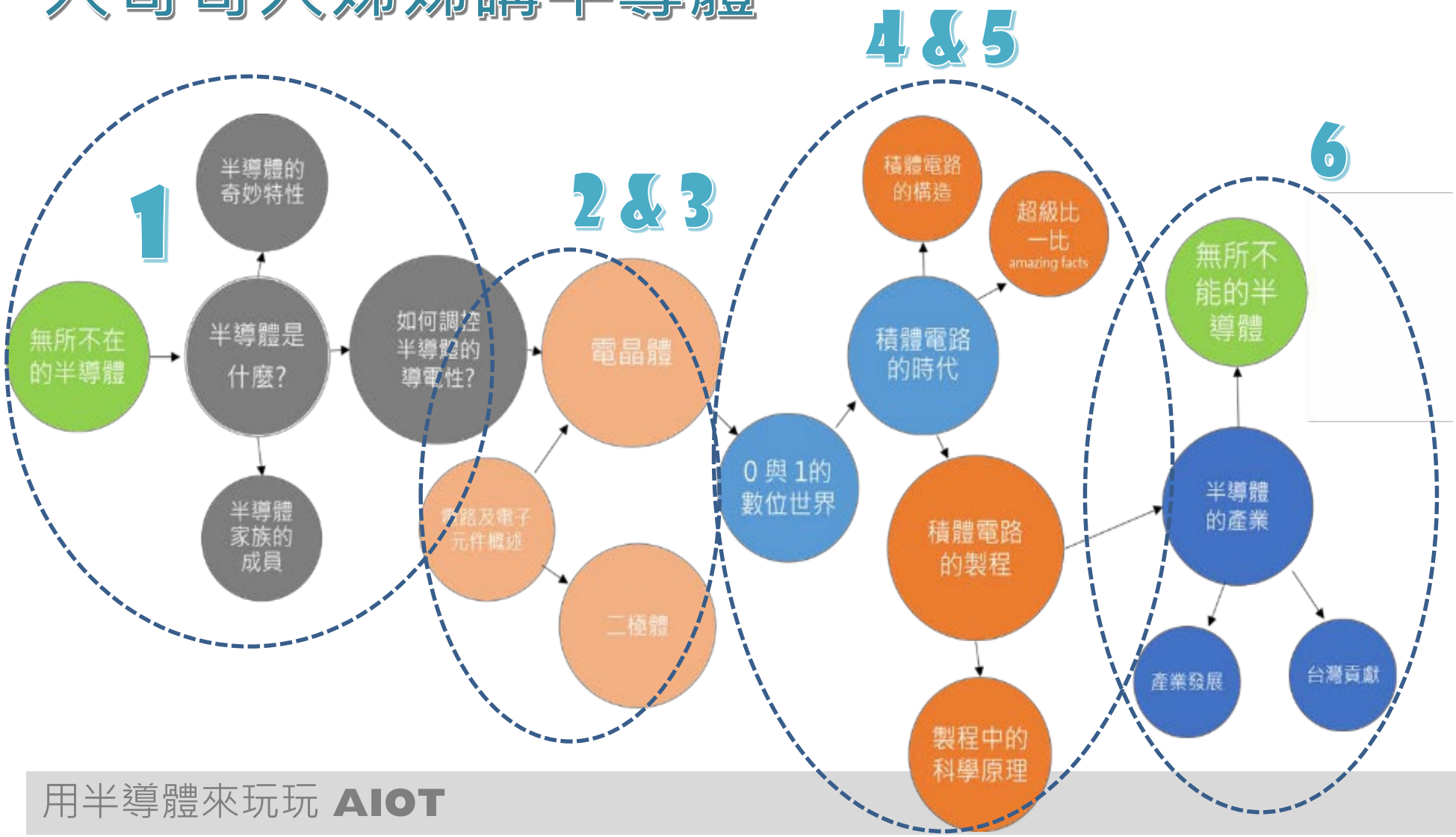
大哥哥大姊姊講半導體

4 認識積體電路-邏輯閘

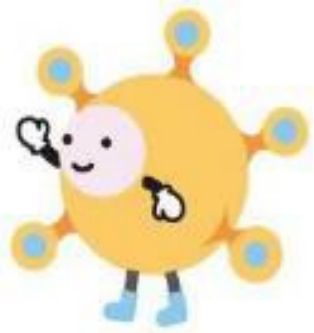
大哥哥大姊姊講半導體



用半導體來玩玩 **AIOT**

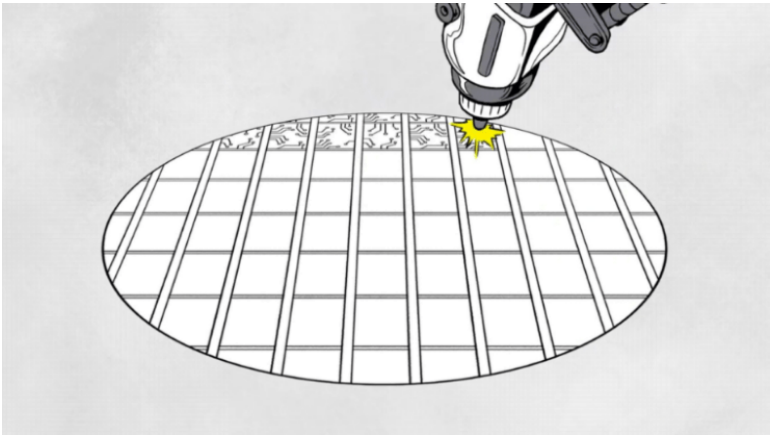
積體電路(Intergrated Circuit)

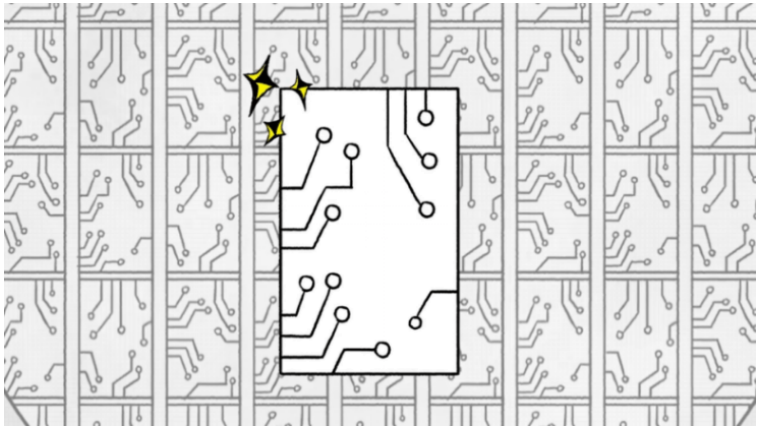
- 積體電路(IC)為利用大量電晶體(Transistor)組成特定功能的元件。
  - 跟指頭一樣小的晶片上就有上億個電晶體。
- 為什麼要積體，如同居住在城市，房子密集發展為更有效運用空間



晶圓(round wafer)

滿滿的電晶體!





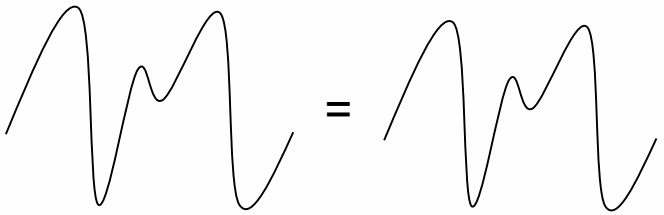
10奈米有多小?
頭髮直徑約50微米，細菌的大小約１微米，病毒的大小約１００奈米，而人類現在的技術能夠做出只有病毒１／１０大小的結構！
電晶體的大小越小，電子在其中流動的所需的距離就越小、功耗越低、速度也就越快！

當電晶體變得更小，我們在同一塊晶圓上能夠塞下的電晶體就會變得更多！

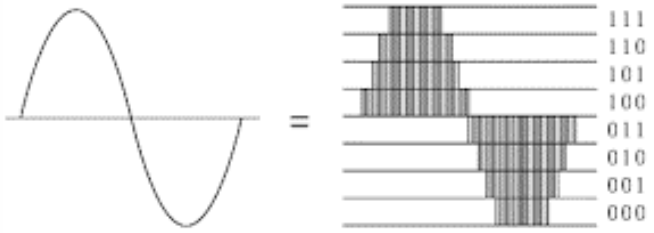
更多的電晶體，更低的生產成本，我們買電子設備時，就能享受更優惠的價格！

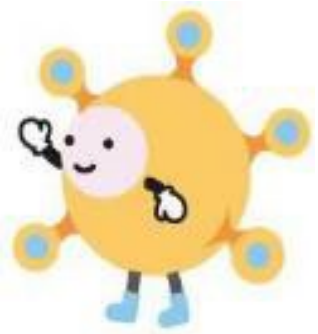
數位 vs. 類比 Digital vs. Analog

- 類比訊號:連續性型式進行傳遞，並為數位訊號最終呈現方式。



- 數位訊號:以0、1的非連續方式傳遞、運算及儲存。





數位訊號怎樣呈現

- 最基礎的數位訊號表達方式為0與1，就如同路上看到路燈不亮(0)與亮(1)
- 回家時怎麼跟家人明確說那些路燈不亮?

每個路燈有亮與不亮兩種狀態，依序表達
$(狀態)^2$ $(狀態)^1$ $(狀態)^0$ =>假設第二根沒亮
1 0 1 (二進制表達:101)
$1*(2)^2 + 0*(2)^1 + 1*(2)^0 = 4 + 0 + 1 = 5$ (十進制表達)

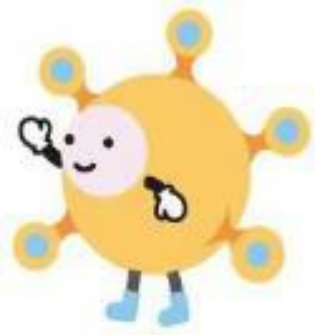
IC晶片種類

- 邏輯晶片

  - 執行OR、AND這類邏輯的晶片，讓輸入訊號給出相對應的輸出。

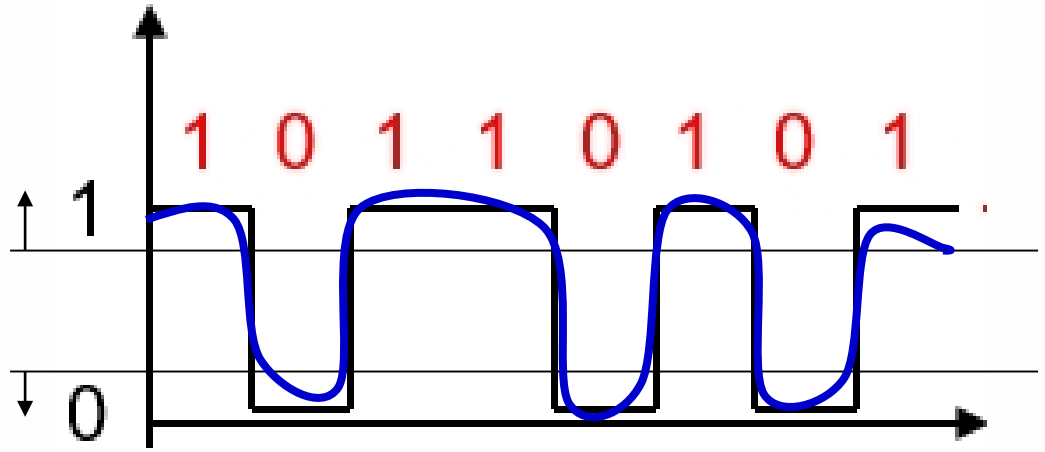
- 類比晶片
  - 用於處理連續性的訊號，擁有耐高壓、耐高電流的特性。

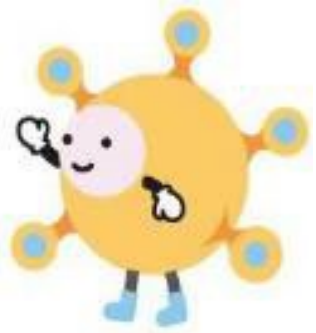
- 記憶體晶片
  - 儲存資料供日後需要資料時可以讀取。
- 微元件晶片
  - 統合數據並進行計算，如電腦的CPU。



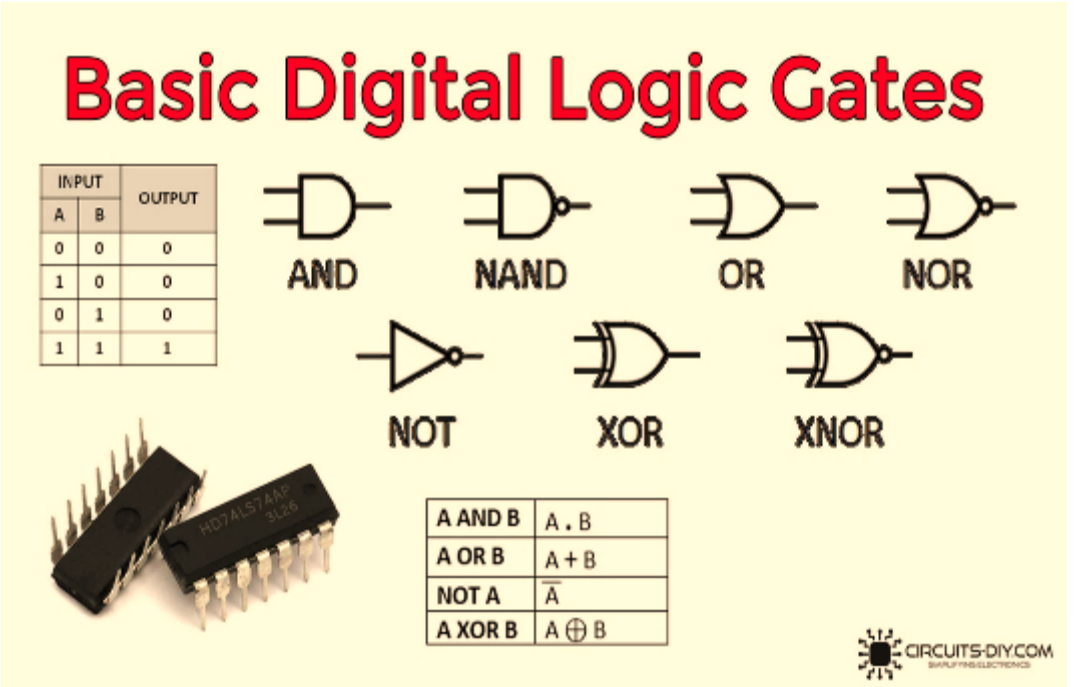
什麼是數位訊號的1與0

- 訊號的1與0是依定義的電壓水準做判斷。
  - 讀訊號時超過定義1的電壓即為1。
  - 讀訊號時低於定義0的電壓即為0。

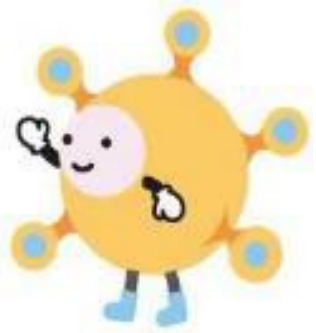




用電晶體，二極體，
電阻及電容完成
Logic gates

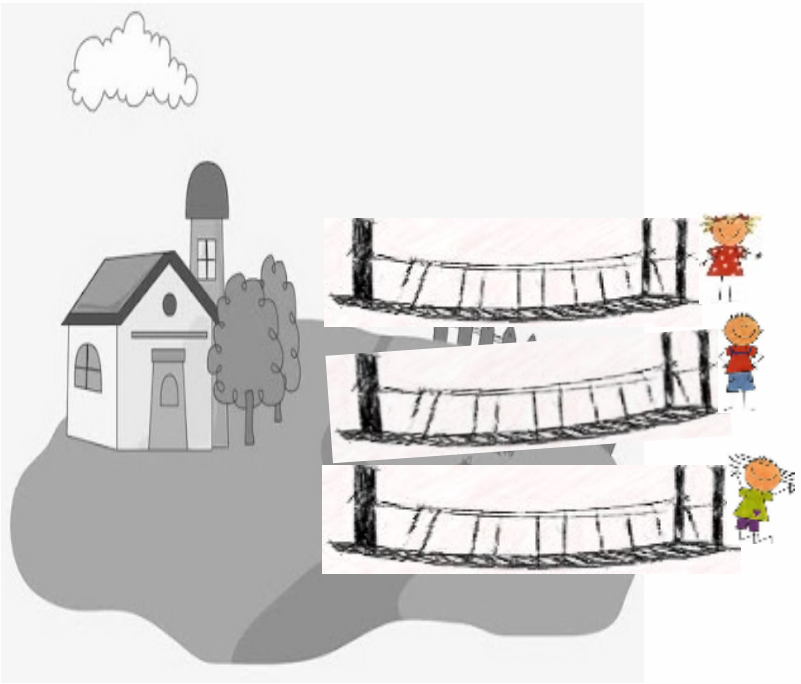


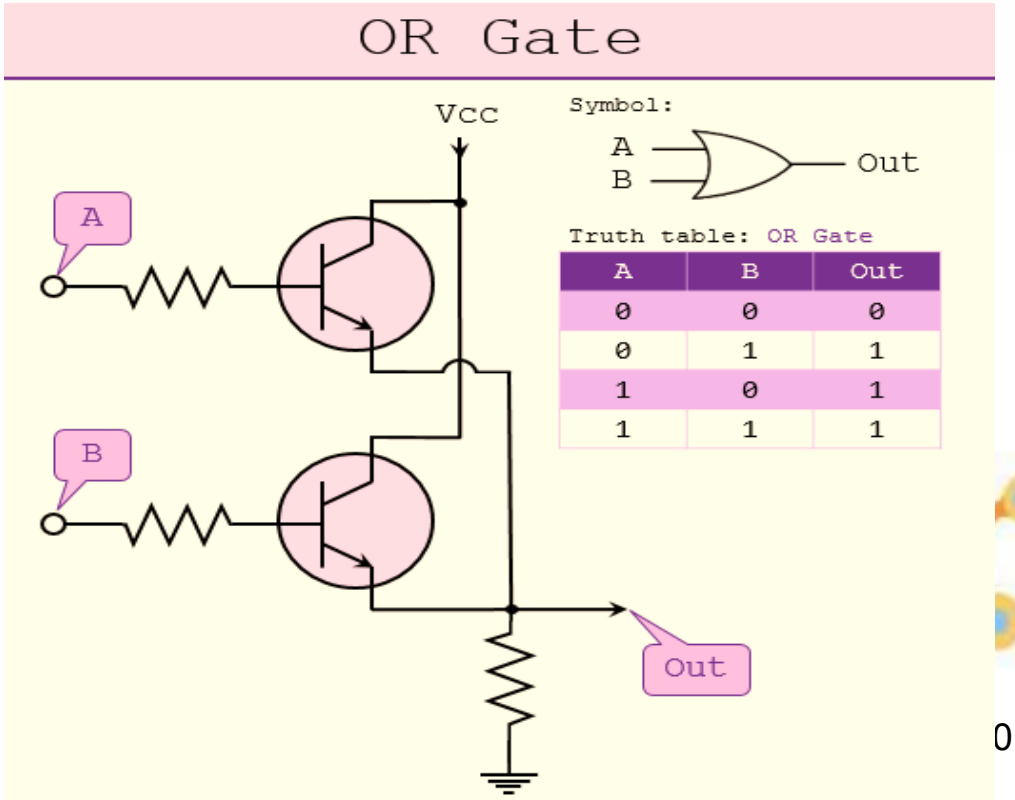
用NAND Gate 組合成加法器
用加法器組合加減乘除
用加減乘除完成積分微分矩陣運算……
用積分微分矩陣運算去完成 ……



或(or gate)

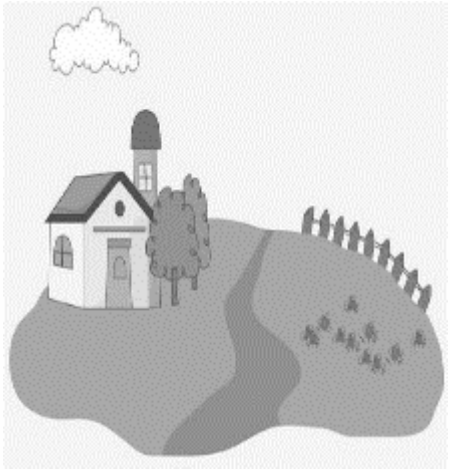
- OR gate輸出訊號，由多個輸入訊號組成，其中一個輸入訊號為1，輸出訊號就是1
- 如同要到校上課，每人走一條吊橋，只要其中有一條吊橋可通過(1)，就有人到校(1)。

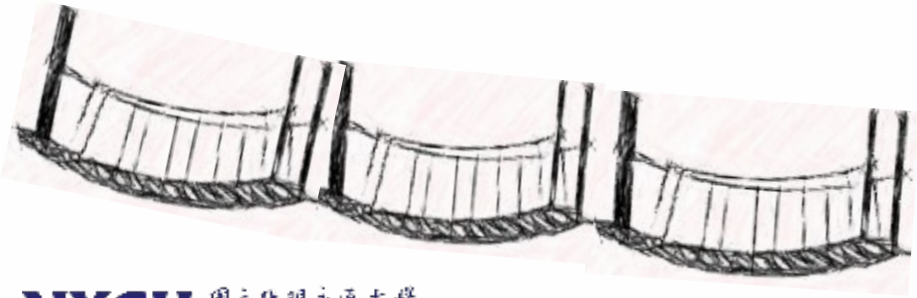


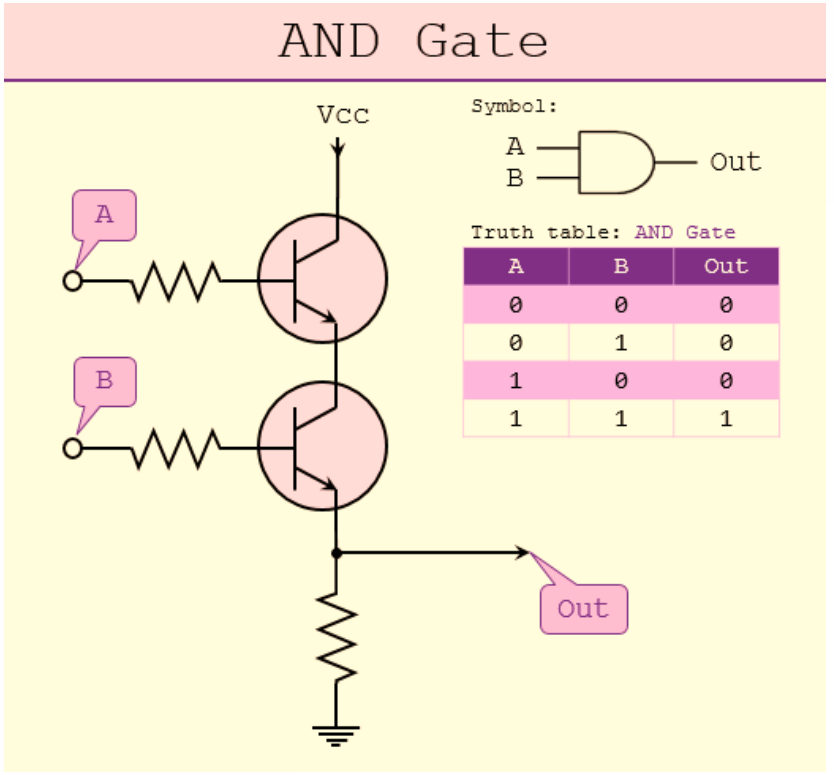


且(and)邏輯閘

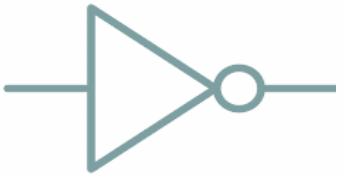
- AND gate輸出訊號由多個輸入訊號組成，只要其中一個輸入訊號為0，其輸出結果就是0。
- 如同有人到校須經過一條由多段吊橋所組成的吊橋，只要其中一條吊橋無法通過(0)，就無法到校(0)。



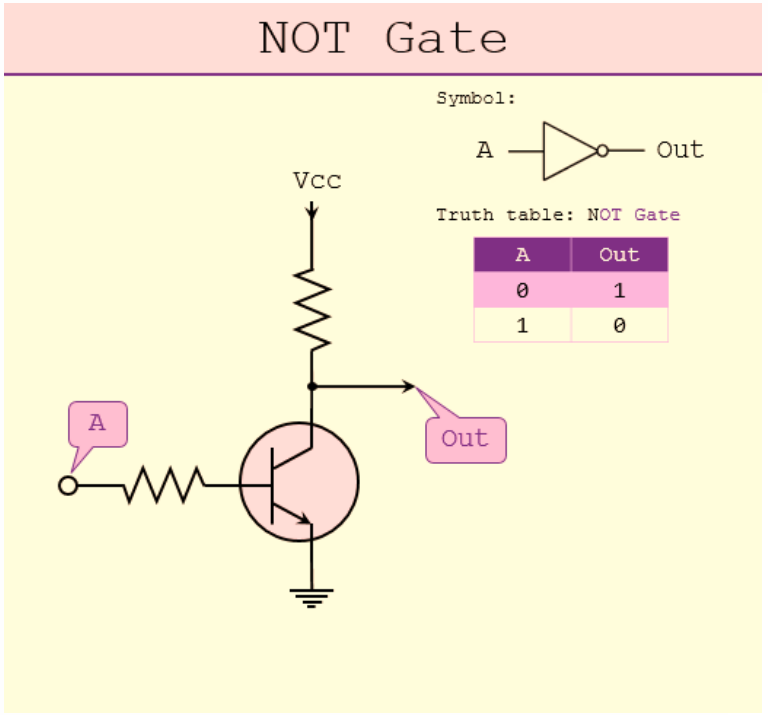




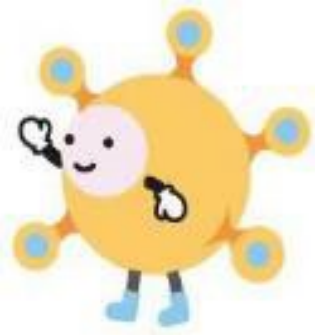
反相器(Not)



- 一個電路訊號，經反相器後輸出訊號與輸入訊號相反。
- 如同經過十字路口，當所在的馬路是綠燈時，則另一條路就會是紅燈，反之亦然。



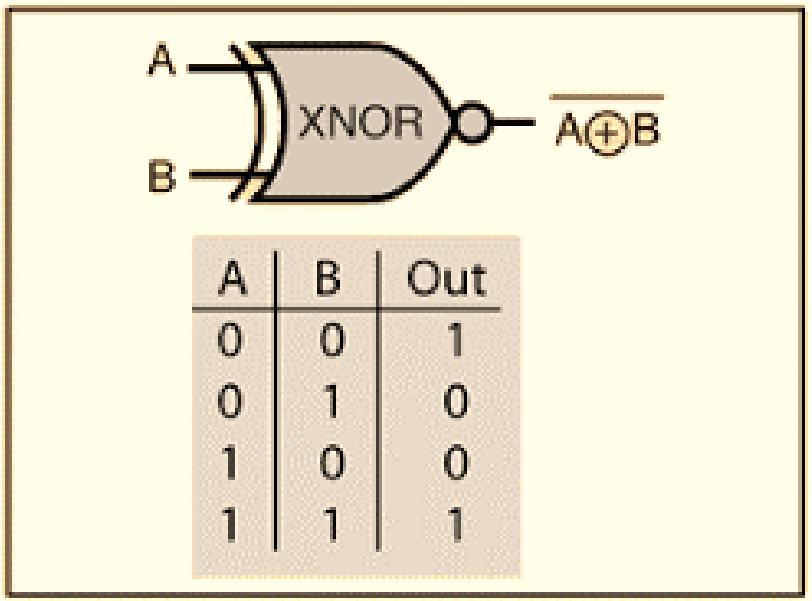
輸入	輸出
0	1
1	0



XOR

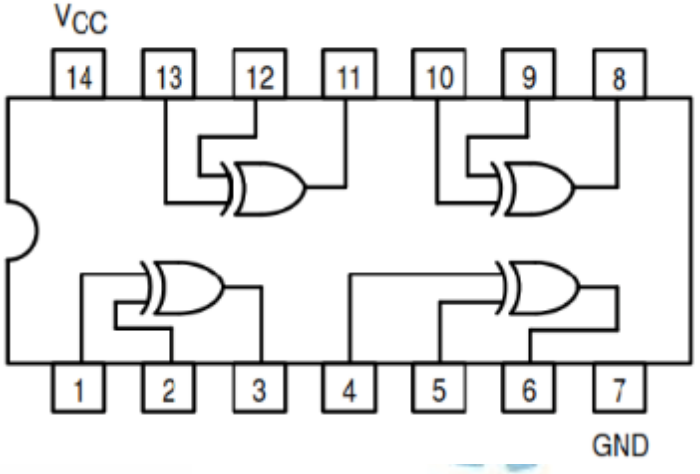
A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

XNOR

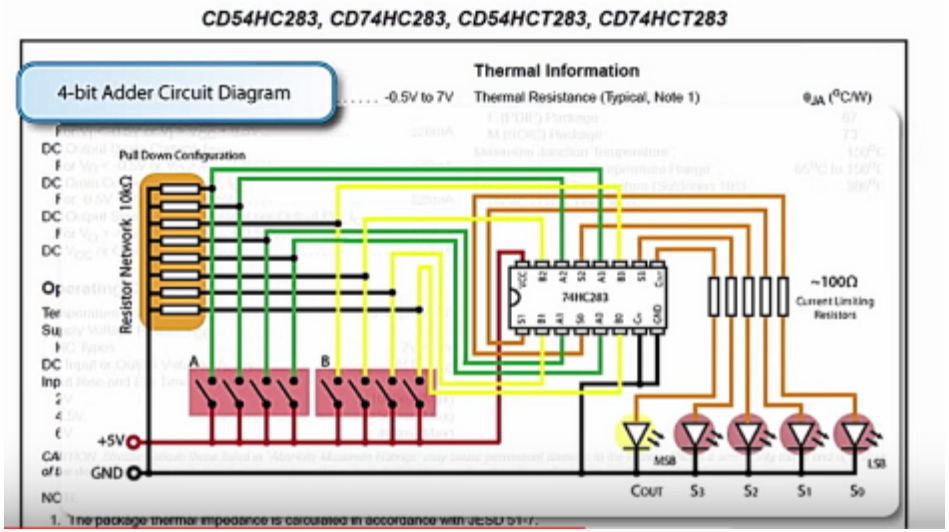


A	B	Out
0	0	1
0	1	0
1	0	0
1	1	1

7486



Data Sheet



[CD54HC283, CD74HC283, CD54HCT283, CD74HCT283 datasheet (Rev. D) (ti.com)]

TEXAS INSTRUMENTS

Data sheet acquired from Harris Semiconductor
SCHS176D

November 1997 - Revised October 2003

CD54HC283, CD74HC283, CD54HCT283, CD74HCT283

High-Speed CMOS Logic 4-Bit Binary Full Adder with Fast Carry

Features

- Adds Two Binary Numbers
- Full Internal Lookahead
- Fast Ripple Carry for Economical Expansion
- Operates with Both Positive and Negative Logic
- Fanout (Over Temperature Range)
  - Standard Outputs . . . . . 10 LSTTL Loads
  - Bus Driver Outputs . . . . . 15 LSTTL Loads
- Wide Operating Temperature Range . . . -55°C to 125°C
- Balanced Propagation Delay and Transition Times
- Significant Power Reduction Compared to LSTTL Logic ICs
- HC Types
  - 2V to 6V Operation
  - High Noise Immunity: N_{IL} = 30%, N_{IH} = 30% of V_{CC} at V_{CC} = 5V
- HCT Types
  - 4.5V to 5.5V Operation
  - Direct LSTTL Input Logic Compatibility, V_{IL} = 0.8V (Max), V_{IH} = 2V (Min)
  - CMOS Input Compatibility, $I_I \leq 1\mu A$ at V_{OL}, V_{OH}

Description

The 'HC283 and 'HCT283 binary full adders add two 4-bit binary numbers and generate a carry-out bit if the sum exceeds 15.

Because of the symmetry of the add function, this device can be used with either all active-high operands (positive logic) or with all active-low operands (negative logic). When using positive logic the carry-in input must be tied low if there is no carry-in.

Ordering Information

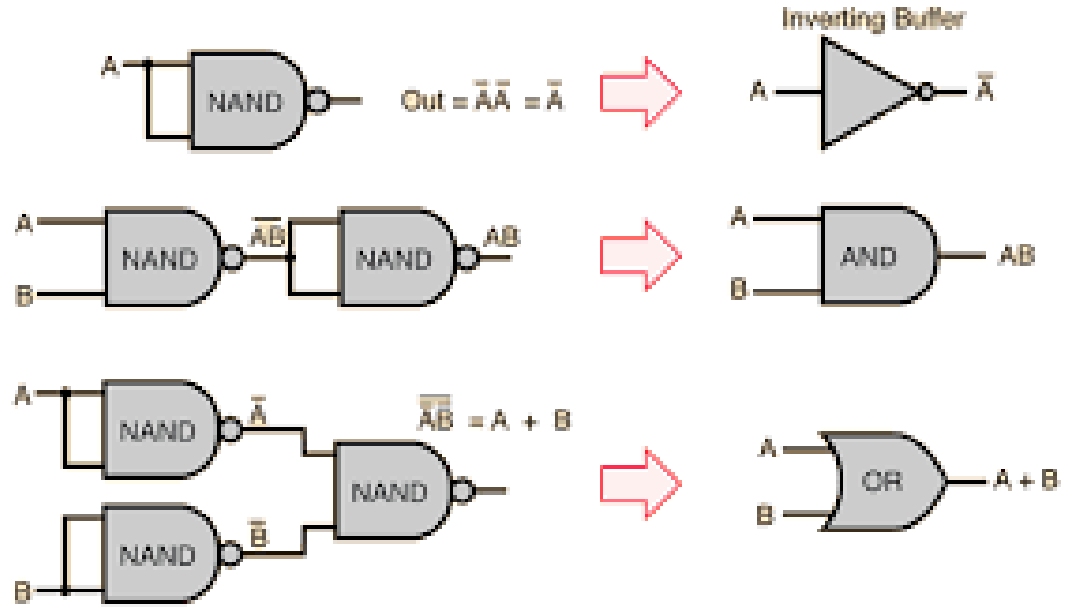
PART NUMBER	TEMP. RANGE (°C)	PACKAGE
CD54HC283F3A	-55 to 125	16 Ld CERDIP
CD54HCT283F3A	-55 to 125	16 Ld CERDIP
CD74HC283E	-55 to 125	16 Ld PDIP
CD74HC283M	-55 to 125	16 Ld SOIC
CD74HC283MT	-55 to 125	16 Ld SOIC
CD74HC283M96	-55 to 125	16 Ld SOIC
CD74HCT283E	-55 to 125	16 Ld PDIP
CD74HCT283M	-55 to 125	16 Ld SOIC
CD74HCT283MT	-55 to 125	16 Ld SOIC
CD74HCT283M96	-55 to 125	16 Ld SOIC

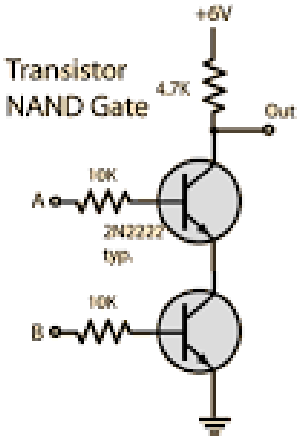
NOTE: When ordering, use the entire part number. The suffix 96 denotes tape and reel. The suffix T denotes a small-quantity reel of 250.

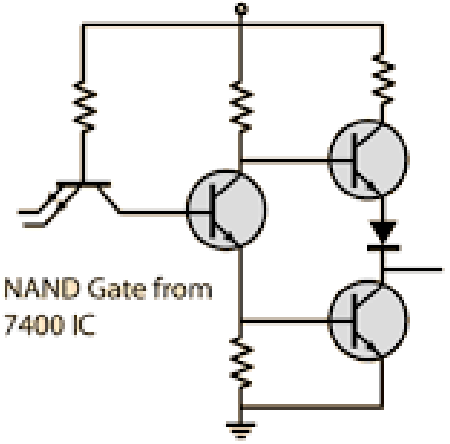
Q = A NAND B

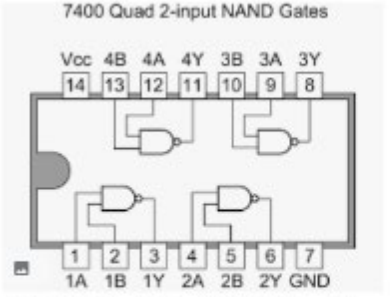
Truth Table

Input A	Input B		Output Q
0	0		1
0	1		1
1	0		1
1	1		0









File:7400 Quad 2-input NAND Gates.PN...
commons.wikimedia.org

JRe'S 7400 ic NAND gate (set of 5 pcs) : Amazon...
amazon.in

Binary

- Example: Base 10 number (Decimal)
- $(123.456)_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2} + 6 \times 10^{-3}$

The subscripts indicate “base-10 numeral system”

- Example: **Base 2 number (Binary)**
- $(101.101)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$
- Example: Base 8 number (Octal)
- $(101.101)_8 = 1 \times 8^2 + 0 \times 8^1 + 1 \times 8^0 + 1 \times 8^{-1} + 0 \times 8^{-2} + 1 \times 8^{-3}$
- Example: Base 16 number (Hexadecimal)
- $(101.101)_{16} = 1 \times 16^2 + 0 \times 16^1 + 1 \times 16^0 + 1 \times 16^{-1} + 0 \times 16^{-2} + 1 \times 16^{-3}$

Integer Number to Base-R Number Conversion

- 10進位整數轉2進位
  - 不斷地除以2，直到結果為0
  - 得到的餘數就是結果
- Example: 53_{10}->Binary number

Convert 53_{10} to binary.

2 | 53		
2 | 26	rem. = 1 = a_0	
2 | 13	rem. = 0 = a_1	
2 | 6	rem. = 1 = a_2	$53_{10} = 110101_2$
2 | 3	rem. = 0 = a_3	
2 | 1	rem. = 1 = a_4	
0	rem. = 1 = a_5	

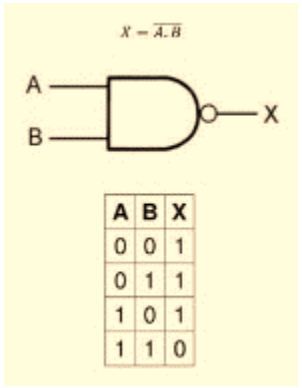
Arithmetic Operation: Addition

- 和小時候學的直式加法差不多
- Binary number addition
- Example: 00011010 + 00001100 = ?

Carry (11→ 1 / → 00 or 01 or 10 →0)

```
    1 1
  0 0 0 1 1 0 1 0
+ 0 0 0 0 1 1 0 0
-----------------
  0 0 1 0 0 1 1 0
```

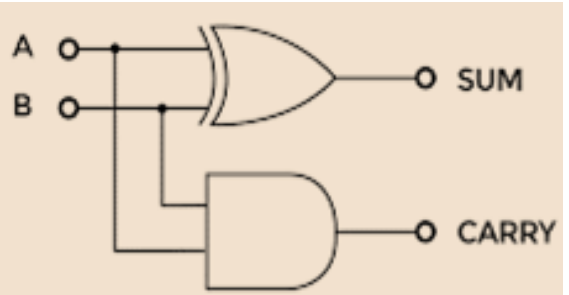
SUM (00 or 11 →0 / →01 or 10 →1)



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

A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

One bit Adder



A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Arithmetic Operation: Subtraction

- Binary number subtraction
- Example: 00100101-00010001 = ?

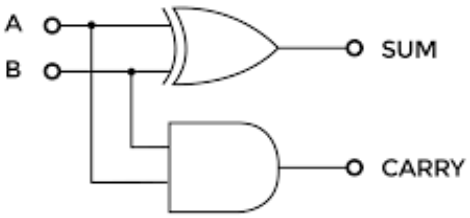
$$
\begin{array}{r}
\ 1 \\
\ 0\ 0\ 1\ {}^{1}0\ 0\ 1\ 0\ 1 \\
-\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1 \\
\hline
\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0
\end{array}
$$

37-17=20

How to simplify the calculation?

Source: https://commons.wikimedia.org/wiki/File:WhiteStork.svg

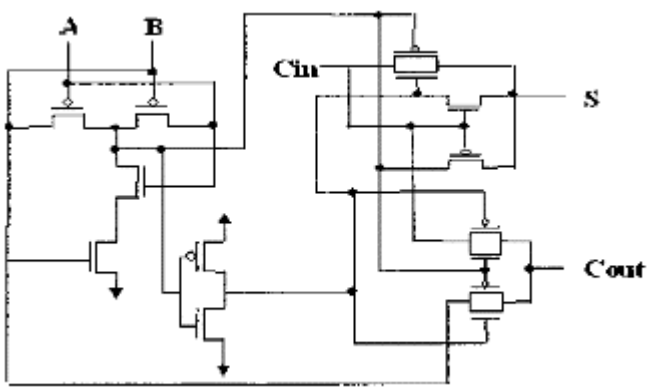
Gate level



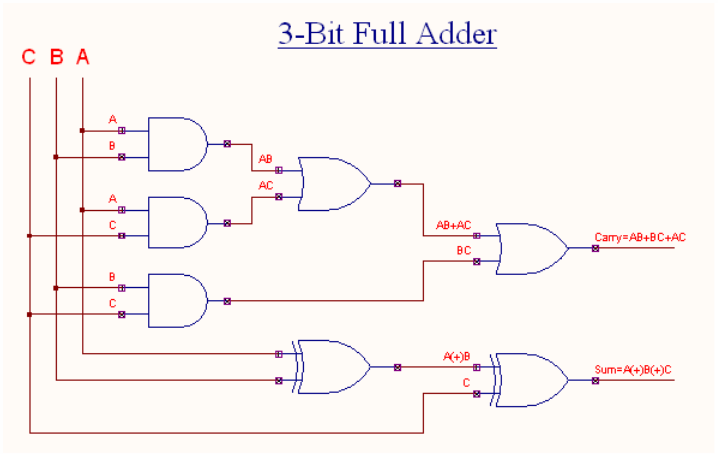
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

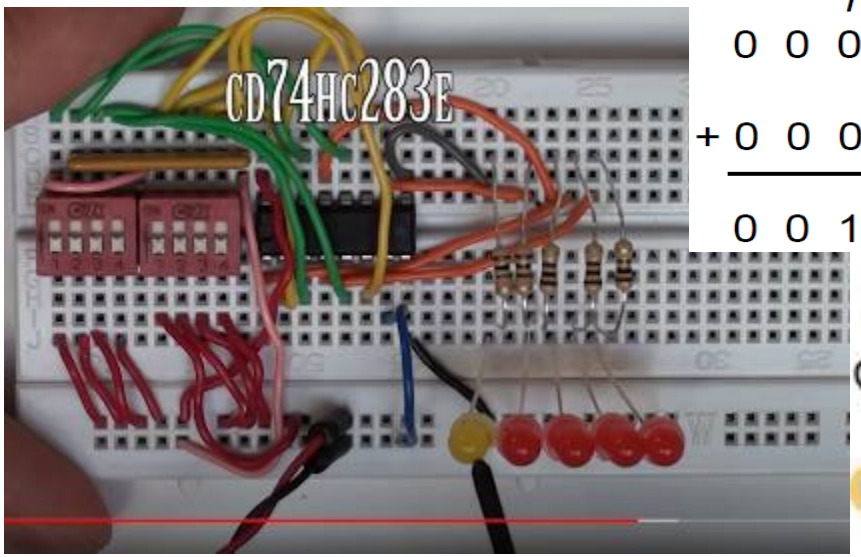
=

Transistor level

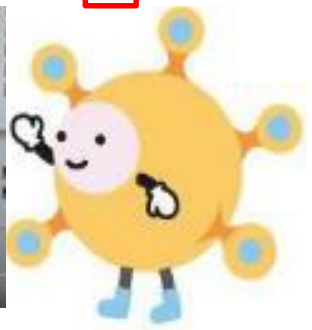


14 transistor 1-bit full adder





				1	1			
	0	0	0	1	1	0	1	0
+	0	0	0	0	1	1	0	0
	0	0	1	0	0	1	1	0



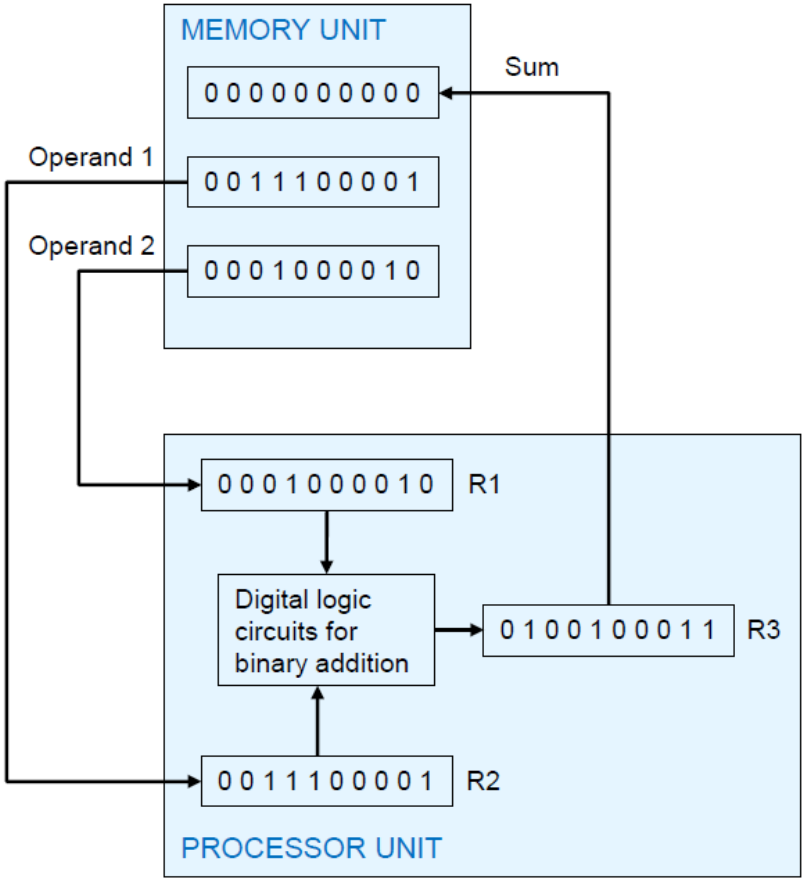
[(57) Digital Electronics: The 4-bit Adder (74HC283) - YouTube]

4-digit Signed Binary Numbers

Decimal	Signed-2's complement
+7	0111
+6	0110
+5	0101
+4	0100
+3	0011
+2	0010
+1	0001
0	0000

Decimal	Signed-2's complement
-8	1000
-7	1001
-6	1010
-5	1011
-4	1100
-3	1101
-2	1110
-1	1111

Binary Information Processing



Source: Yu-Te Liao (NYCUEE Professor) 2019 Lecture Notes for Logic Design

bit.ly/**3B2rbmv**