數位系統回顧

陳培殷

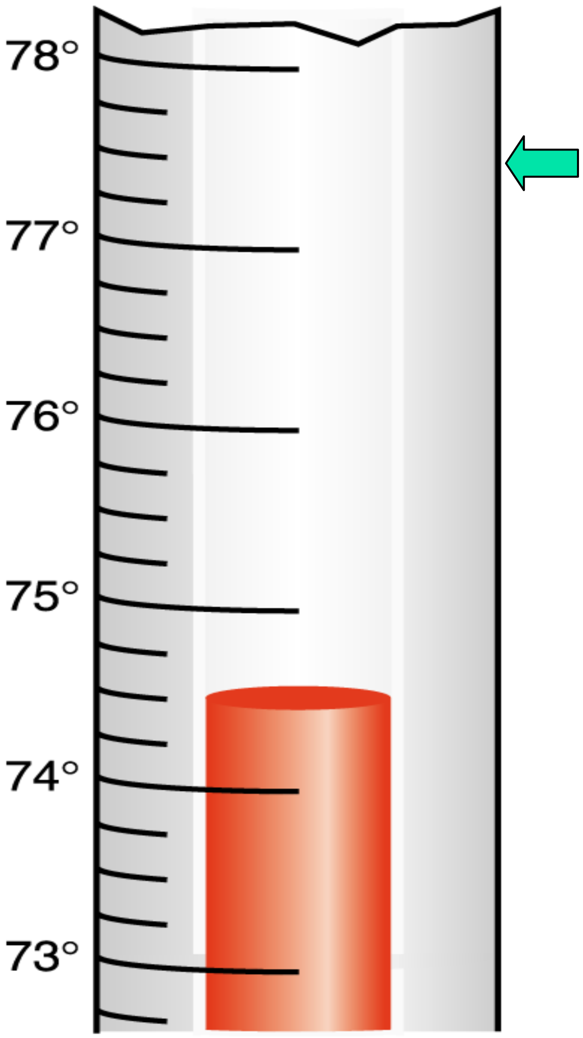
Analog and Digital Information (1/6)

- Information (資訊) can be represented in one of two ways: **analog** (類比) or **digital** (數位).

Analog data A continuous (連續性) representation, analogous to the actual information it represents.

Digital data A discrete (離散) representation, breaking the information up into separate elements. A mercury thermometer is an analog device. The mercury rises in a continuous flow in the tube in direct proportion to the temperature.

Analog and Digital Information (2/6)



Analog device(類比裝置) 水銀溫度計

74.568.. degree Fahrenheit

A mercury thermometer continually rises in direct proportion to the temperature

Digitize(數字化): The act of breaking information down into discrete pieces

Digital device: (數位裝置) 耳溫槍

If the resolution is 0.01 degree, the answer is 74.56 or 74.57

If the resolution is 0.1 degree, the answer is 74.5 or 74.6

Analog and Digital Information (3/6)

- Natural world is continuous and infinite

  There are infinite real numbers between 5.494 and 5.495

5.4944, 5.494449, 5.49442357, 5.4949999, ……

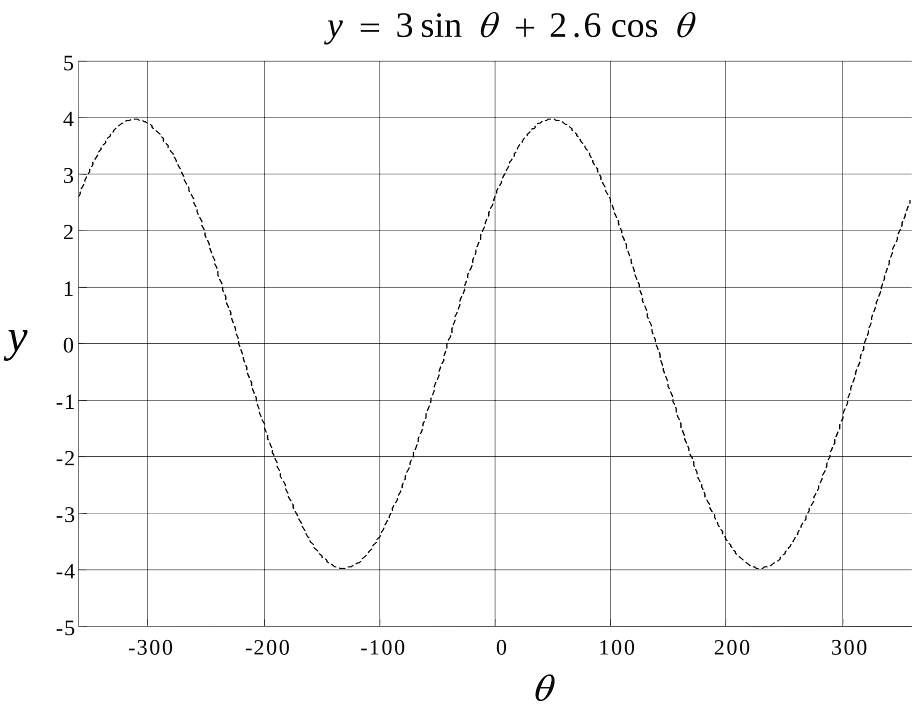
- Hardware (digital system) is finite (有限的).

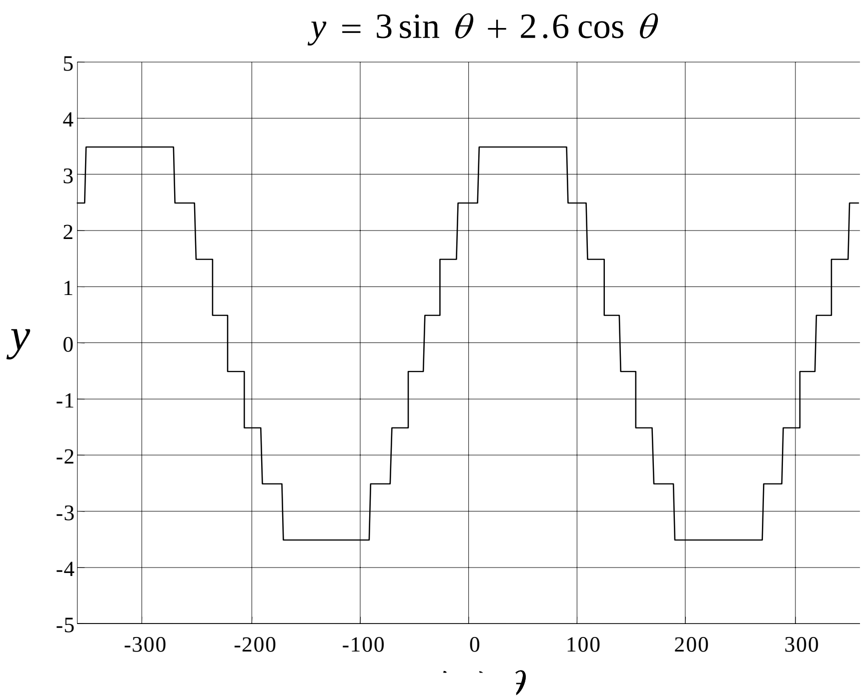
  Computer memory and other hardware devices have only so much room to store and manipulate a certain amount of data. The goal, is to represent enough of the world to satisfy our computational needs and our senses of sight and sound.

Analog and Digital Information (4/6)

Digitize *y* with 8 discrete steps.

-3.5, -2.5, -1.5, -0.5, 3.5, 2.5, 1.5, 0.5





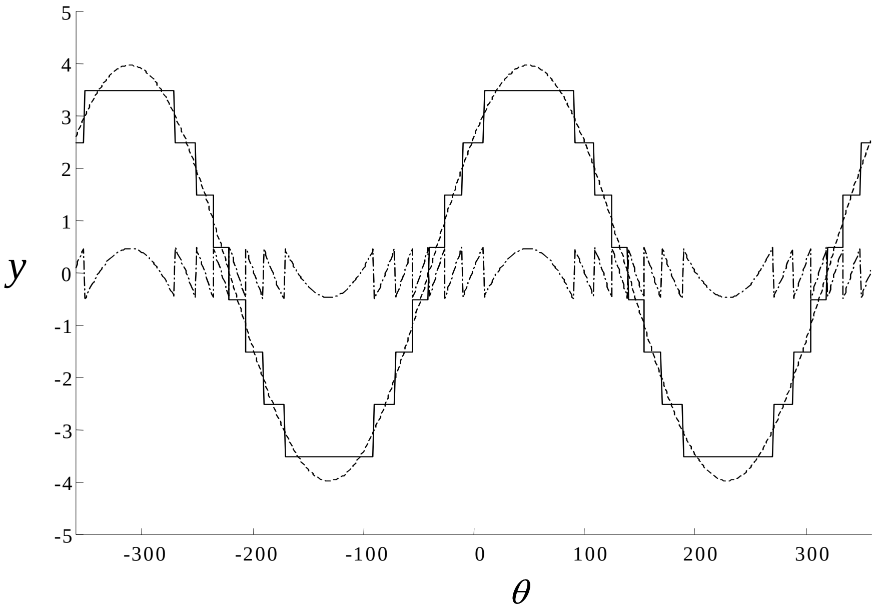
Analog

-0.5<error<0.5

Digital

Analog and Digital Information (5/6)

$$y = 3\sin\theta + 2.6\cos\theta$$

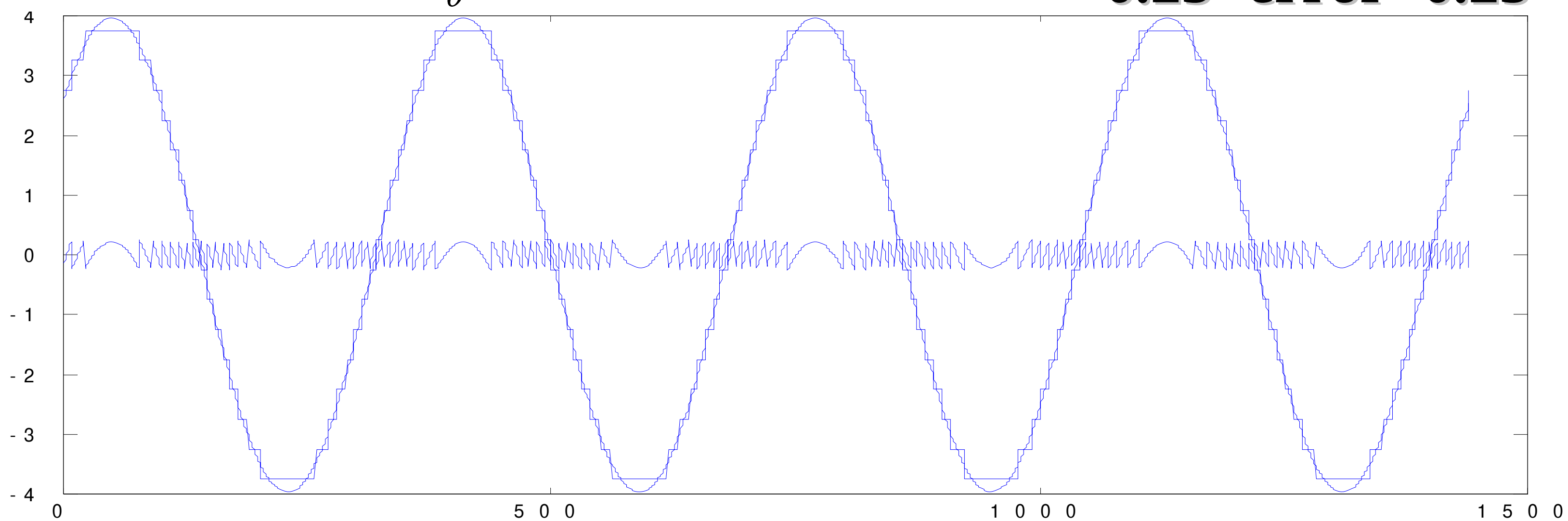


Digitize *y* with 8 discrete steps.

-0.5<error<0.5

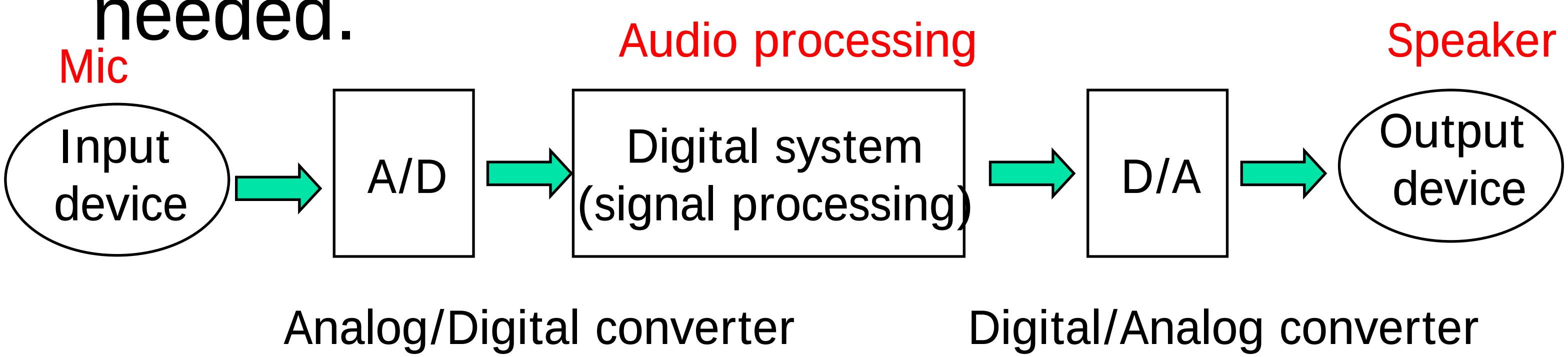
Digitize *y* with 16 discrete steps.

-0.25<error<0.25



Analog and Digital Information (6/6)

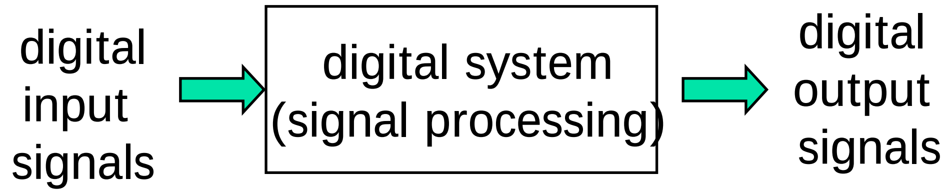
- Most sensors are analog devices.
- Computers, cannot work well with analog information. So we **digitize** information by breaking it into pieces and representing those pieces separately.
- Therefore an analog to digital converter is needed.



Traditional TV input signals

Digital Systems

- **Present technology period -- Digital Age**
- **Digital systems**
  - **Digital phone**
  - **Digital television**
  - **Digital camera**
  - **Electronic calculators, PDA's**

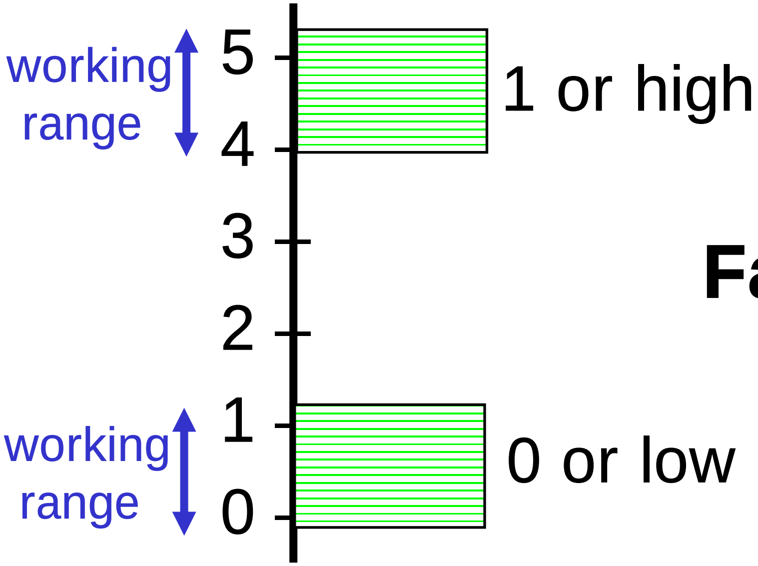


- **Digital computers**
  - **Follow a sequence of instructions (--program)**
  - **General-purpose for information processing**

Signal

- **Discrete elements of information are represented by physical quantity.**
- **Electrical signals such as voltages and currents are the most common.**
- **Most digital systems use two discrete values (binary). It is easy to realized with the current or voltage.**
  - **digits 0 and 1**
  - **False (F) and True (T)**
  - **Low (L) and High (H)**
  - **On and Off**

Example: Voltage

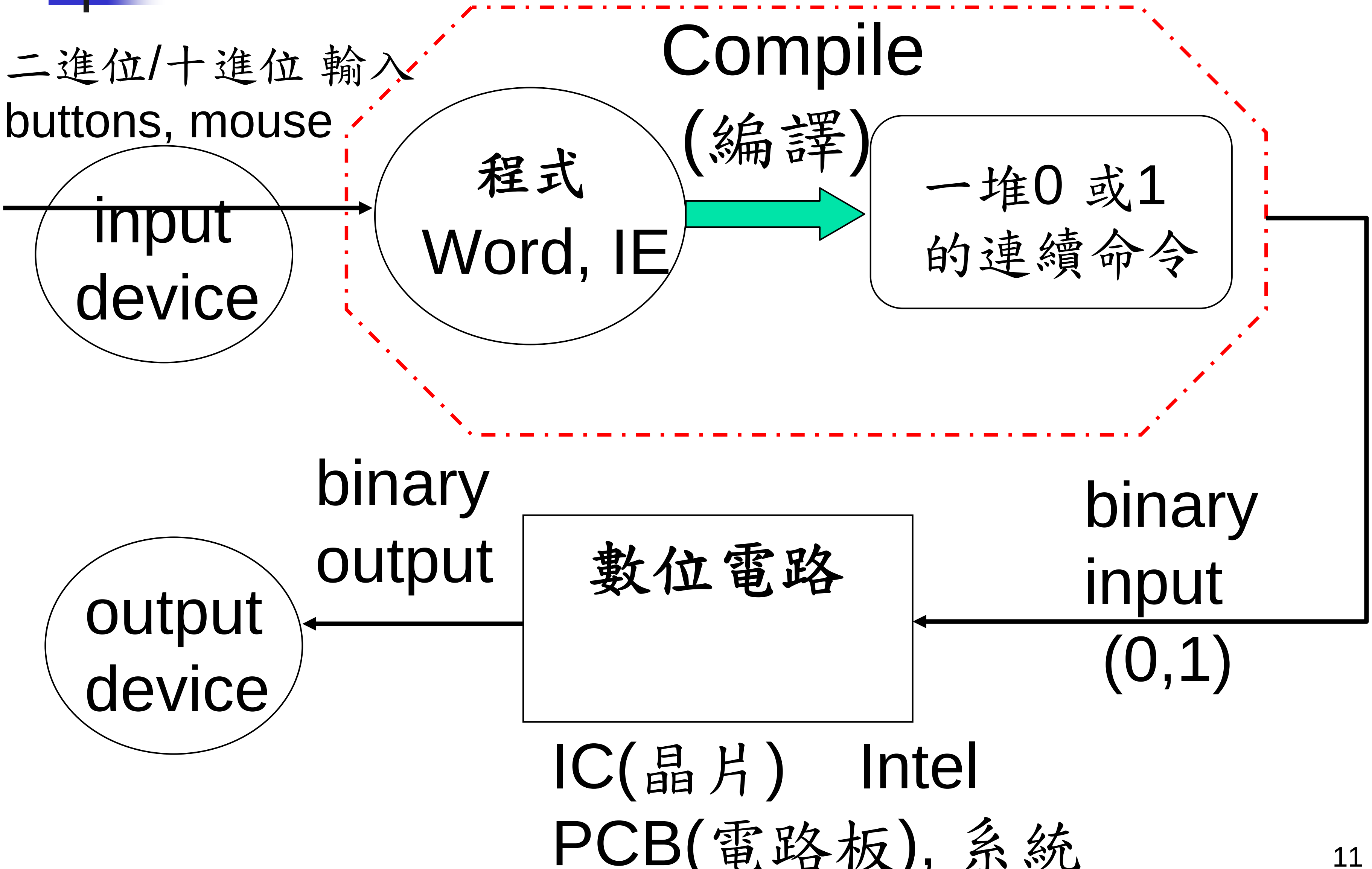


Fault tolerance is allowable.

Why binary ?   Almost all digital systems use binary.

Distinguishing 2 levels is easier and less distortion than 10 levels for real-world electrical devices

Overview of Digital System



Computers and Electricity

- Consumer electronic devices (消費電子裝置) use binary

  range of 0~1 volt is “low” -- binary 0

  range of 2~5 volt is “high” -- binary 1

  Now, many devices use different voltage for power saving.

  Higher voltage and faster frequency need more Power (P)

- **Gate (邏輯閘)** A device that performs a basic operation on electrical signals, accepting one or more input signals and producing a single output signals
- **Circuits (電路)** A combination of interacting gates designed to accomplish a specific logical function (the output of a gate often serves as an input for one or more other gates)

AND Gate

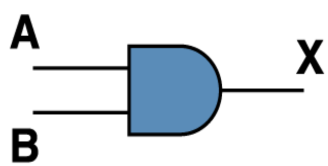
- An AND gate （及閘 或 且閘）accepts two input signals
- If the two input values for an AND gate are both 1, the output is 1; otherwise, the output is 0

Boolean Expression

$$X = A \cdot B$$

$$0 \cdot 0 = 0$$
$$0 \cdot 1 = 0$$
$$1 \cdot 0 = 0$$
$$1 \cdot 1 = 1$$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Input output

OR Gate (或閘)

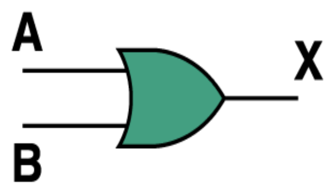
- If the two input values are both 0, the output value is 0; otherwise, the output is 1

Boolean Expression

$$X = A + B$$

$$0 + 0 = 0$$
$$0 + 1 = 1$$
$$1 + 0 = 1$$
$$1 + 1 = 1$$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Input | output

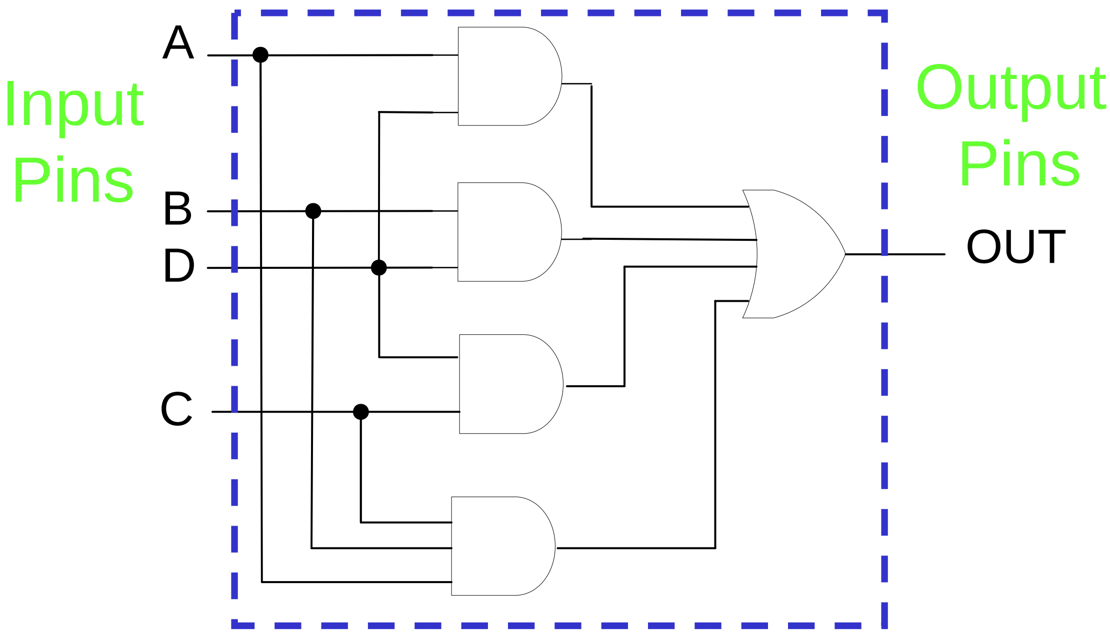
Various representations of a OR gate

A Simple Circuit – Alarm

Assume that four persons might come. Alarm is activated when (1) more than three persons come or (2) the fourth person come together with other persons

A	B	C	D	Out
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Out = AD + BD + CD + ABC



What are inputs? What are outputs?

What function? How to design?

Integrated Circuits (IC)

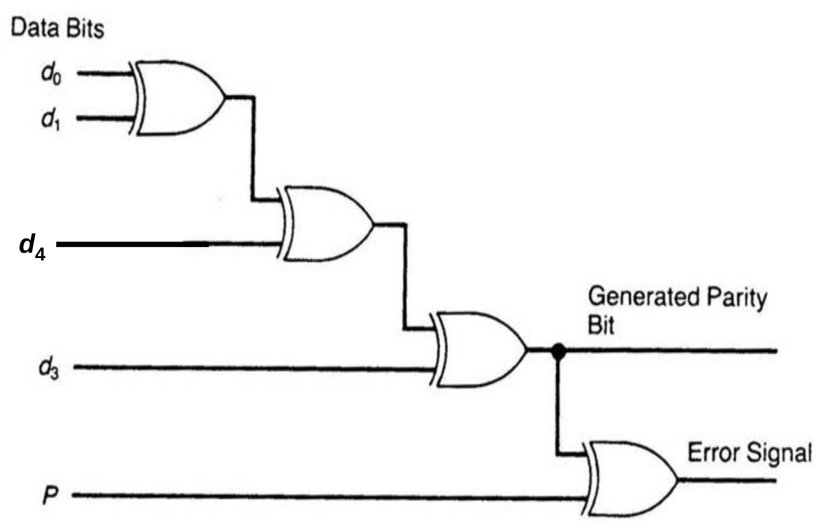
- **Integrated circuit(積體電路)** (also called a *chip*) A piece of silicon(矽) on which multiple gates (circuits) have been embedded

  These silicon pieces are mounted on a plastic or ceramic package with pins along the edges that can be soldered onto circuit boards or inserted into appropriate sockets

  =>A circuit implemented with semiconductor (silicon) technology.

Combinational Circuit (組合電路)

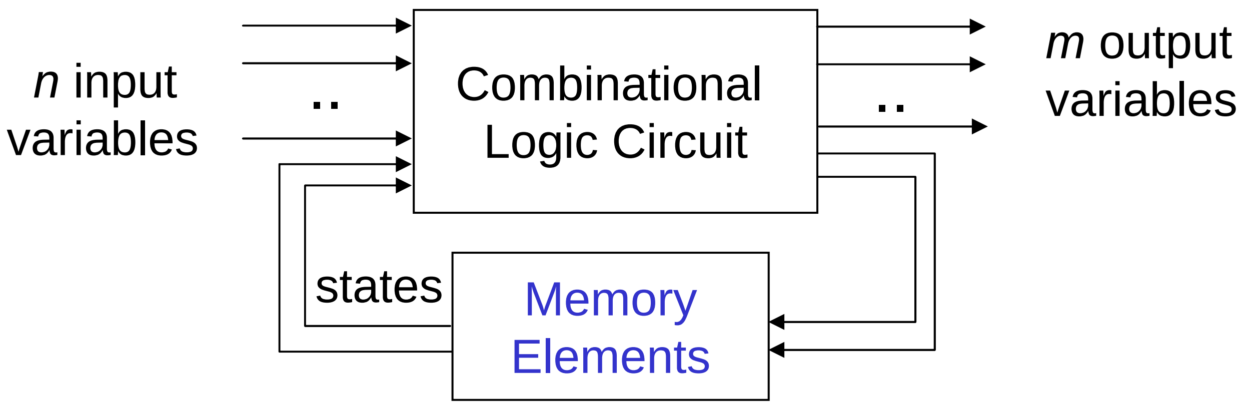
A combinational circuit consists of logic gates whose outputs at any time are determined *directly from the present combination of inputs* without regard to previous inputs.



No clock !

Sequential Circuit (循序電路1/2)

A sequential circuit is a system whose outputs at any time are determined *from the present combination of inputs and the previous inputs or outputs*.



- Sequential components contain memory elements
- The output values of sequential components depend on the input values and the values stored in the memory elements
- Example: Ring counter that starts the answering machine after 4 rings

Sequential Circuit (2/2)

Sequential components can be: asynchronous or synchronous

Asynchronous sequential circuit:

Change their states and outputs whenever a change in inputs occurs

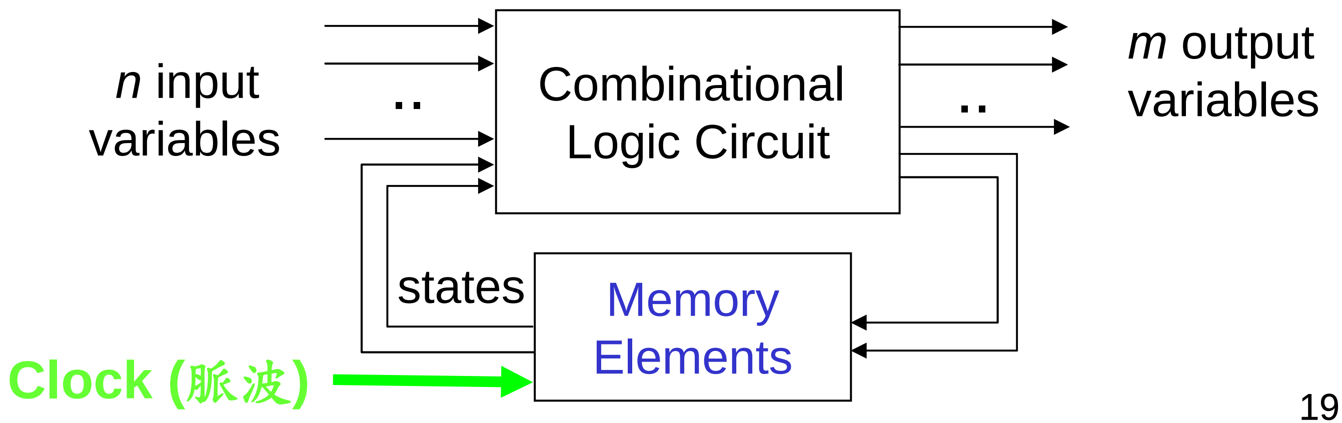
Synchronous sequential circuit:

Change their states and outputs at fixed points of time (specified by clock signal)

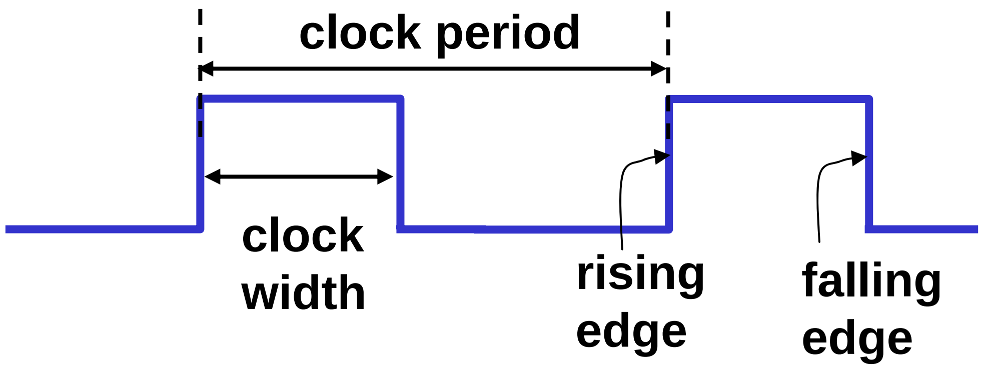
Most circuits are synchronous circuits (easy and tool-supportable).

Synchronous storage components store data and perform some simple operations. Synchronous storage components include:

(1) registers
(2) counters
(3) register files
(4) memories
(5) queues
(6) stacks



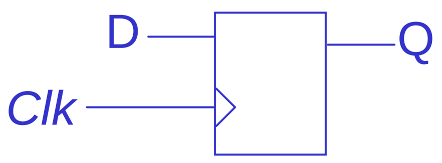
Clock Period



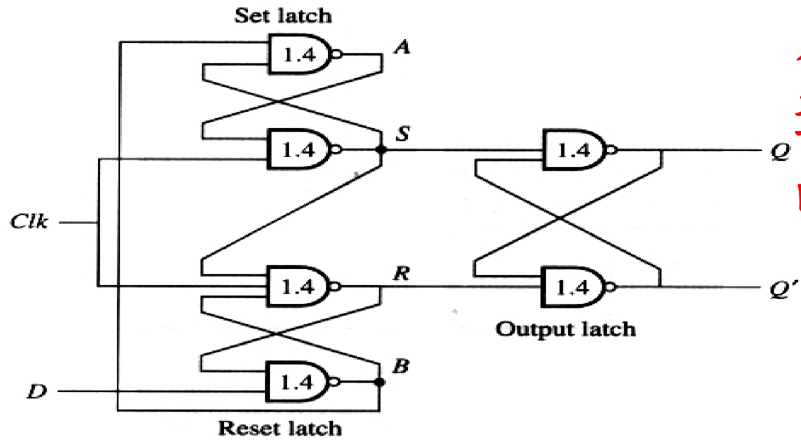
- Clock period (measured in micro or nanoseconds) is the time between successive transitions in the same direction
- Clock frequency (measured in MHz or GHz) is the reciprocal of clock period
- Clock width is the time interval during which clock is equal to 1
- Duty cycle is the ratio of the clock width and clock period
- Clock signal is active high if the changes occur at the rising edge or during the clock width. Otherwise, it is active low

Storage Component - D Flip-Flop

D flip-flop

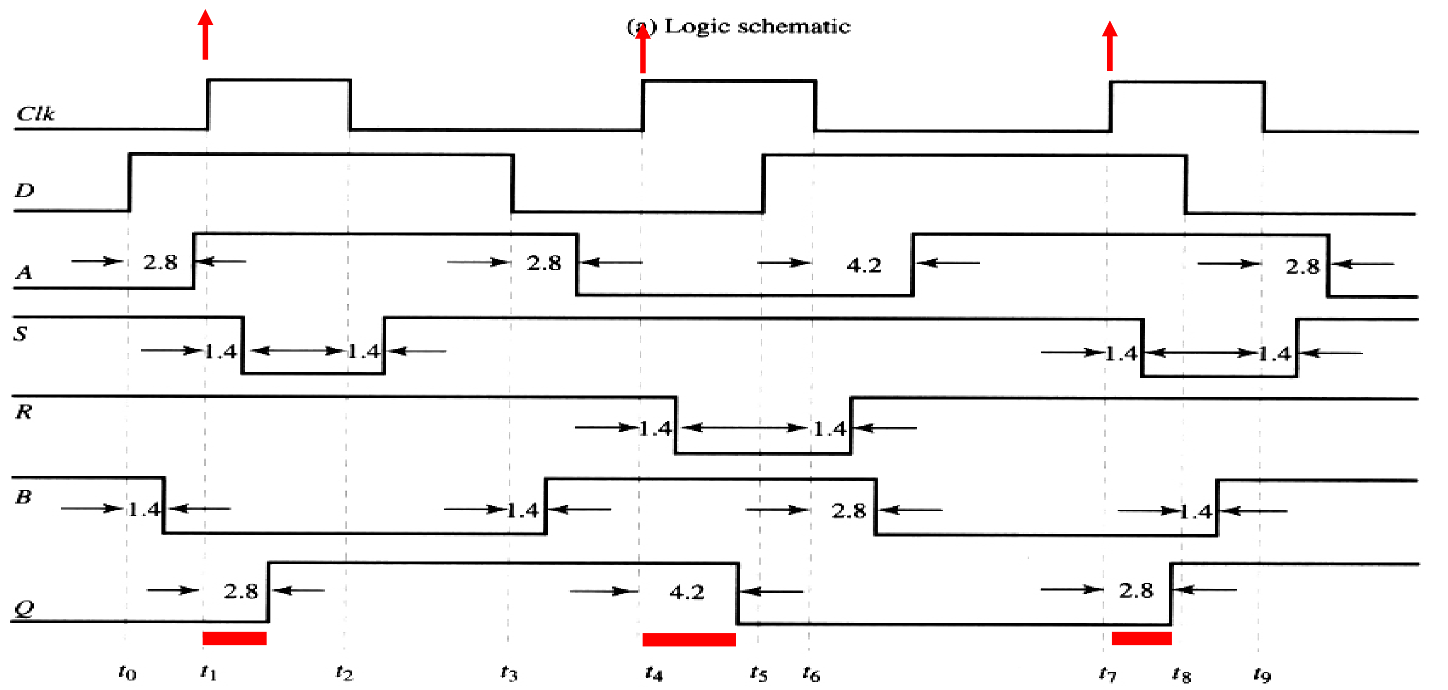


Edge-triggered flip-flop



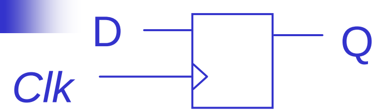
(a) Logic schematic

只有當clk上升邊緣到達時,D的值才會放入Q

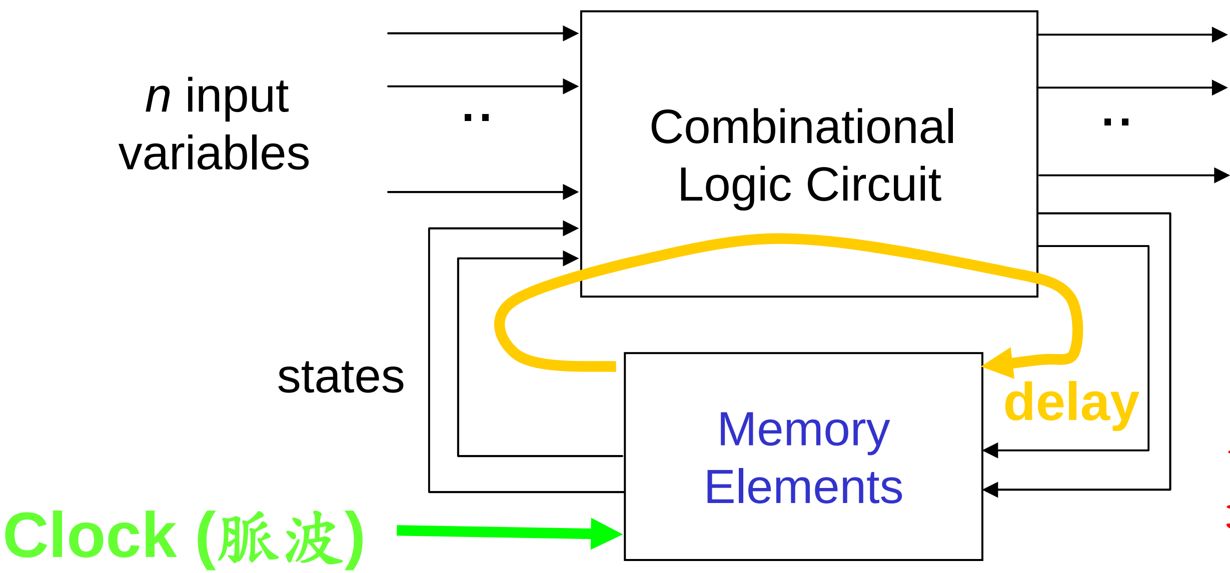


(b) Timing diagram

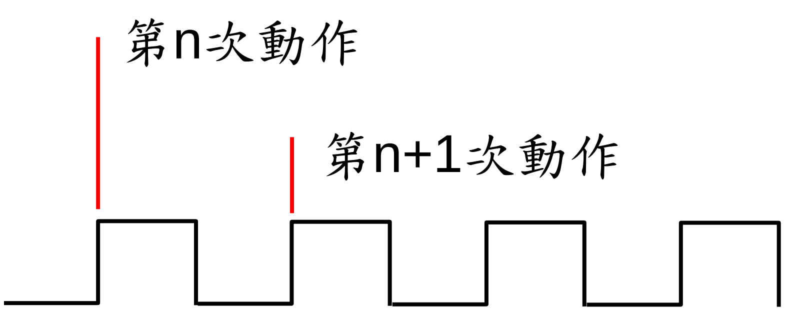
Synchronous Sequential Circuit



正緣觸發 vs. 負緣觸發



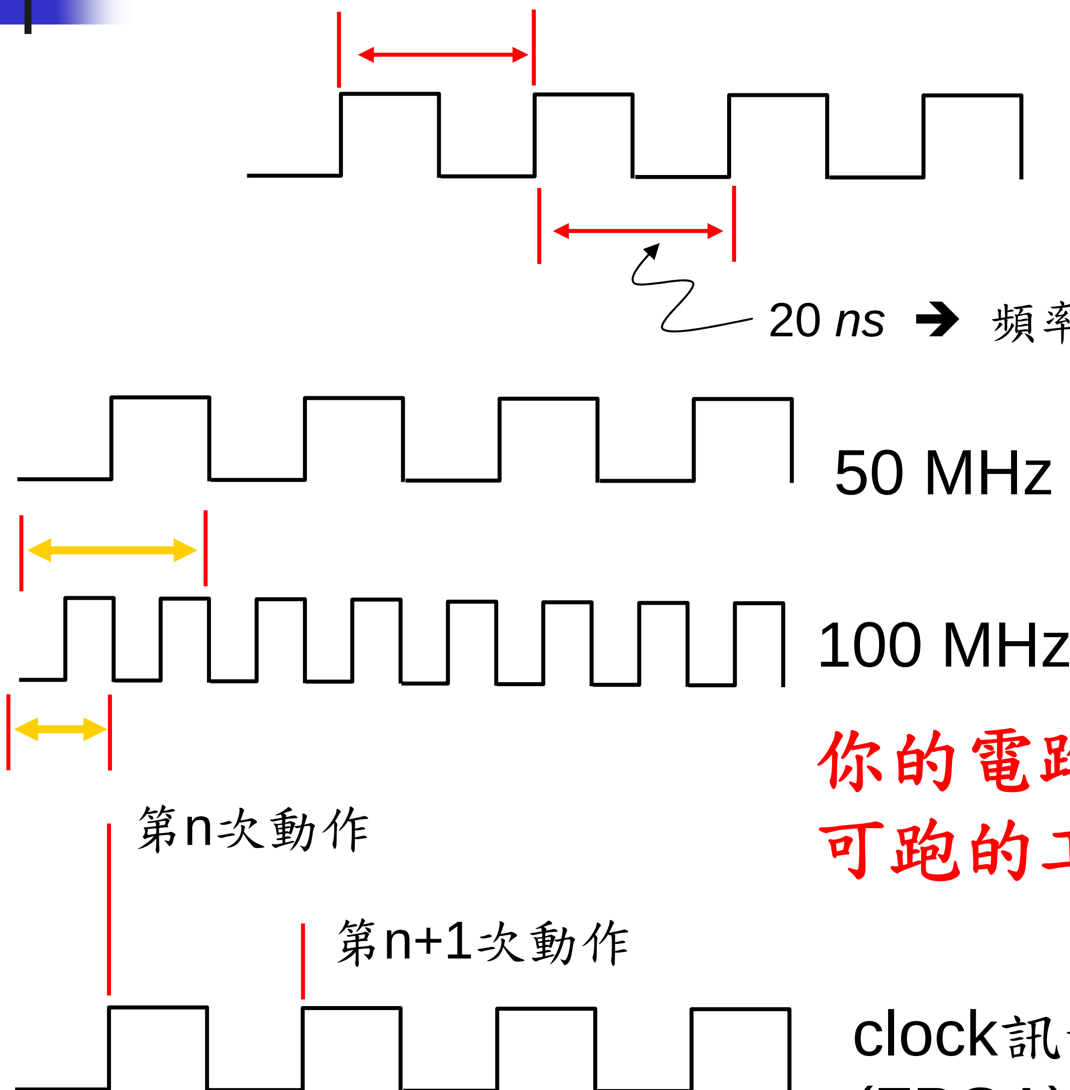
只有當clk上升邊緣到達時,D的值才會放入Q



每次clock正緣來完成一個運算動作➔ 可得出電路(IC)的工作速率(頻率)

Clock

週期性方波



20 *ns* ➔ 頻率= $\frac{1}{20\,ns} = 50\text{ MHz}$

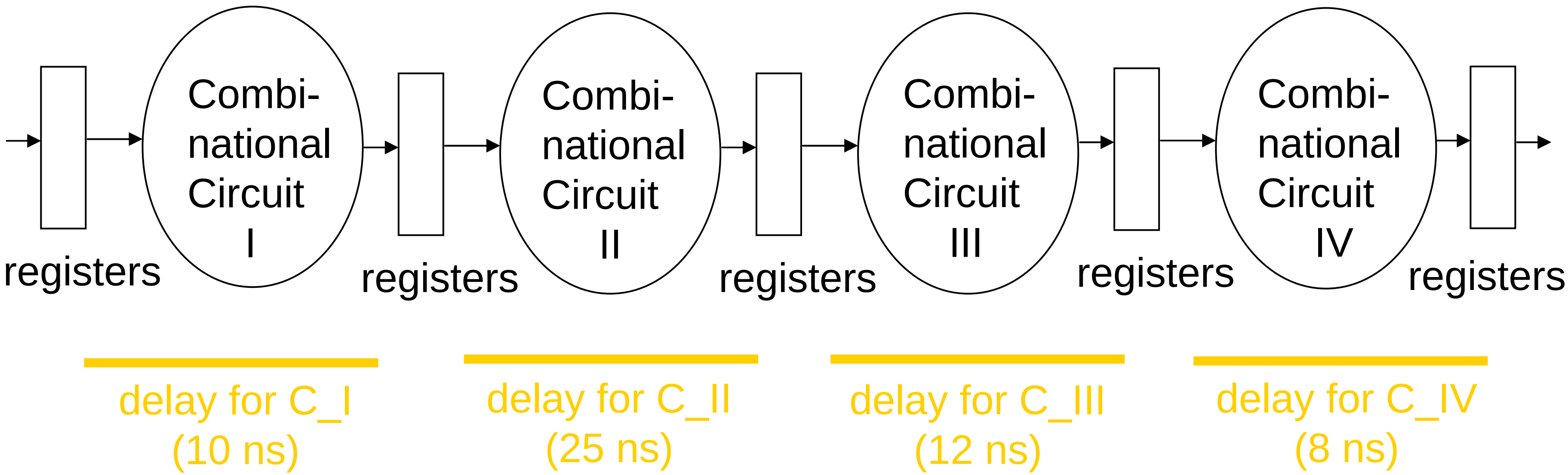
50×10^6 次/秒

你的電路所需的delay愈少
可跑的工作頻率愈高(愈快)

clock訊號會進入電路中(FPGA)以同步方式控制其行為

Clock Period/Clock Rate

How to decide the clock period in a system?



1. Find out the longest delay among combinational circuits C_I, C_II, C_III and C_IV.
2. The longest delay is named as the critical path (here is 25 ns).
3. The clock period can be set as little longer than the critical path, why?
4. $$\text{clock frequency} = \frac{1}{\text{clock period}} \quad \left(\text{here } \frac{1}{25\text{ns}} = \frac{1}{25\times 10^{-9}} = 40 \text{ MHz}\right)$$

IC Design flow

IC design flow

Full Custom

Semi Custom (Cell-Based Design)

ASIC

a. Standard Cells:

TSMC or UMC -cells

b. FPGA or PLD Programmable logic:

Xilinx, Altera or Actel -cells

Full (Fully) Custom Design:

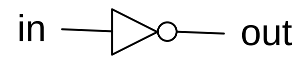
a. For analog circuits and digital circuits requiring custom optimization
b. Gates, transistors and layout are designed and optimized by the engineer

Semi Custom Design:

a. For larger digital circuits
b. Real gates, transistors and layout are synthesized and optimized by related software tools
c. Realization with hardware description language (HDL) such as VHDL and Verilog

Full Custom Design (全客戶式設計)

CMOS Inverter



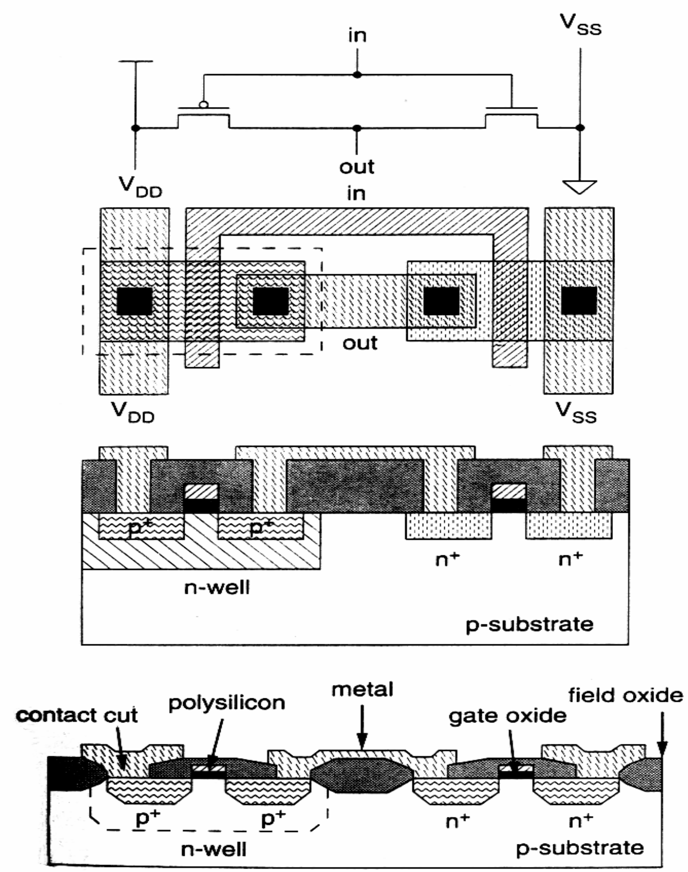
a. Long design cycle

(transistors and wires)

b. No CPLD or FPGA solutions

c. Analog circuit

d. Smaller digital circuit



done by chip designer

masking

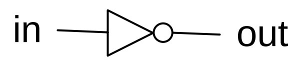
done by TSMC, UMC

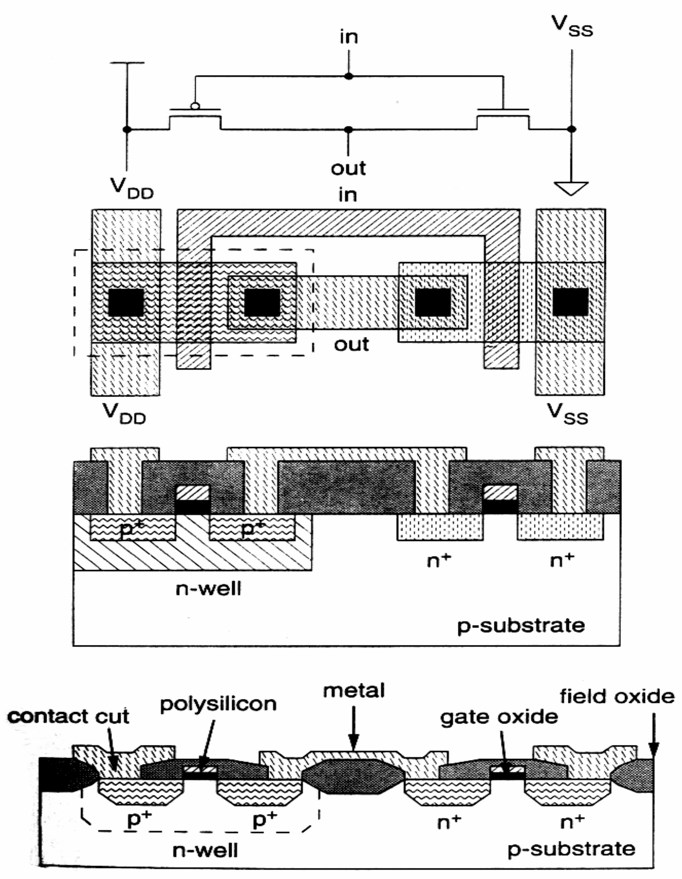
Packing, Testing

全客戶 vs. 半客戶式設計

↑ 半客戶設計

↓ 全客戶設計

CMOS Inverter 

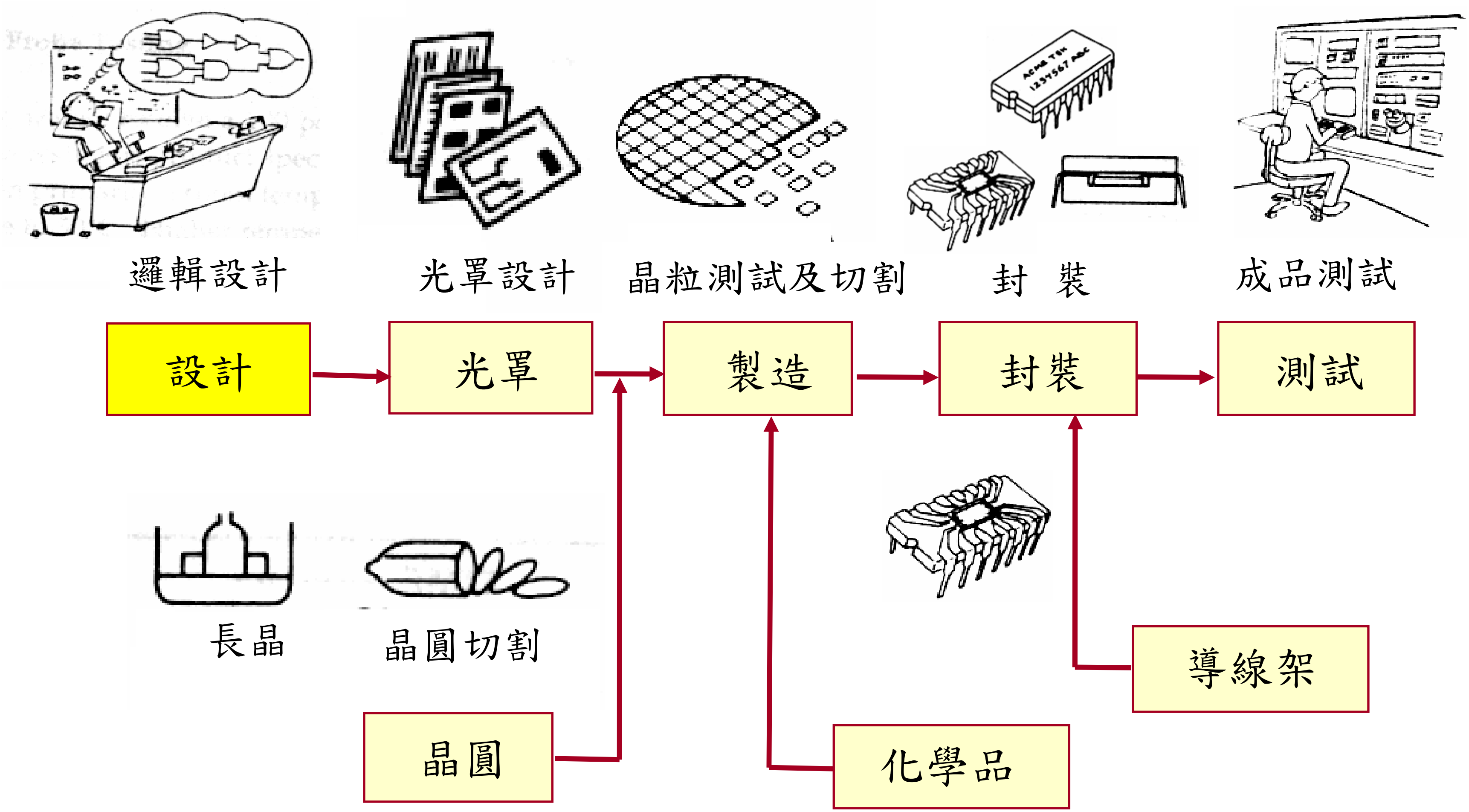


done by chip designer (full-custom) or tools (cell-based)

masking

done by TSMC, UMC

IC Industry in Taiwan



積體電路 (IC) 分類

- SSI (Small-Scaled Integrated Circuits)
  - 小型積體電路→含數十個元件 (1970s)
- MSI (Medium-Scaled IC)
  - 中型積體電路→含數百個元件
- LSI (Large-Scaled IC)
  - 大型積體電路→含數千個元件 (1980s)
- VLSI (Very Large Scaled IC)
  - 超大型積體電路→含數萬個元件 (1990s)
- SoC (System on a Chip)
  - 單晶片系統→含數百/千萬個元件 (2000s)

Semi Custom Design (半客戶式設計)

Semi Custom Design (Cell-Based)

a. Product specification

b. Modeling with HDL

c. Synthesis (by using suitable standard cell)

d. Simulation and verification

e. Physical placement and layout

f. Tape-out (real chip) -- implemented by suitable Fab companies

g. Testing -- implemented by suitable tools and mechanisms

(b–e) -- implemented with suitable tools

Two different solutions :

- **PLD** → FPGA or CPLD — Xilinx, Altera
- **Standard cell** → Real ASIC chip — Fab (TSMC, UMC, ..)

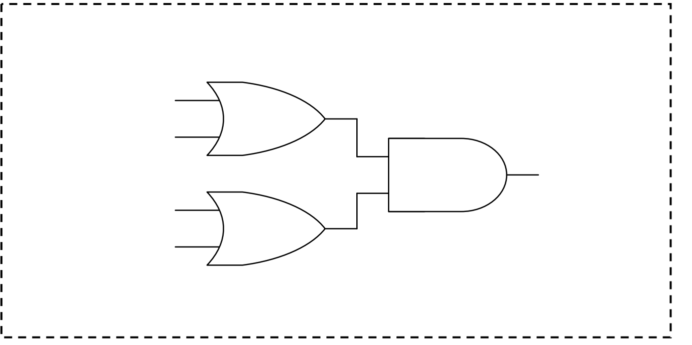
Design Entry for Cell-Based Flow

Choose the design entry method:

Schematic

Gate level design

Intuitive & easy to debug



HDL (Hardware Description Language)

Descriptive & portable

Easy to modify

```
always @(IN)
begin
  OUT = (IN[0] | IN[1]) &
     (IN[2] | IN[3]);
end
```

Mixed HDL & Schematic

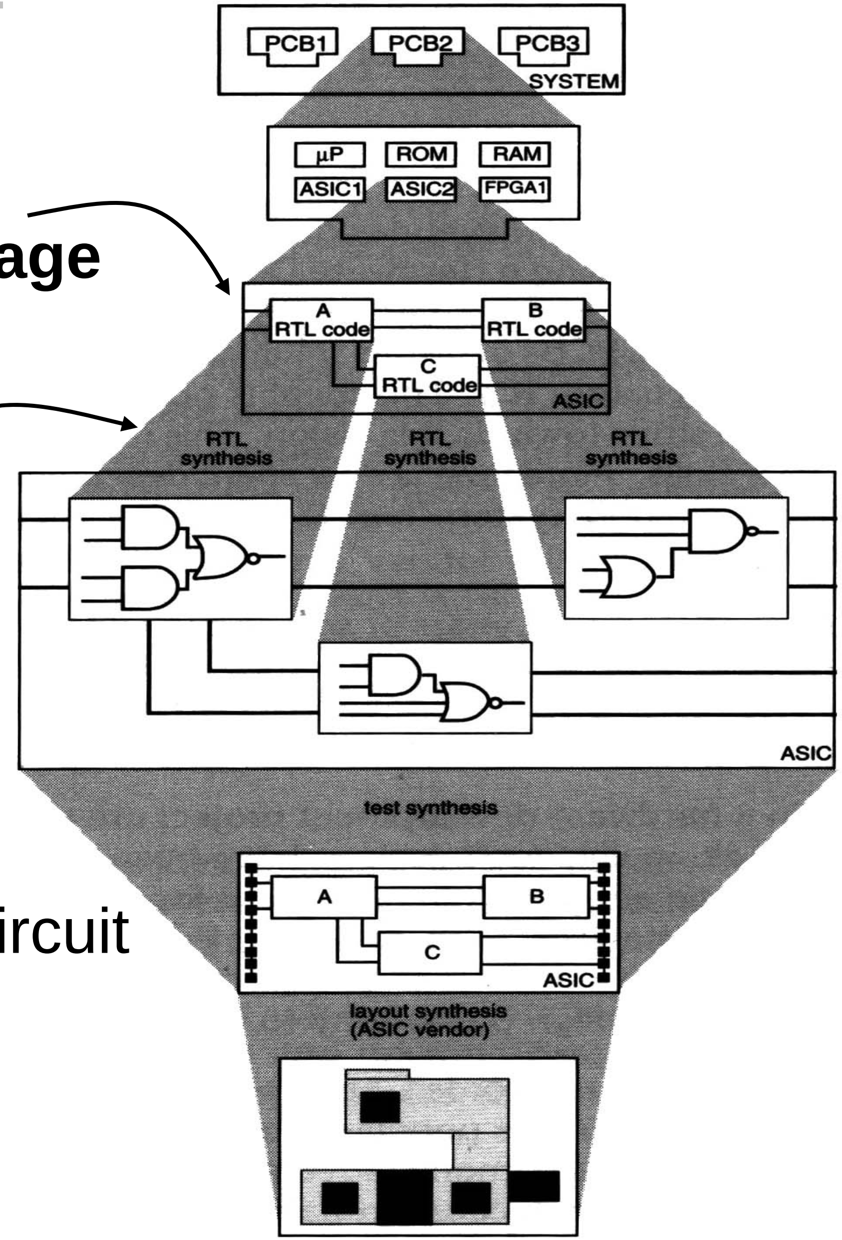
…

Hierarchical Components in PCB

1. **Describe the circuits with Hardware Description Language (HDL硬體描述語言)**
   Verilog or VHDL

2. **Synthesis (合成) the circuits …. by tools**

application specific integrated circuit (ASIC晶片)
IC or chip



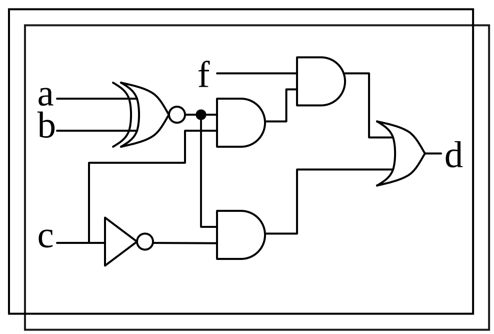
Synthesis (1/3)

- *Synthesis = Translation+Optimization+Mapping*

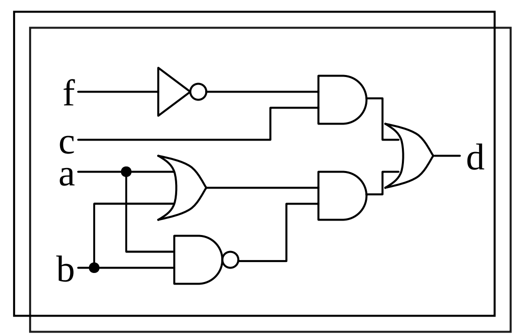
```
always @(…)
  if (a==b)
    if (c==1)
      d=f;
    else
      d=1;
  else
    d=0;
```

HDL Source

Translate into Boolean Representation



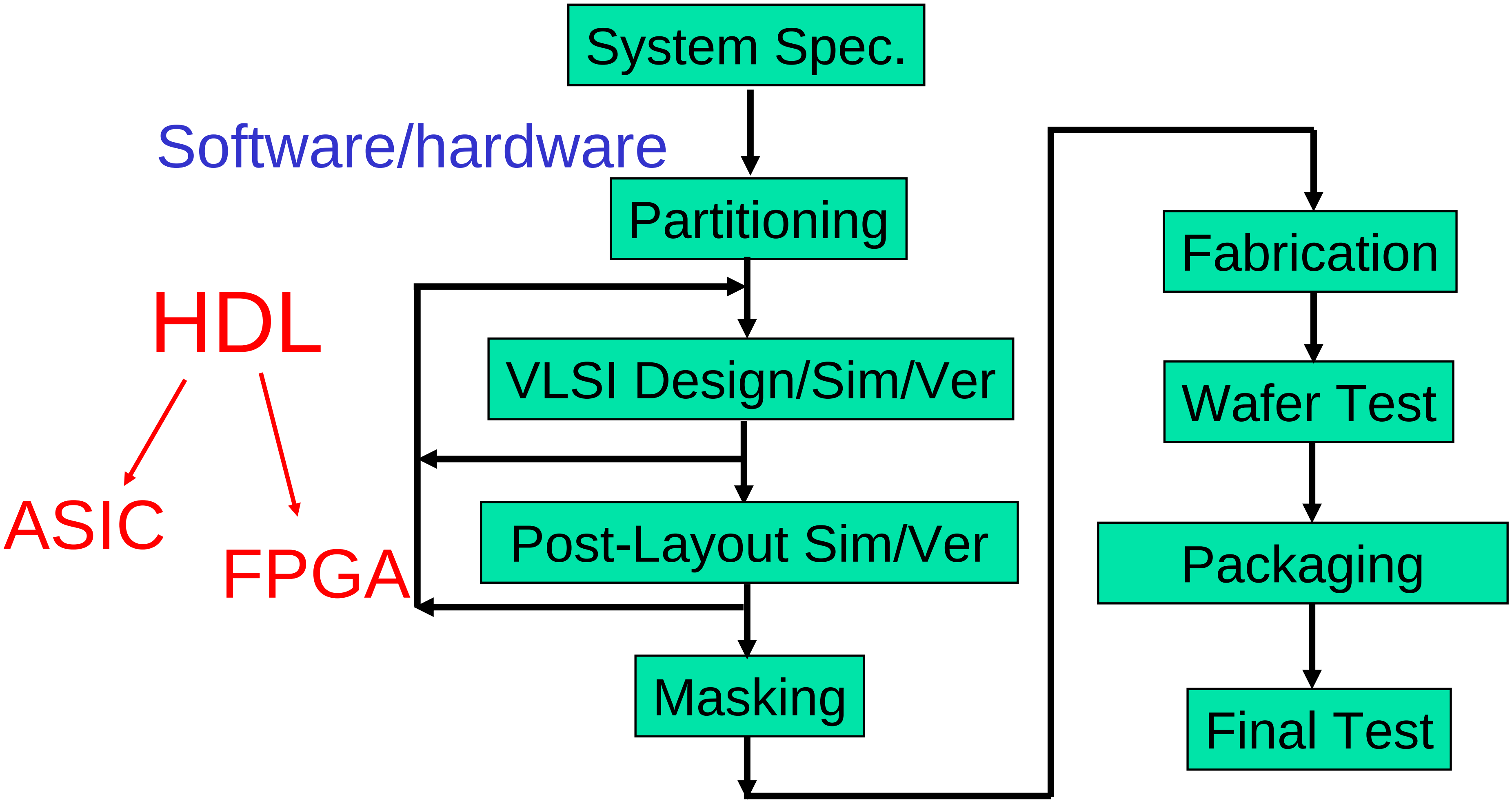
Optimize + Map



Target Technology

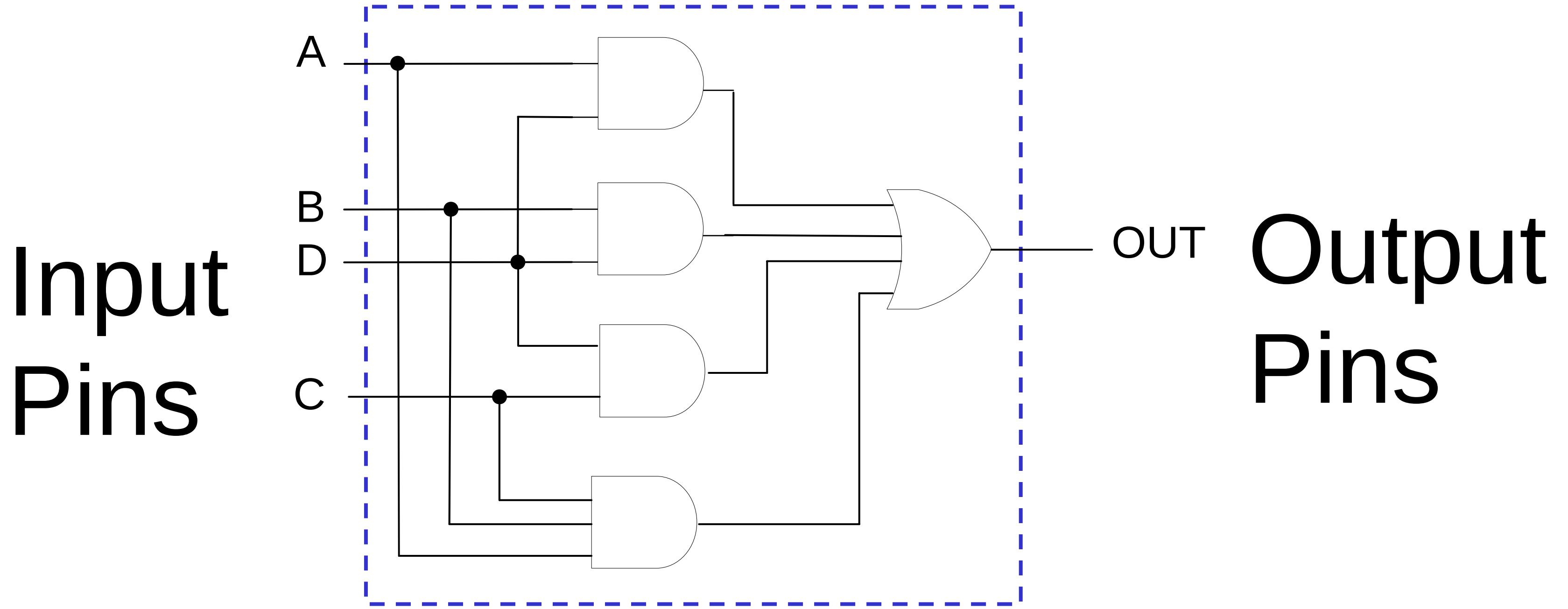
Process of logic synthesis

Circuit Design Process

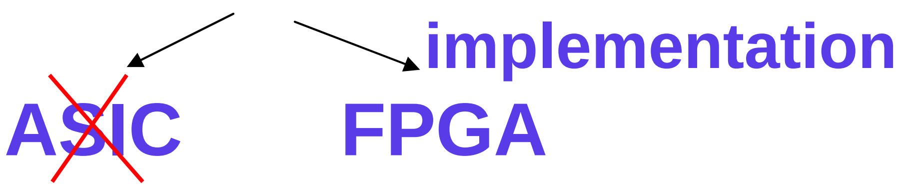


IC Design Flow? (1/3)

1. Design



Users design their circuits with HDL.



IC Design Flow? (2/3)

2. Synthesis

Users simulate (模擬) or synthesis (合成) his circuit with the related tools.

ASIC:

~~**Cadence Verilog-XL, Synopsys,….**~~

FPGA

- **Altera:**
  - **Quartus**
- **Xilinx:**
  - **ISE**

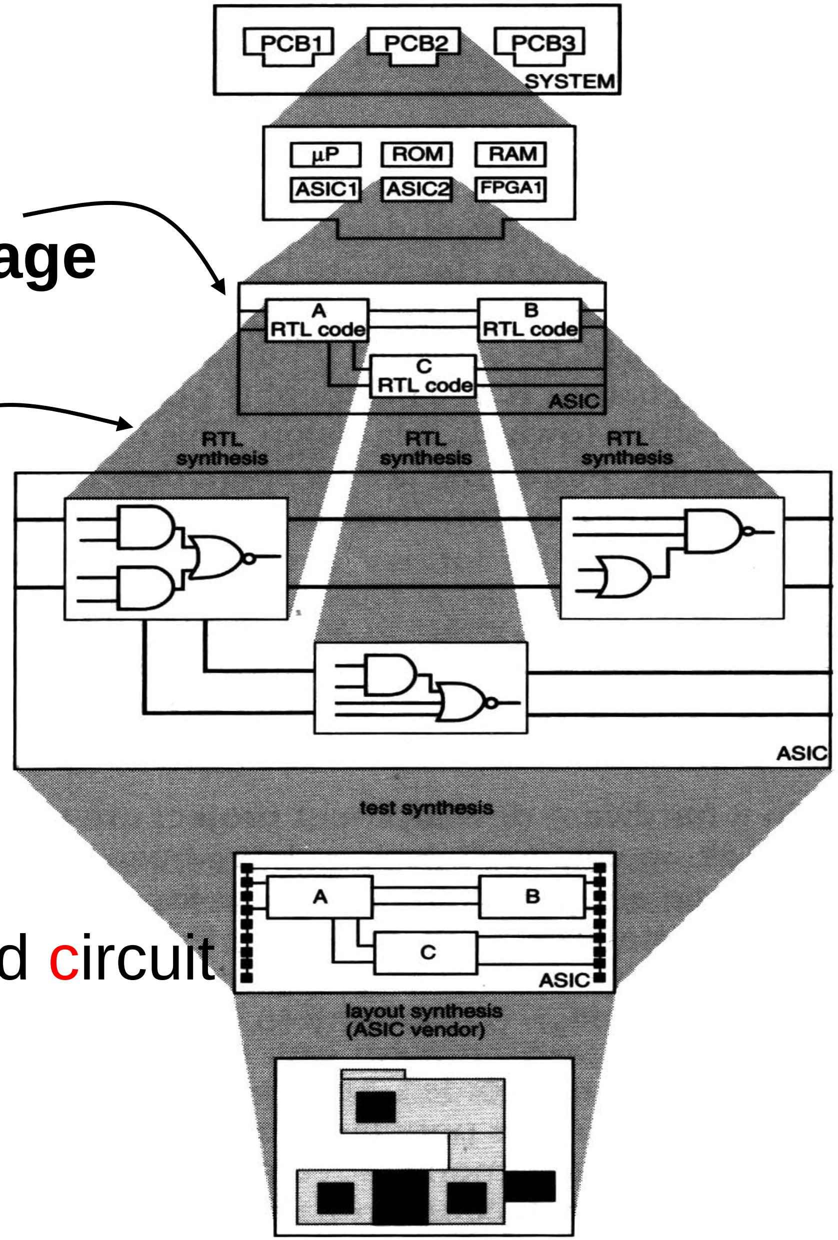
ModelSim, Synplicity

IC Design Flow? (3/3)

Describe the circuits with Hardware Description Language (HDL硬體描述語言)

Verilog or VHDL

Synthesis (合成) and P&R for the circuits…. by tools



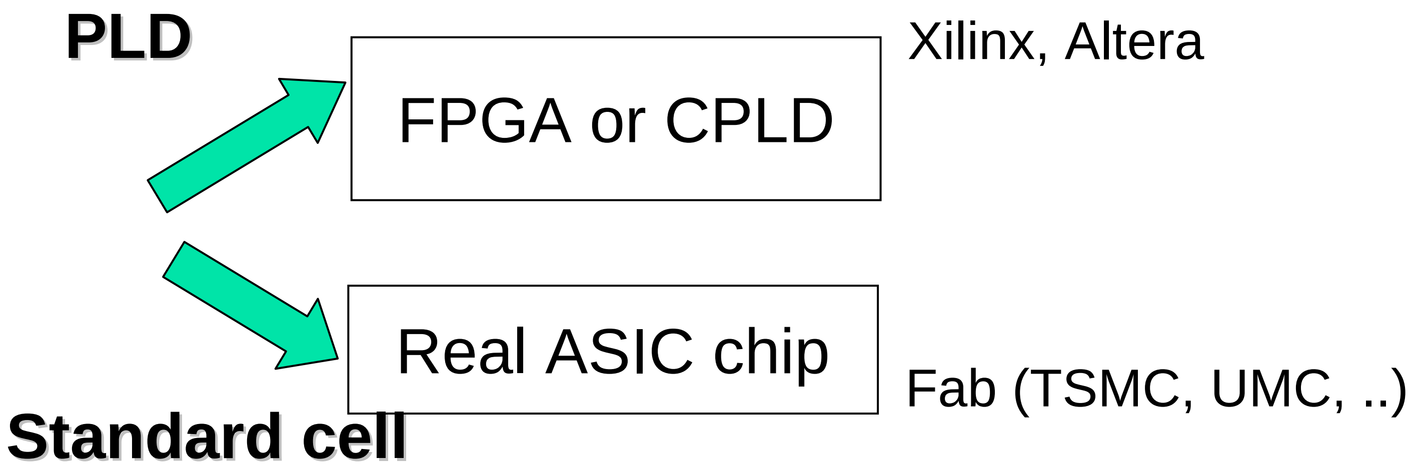
3. Implementation

a. application specific integrated circuit (ASIC晶片) IC or chip

b. FPGA (cheaper)

ASIC vs. FPGA

more flexible, shorter design cycle, suitable for smaller production



less flexible, long design cycle, larger-scale production to reduce price

We use SMIMS FPGA board in this class.