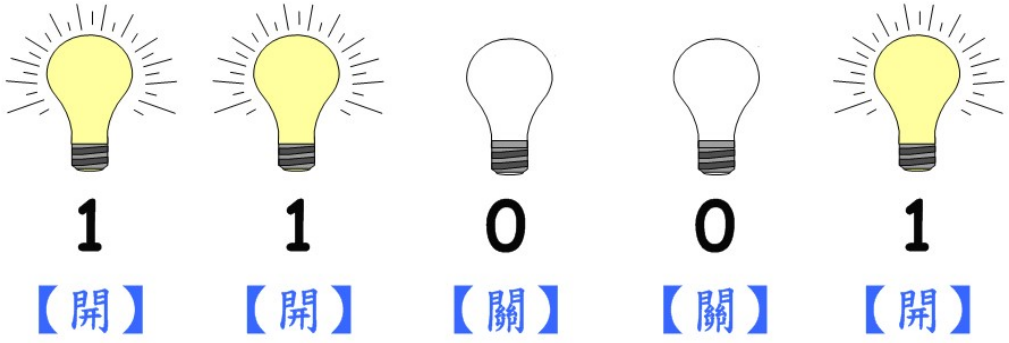
```
import math as m
x = m.sin(m.pi)
Print(x)
```

1.2246467991473532e-16 ← Why is it ***NOT 0*** ?!?!

電腦儲存數值的方法:二進制

二進制及電腦的儲存單位

- 電腦使用的資料表示法是二進位
  - 一般日常最常使用的是十進制、十二進制（例如一打）或六十進制（例如：時間）。十進位數（Decimal Digit）中每一個位數都有0~9等十種變化，每逢『十』就必須進位。
  - 但某些狀況，採用二分法更能夠簡化問題，如圖中，可以用11001來表示五顆燈泡的開關狀態。

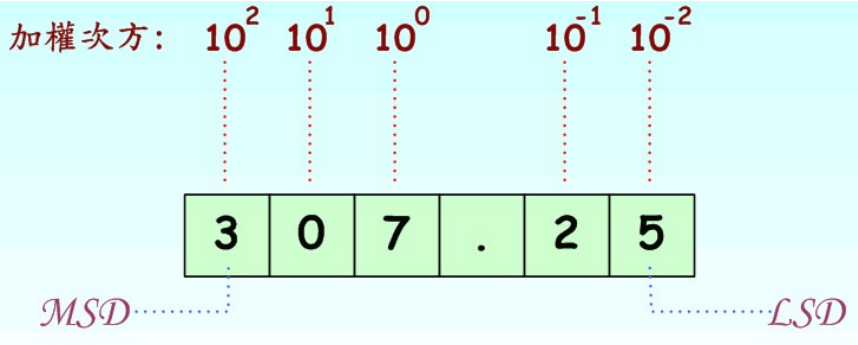


二進制及電腦的儲存單位

- 電腦使用電子元件來儲存及運算資料，而這些電子元件通常只能顯示兩種狀態：開（ON）或關（OFF），因此電腦使用二進位數（**Binary Digit**）來表示資料。
  - 二進位的每一個位數稱之為位元（**Bit**）。可用來表示0(關)或1(開)的狀態
  - 位元（**Bit**）是記憶體的最小儲存單位，但只能夠產生2種變化（0與1）
  - 為了表達更多狀態的變化，必須組合多個位元。通常會將**8個位元（Bits）組合成1個位元組（Byte）**，也就是1Byte=8 Bits，如此一來就可以產生2^8 **=256**種變化。

數字系統：十進制

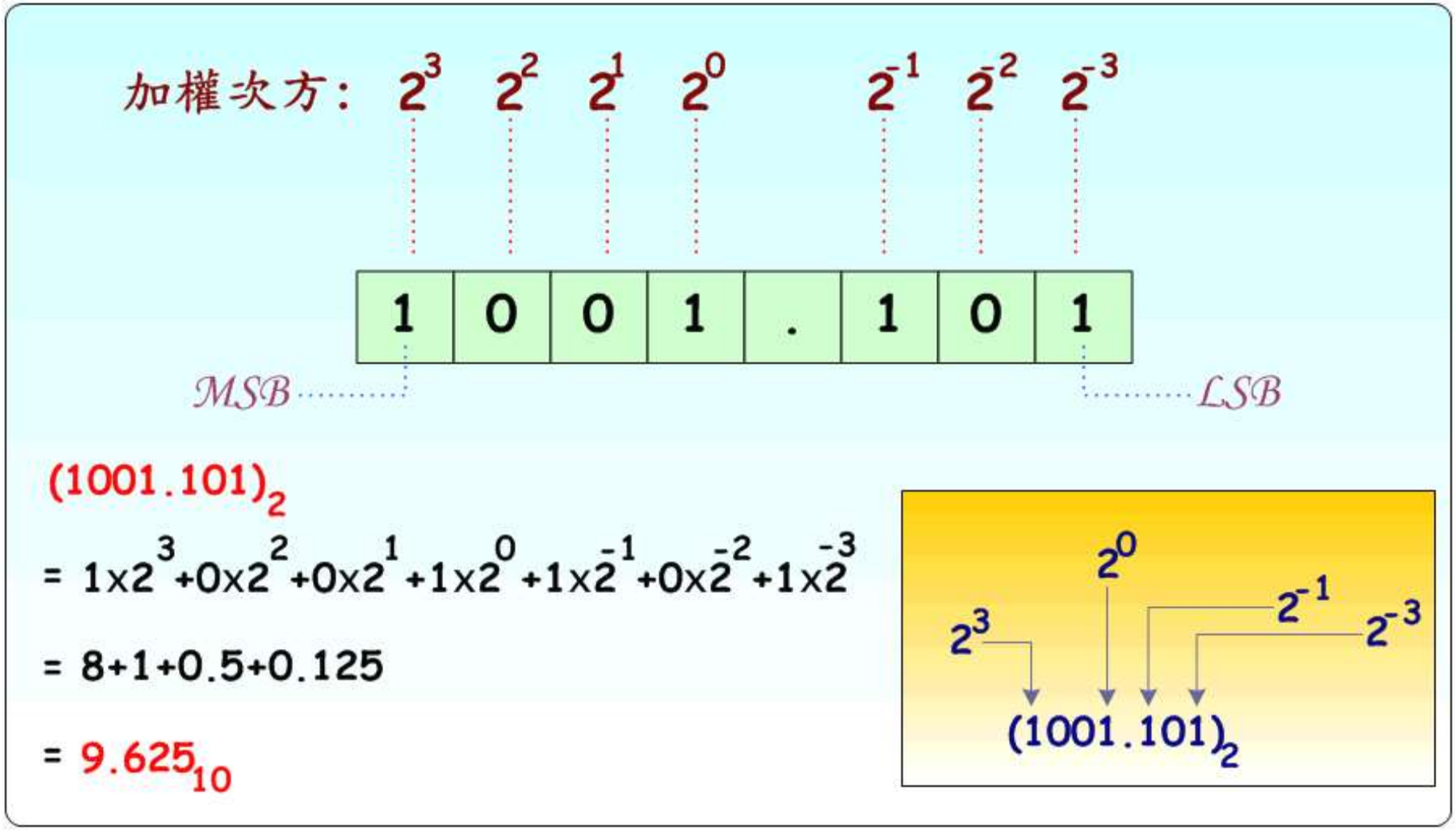
- 在十進位系統中，可以將一個數值分解如下：
  - $307.25 = 3*10^2 + 0*10^1 + 7*10^0 + 2*10^{-1} + 5*10^{-2}$



  - 每一位數能接受的數字符號為：0，1，2，3，4，5，6，7，8，9。
  - 每一位數所代表的量，根據其位置而有不同的指數加權（底數為10）。
  - 整數部份由小數點的左邊以10的正冪次方向左逐一遞增（次方由0開始）
  - 小數部份由小數點的右邊以10的負冪次方向右逐一遞減（次方由-1開始）
  - 10進制在做運算時，每一位數逢10向左進位。

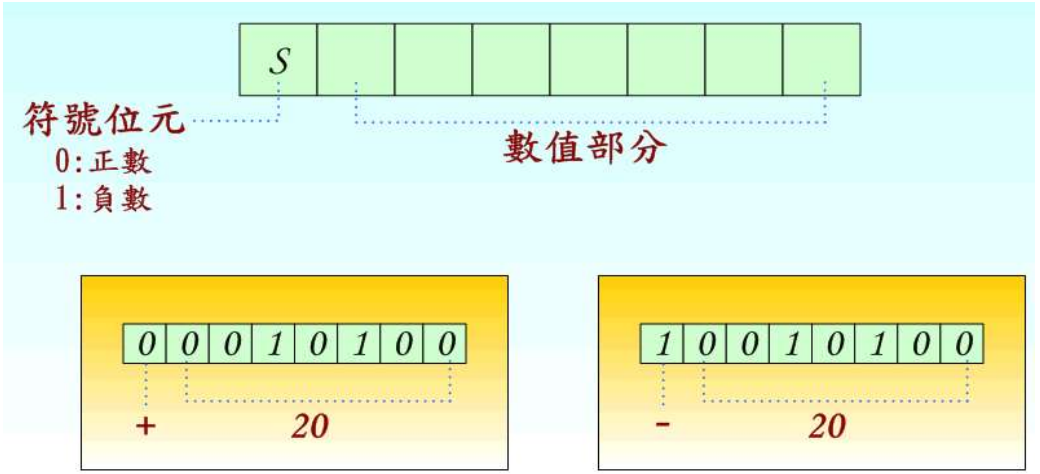
二進制數字系統

– 【範例】：1001.101_2之位數及加權計算如下：

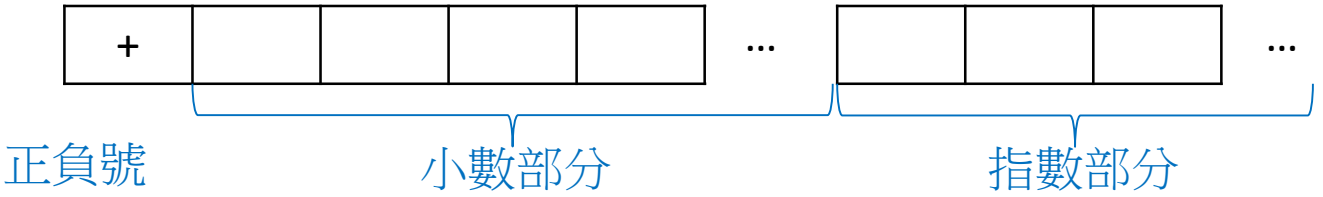


能夠儲存數值的位元數有限→能表達的數值範圍有限

- 用其中一個位元(通常在最左邊) 來表示該數值為正數還是負數
  - 使用n個位元時，數值的表達範圍只剩n-1個位元可以使用
  - 正整數的表達範圍是+0~+(2^{n-1}-1)
  - 負整數的表達範圍是-(2^{n-1}-1)~-0
  - 使用帶符號大小表示0的時候，+0與-0是不一樣的。
- 以8位元來表示正負整數：



實數的儲存方式 (32位元的機器為例)



- 小數部分占的位元數愈多，表達的有效數字愈多，精度愈高
- 指數部分占的位元數愈多，則能表示的數值範圍愈大。

- 單精度 (32位元儲存一個實數) float
  - 有效數字6~7，範圍3.4E-38～3.4E+38

- 倍精度 (64位元儲存一個實數) double
  - 有效數字15~16，範圍1.7E-308～1.7E+308

So…why math.sin(pi) is not zero!?

- π是無限小數 3. 14159265358979323846…
  電腦上只能用有限的位元來儲存:
  math.pi=3.141592653589793
  m.radians(180) =3.141592653589793

- 輸入sin函數的並不是完整的π值→結果很接近但不等於0
- m.sin(m.pi) = 1.2246467991473532e-16