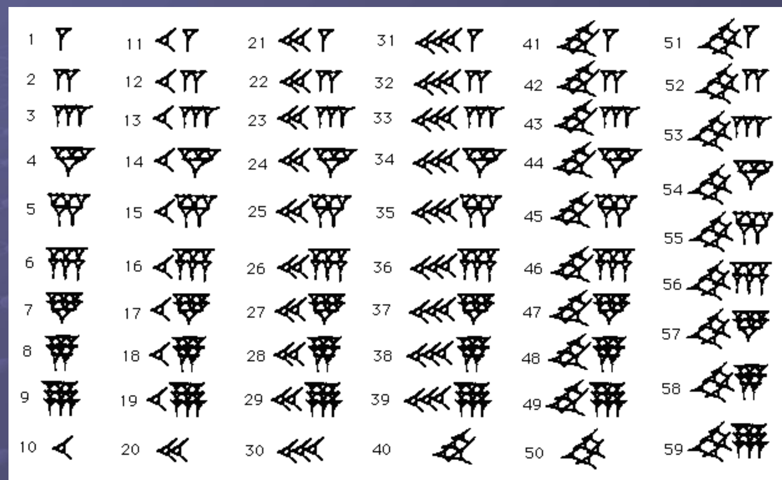
Numerical data

$$\{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \}$$

$$427_{10} = (10^2 \times 4) + (10^1 \times 2) + (10^0 \times 7)$$

- Q : How do we represent numbers?
  - In mathematical numbering systems, the ***base*** or ***radix*** is the number of unique digits, including zero, that a ***positional numeral system*** uses to represent numbers
  - The decimal system is most commonly used today, base ten, the maximum number a single digit can reach is 9, after which additional digits must be added to represent larger numbers

Radix

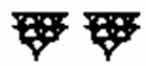


- Babylonian civilisation used base '60' (what's missing?)
  - Positional system, encoded using two basic symbols

Radix

$$427_{10} = 77_{60} = (60^2 \times 0) + (60^1 \times 7) + (60^0 \times 7)$$

$$60^2 = 3600 \quad : \quad 0$$
$$60^1 = 60 \quad : \quad 7 \quad (60^1 \times 7 = 420)$$
$$60^0 = 1 \quad : \quad 7 \quad (60^0 \times 7 = 7)$$

$77_{60} =$ 

- Converting a base 10 number to base 60
  - Same process as for base 10, but now each digit can represent the values of 0 – 59, missing 0 symbol :(
  - Result encoded using Babylonian symbols

Radix

Base 40 : { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39 }

$$427_{10} = (40^2 \times 0) + (40^1 \times 10) + (40^0 \times 27) = (10)(27)_{40}$$
$$= AR_{40} \quad (0 - 9, A - Z, \ldots)$$

Base 5 : { 0, 1, 2, 3, 4 }

$$427_{10} = (5^3 \times 3) + (5^2 \times 2) + (5^1 \times 0) + (5^0 \times 2)$$
$$= 3202_5$$

- Working in different number bases
  - Greater than base 10 and less than base 10.

Radix

Base 16 : { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 }
{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F }

256 16 1

$$123_{10} = (16^2 \times \square) + (16^1 \times 7) + (16^0 \times B)$$

Base 8 : { 0, 1, 2, 3, 4, 5, 6, 7 }

64 8 1

$$\square_{10} = (8^2 \times 1) + (8^1 \times 7) + (8^0 \times 3)$$

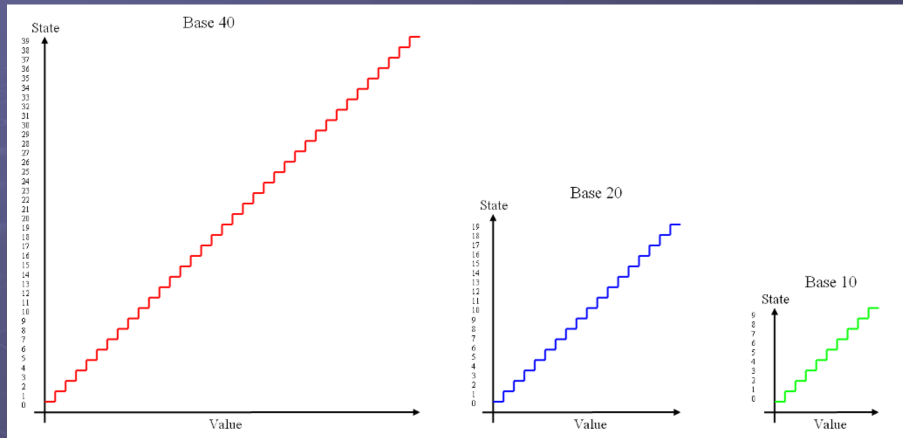
Base 4 : { 0, 1, 2, 3 }

64 16 4 1

$$123_{10} = (4^3 \times 1) + (4^2 \times 3) + (4^1 \times 2) + (4^0 \times \square)$$

- Quick quizzz

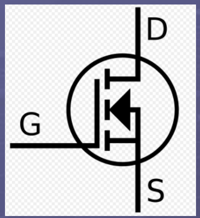
Radix

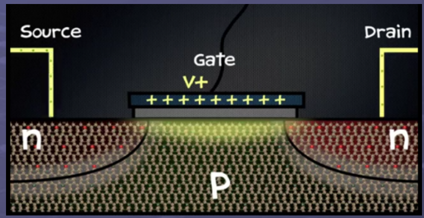


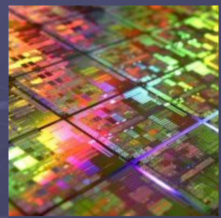
- Q : What is the best way to represent numbers in a computer?
  - Moving to a higher base : less digits, more symbols
  - Moving to a lower base : more digits, less symbols

Slide 9

Technology



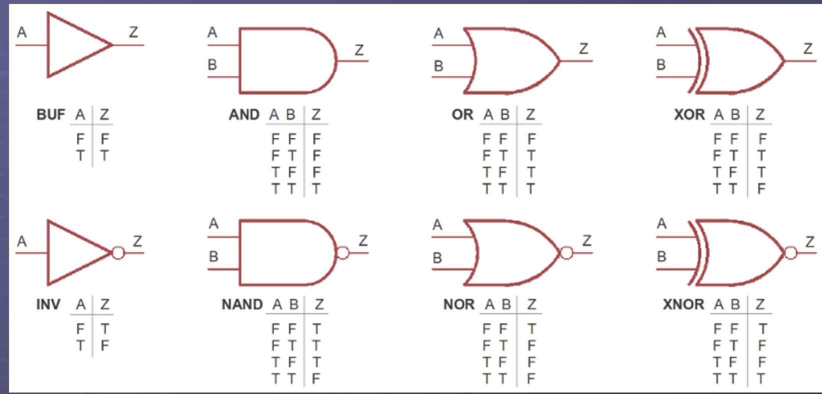




- Q : What is the best way (base) to represent numbers in a computer?
- A : It depends. What base is the most efficient in terms of processing (time) and storage (capacity) for a given technology.
- Technology most commonly used today is based on the transistor : Metal Oxide Semiconductors (MOS).
- Q : If a technology has two stable operating states what base should we use?

Slide 10

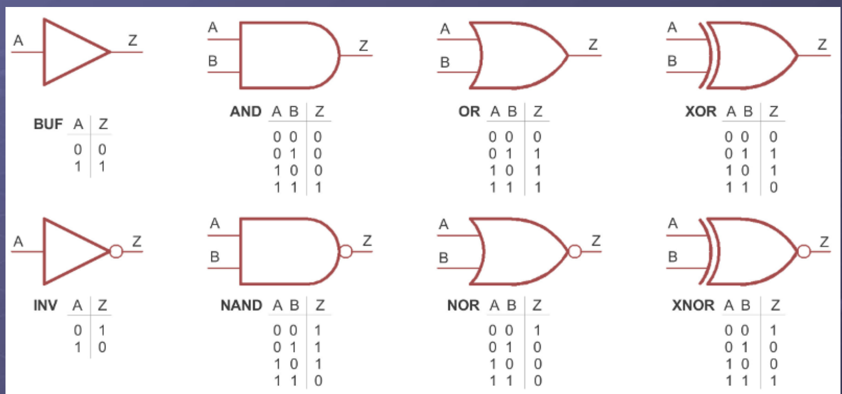
Technology



- Q : How can we process base-2 data?
  - Luckily we already have a branch of mathematics to do this : Boolean algebra.
  - We can encode a 1 as TRUE and 0 as FALSE, but …

Slide 11

Technology

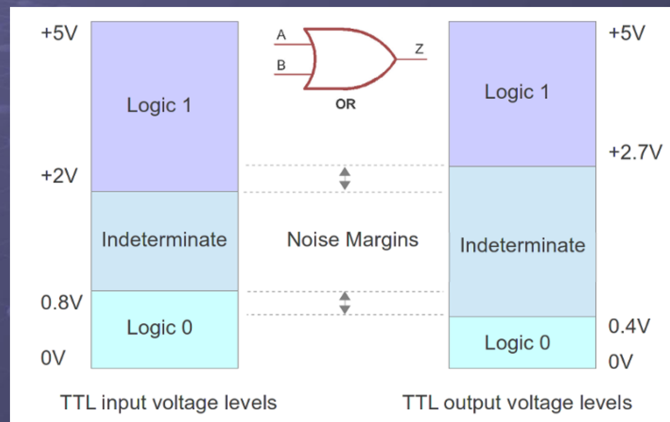


- Q : How can we process base-2 data?
  - Luckily we already have a branch of mathematics to do this : Boolean algebra.
  - We can encode a 1 as TRUE and 0 as FALSE, but …

Slide 12

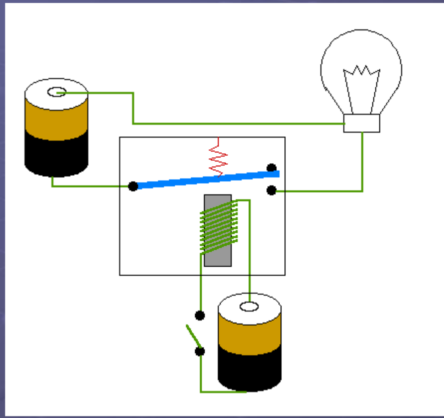
Technology

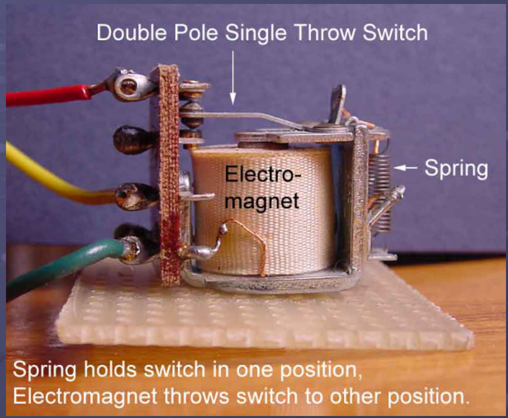
- An advantage of a base-2 (binary) representation is that it minimises the number of symbols (states) a technology needs to implement.



- Q : How can we communicate base-2 data
  - Another advantage of having less symbols is noise immunity

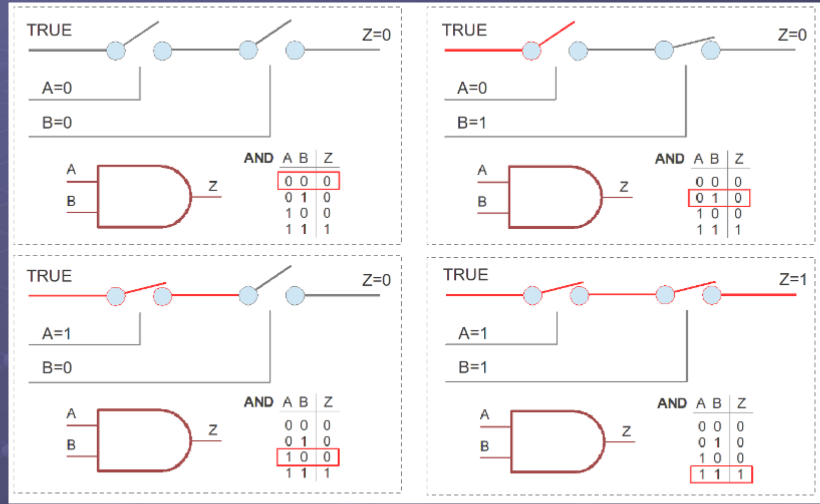
Relay logic





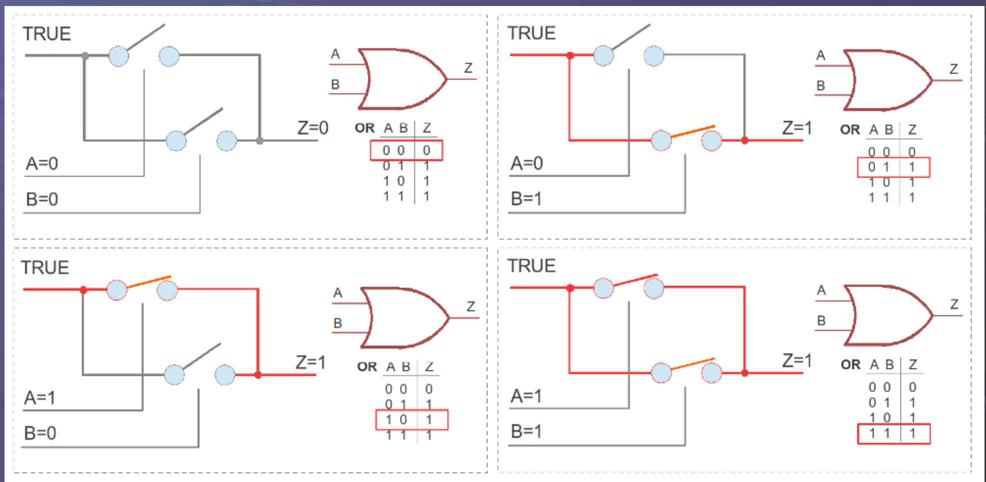
- To explain Boolean logic gate we will use ladder logic based on relays
  - Voltage controlled switch the same as a transistor, just bigger

Logic gates



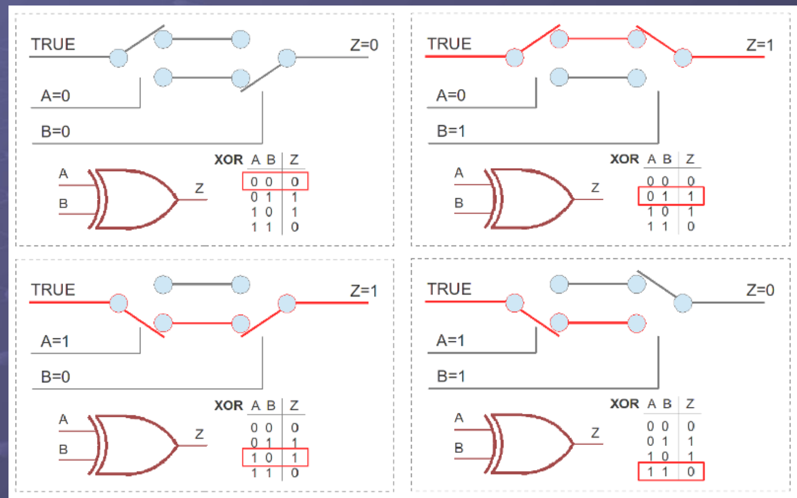
- AND gate

Logic gates



- OR gate

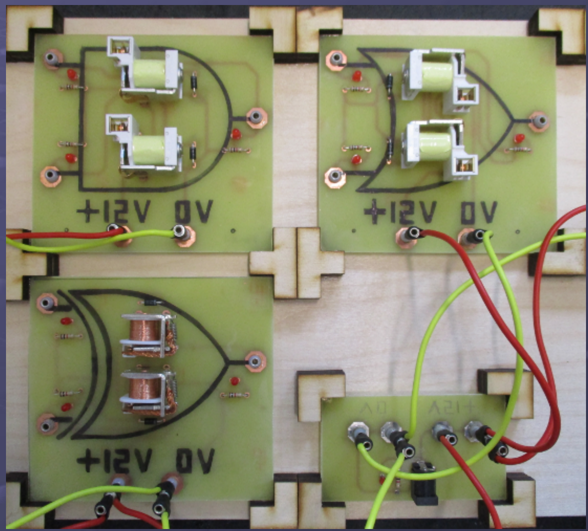
Logic gates



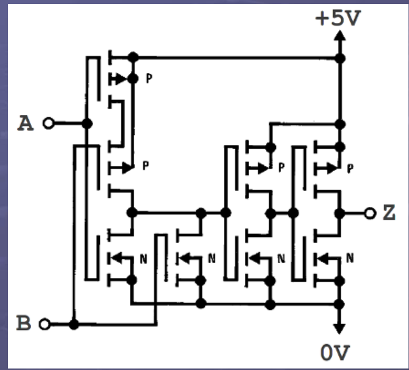
- XOR gate

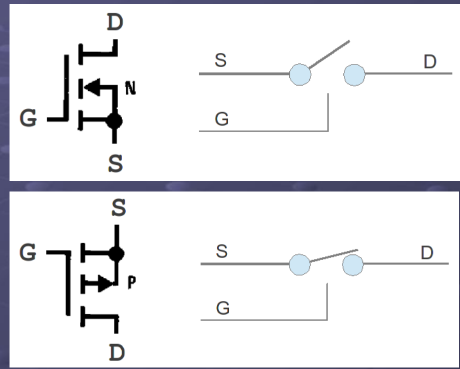
Demo : relay logic

- The three core logic gates:
  - AND
  - OR
  - XOR
- Using only these gates we can build a computer.
  - INV can be made from an XOR gate.



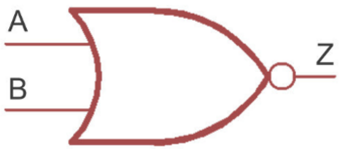
Technology



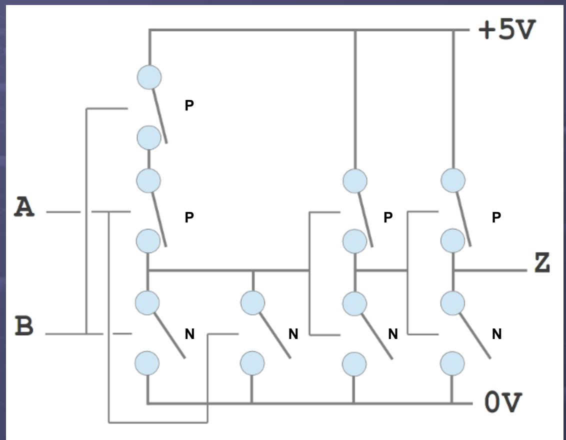


- Complementary Metal Oxide Semiconductors (CMOS)
  - P-channel : equivalent to a normally closed relay
    - Logic 1 on Gate opens contacts
  - N-channel : equivalent to a normally open relay
    - Logic 1 on Gate closes contacts

Technology



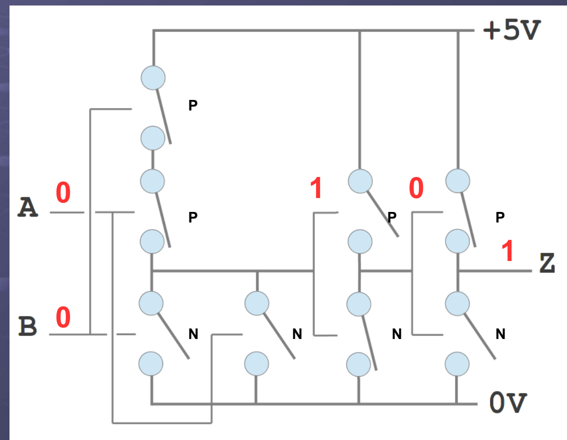
NOR A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0



- NOR gate : 4001 integrated circuit (IC)
  - Output Z=1 when A=B=0

Technology

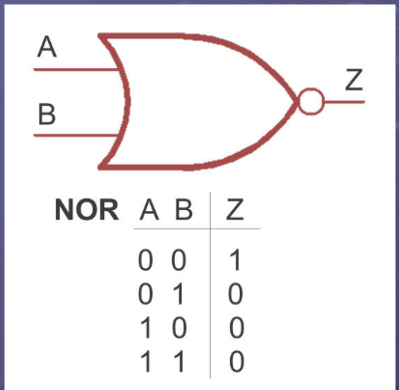
NOR A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0

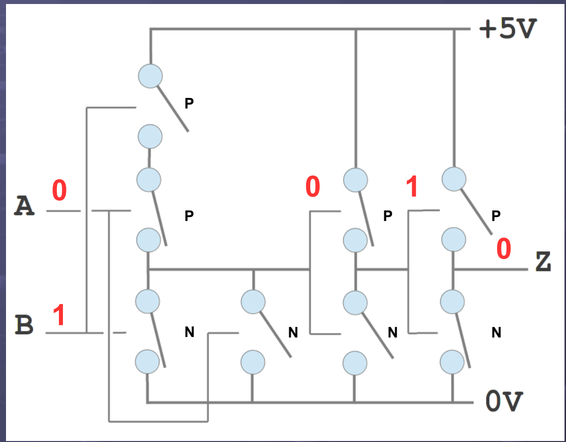


- NOR gate : 4001 integrated circuit (IC)
  - Output Z=1 when A=B=0

Slide 21

Technology

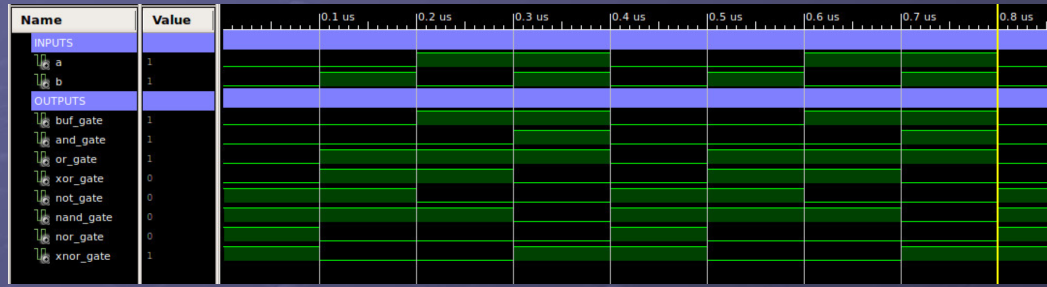


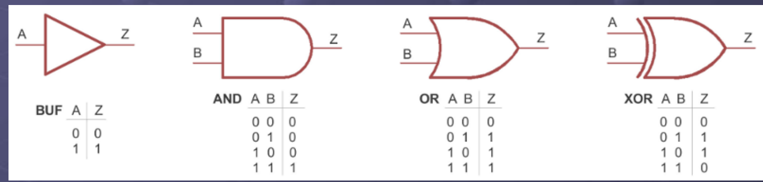


- NOR gate : 4001 integrated circuit (IC)
  - Output Z=1 when A=B=0

Slide 22

Example : Logic.zip





- Analyse of logic gates is normally performed through simulation or waveform traces

Slide 23

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

- Converting a base 10 number to base 2

Slide 24

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

A number is made up of Binary digITS or Bits

- Converting a base 10 number to base 2

Slide 25

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

<u>Bit positions</u>
Eight bit number
Bit 7 to Bit 0

- Converting a base 10 number to base 2

Slide 26

Key skills : working in base 2

<u>IMPORTANT</u>
Always remember to start counting from ZERO
The first bit is not ONE

- Converting a base 10 number to base 2

Slide 27

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

Eight bits is called a Byte

- Converting a base 10 number to base 2

Slide 28

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

Four bits is called a nibble

- Converting a base 10 number to base 2

Slide 29

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate r

Least significant bit position (LSB)

- Converting a base 10 number to base 2

Slide 30

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Re								

Most significant bit position (MSB)

- Converting a base 10 number to base 2

Slide 31

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result								

Intermediate results

- Converting a base 10 number to base 2

Slide 32

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0							

Intermediate results

$$\begin{array}{r} 99 \\ -128 \\ \hline -29 \end{array}$$

- Converting a base 10 number to base 2

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1						

Intermediate results

99	99
-128	-64
-29	35

- Converting a base 10 number to base 2

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1					

Intermediate results

99	99	35
-128	-64	-32
-29	35	3

- Converting a base 10 number to base 2

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1	0				

Intermediate results

99	99	35	3
-128	-64	-32	-16
-29	35	3	-13

- Converting a base 10 number to base 2

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1	0	0			

Intermediate results

99	99	35	3	3
-128	-64	-32	-16	-8
-29	35	3	-13	-5

- Converting a base 10 number to base 2

Slide 37

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1	0	0	0		

Intermediate results

99	99	35	3	3	3
-128	-64	-32	-16	-8	-4
-29	35	3	-13	-5	-1

- Converting a base 10 number to base 2

Slide 38

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1	0	0	0	1	

Intermediate results

99	99	35	3	3	3	3
-128	-64	-32	-16	-8	-4	-2
-29	35	3	-13	-5	-1	1

- Converting a base 10 number to base 2

Slide 39

Key skills : working in base 2

Convert decimal value 99_{10} to base 2

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	0	1	1	0	0	0	1	1

Intermediate results

99	99	35	3	3	3	3	1
-128	-64	-32	-16	-8	-4	-2	-1
-29	35	3	-13	-5	-1	1	0

- Converting a base 10 number to base 2

Slide 40

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

- Converting a base 2 number to base 10

Slide 41

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0
+128
128

- Converting a base 2 number to base 10

Slide 42

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128
+128	+64
128	192

- Converting a base 2 number to base 10

Slide 43

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192
+128	+64	+ 0
128	192	192

- Converting a base 2 number to base 10

Slide 44

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192	192
+128	+64	+ 0	+ 0
128	192	192	192

- Converting a base 2 number to base 10

Slide 45

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192	192	192
+128	+64	+ 0	+ 0	+ 8
128	192	192	192	200

- Converting a base 2 number to base 10

Slide 46

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192	192	192	200
+128	+64	+ 0	+ 0	+ 8	+ 4
128	192	192	192	200	204

- Converting a base 2 number to base 10

Slide 47

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192	192	192	200	204
+128	+64	+ 0	+ 0	+ 8	+ 4	+ 0
128	192	192	192	200	204	204

- Converting a base 2 number to base 10

Slide 48

Key skills : working in base 2

Convert binary value 11001101_2 to base 10

Bit	7	6	5	4	3	2	1	0
Value	128	64	32	16	8	4	2	1
Result	1	1	0	0	1	1	0	1

Intermediate results

0	128	192	192	192	200	204	204
+128	+64	+ 0	+ 0	+ 8	+ 4	+ 0	+ 1
128	192	192	192	200	204	204	205

- Converting a base 2 number to base 10

Key skills : working in base 2

Base 2 : { 0, 1 }

	32	16	8	4	2	1

$33_{10} = (2^5 \times 1) + (2^4 \times 0) + (2^3 \times 0) + (2^2 \times 0) + (2^1 \times \square) + (2^0 \times \square)$

$\square_{10} = (2^5 \times 1) + (2^4 \times 1) + (2^3 \times 0) + (2^2 \times 0) + (2^1 \times 1) + (2^0 \times 0)$

	128	64	32	16	8	4	2	1
$123_{10} =$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$
$\square_{10} =$	1	0	1	0	1	0	1	0

- Quick quizzz

Key skills : working in base 2

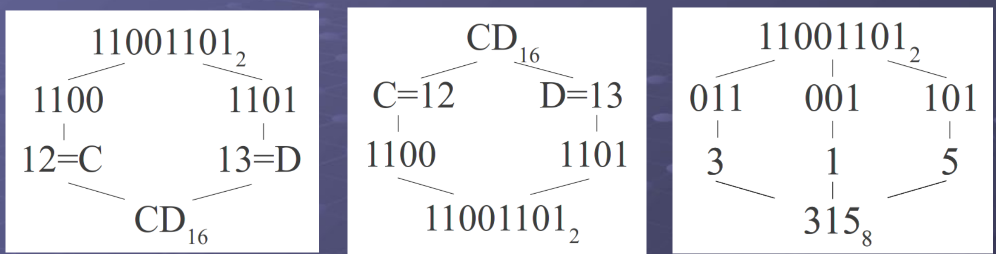
Dec	Hex	Bin
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111

Dec	Hex	Bin
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Dec	Octal	Bin
0	0	000
1	1	001
2	2	010
3	3	011
4	4	100
5	5	101
6	6	110
7	7	111

- Other commonly used bases
  - Hexadecimal : base 16, binary string split into nibbles
    - 205_{10} = 0xCD or CD_{16}
  - Octal : base 8, binary string split into triples.
    - 205_{10} = 0315 or 315_8

Key skills : working in base 2



Digit	16^2	16^1	16^0
Value	256	16	1
Number	0	C	D

Result = $(0\times256) + (12\times16) + (13\times1) = 205_{10}$

- Converting a base 2 number to/from base 8, 10 and 16

Summary

- Key concepts :
  - Number bases
    - Positional numeral system uses to represent numbers.
    - Working in different number bases: 2, 8, 10 and 16.
  - Binary number representation
    - Base 2, bit, { 0, 1 }, byte, nibble, MSB, LSB.
    - Easy to implement using electronic circuits (switch logic).
      - Less symbols
    - Conversion to and from decimal representations.
  - Boolean logic
    - Basic operations : INV (NOT), AND, OR, XOR.
    - Ladder logic, Circuit symbols.