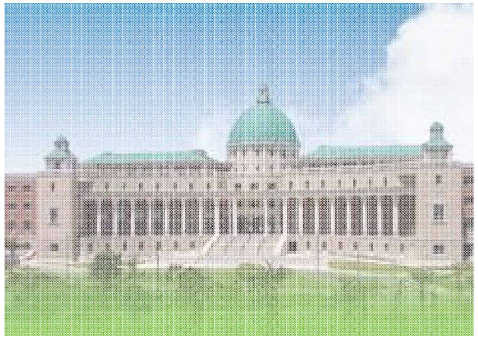
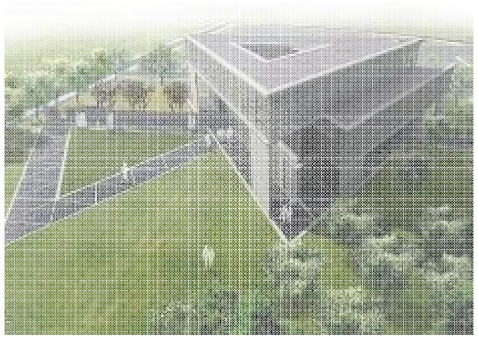



資訊與科技

數字系統與資料表示法

資料來源：新世代計算機概論(第五版)，學貫出版社

電腦的基本單位

- 位元 (bit)
- 二進位系統 (binary system)
- 位元組 (byte)
- 字元 (character)
- 字組 (word)

電腦的基本單位

單位	位元數目	簡寫
位元組 (byte)	8位元	B
字組 (word)	16位元	W
雙字組 (double word)	32位元	DW
四字組 (quad word)	64位元	QW

電腦的基本單位

單位	簡寫	準確值	近似值
千位元組 (kilobyte)	KB	2^{10}Bytes	10^{3}Bytes
百萬位元組 (megabyte)	MB	2^{20}Bytes	10^{6}Bytes
十億位元組 (gigabyte)	GB	2^{30}Bytes	10^{9}Bytes
兆位元組 (terabyte)	TB	2^{40}Bytes	10^{12}Bytes
千兆位元組 (petabyte)	PB	2^{50}Bytes	10^{15}Bytes
百京位元組 (exabyte)	EB	2^{60}Bytes	10^{18}Bytes

數字系統

- 任何一個屬於K進位系統的正數N都可表示成如下多項式：

$$N = d_{p-1}K^{p-1} + d_{p-2}K^{p-2} + \ldots + d_1K^1 + d_0K^0 + d_{-1}K^{-1} + d_{-2}K^{-2} + \ldots + d_{-q}K^{-q}$$

$$= \sum_{i=-q}^{p-1} d_i K^i \quad , \quad 0 \le d_i \le K-1 \quad , \quad -q \le i \le p-1$$

- N通常寫成$N_K = (d_{p-1}d_{p-2}\cdots d_1d_0.d_{-1}d_{-2}\cdots d_{-q})_K$

數字系統

- 12345.678_{10}是一個十進位數字，我們可以將它表示成如下多項式：

  $12345.678_{10} = 1 \times 10^{4} + 2 \times 10^{3} + 3 \times 10^{2} +$

  $4 \times 10^{1} + 5 \times 10^{0} + 6 \times 10^{-1} +$

  $7 \times 10^{-2} + 8 \times 10^{-3}$

- 1101010.11_{2}是一個二進位數字，我們可以將它表示成如下多項式：

  $1101010.11_{2} = 1 \times 2^{6} + 1 \times 2^{5} + 0 \times 2^{4} +$

  $1 \times 2^{3} + 0 \times 2^{2} + 1 \times 2^{1} +$

  $0 \times 2^{0} + 1 \times 2^{-1} + 1 \times 2^{-2}$

數字系統

- 1234.567_8是一個八進位數字，我們可以將它表示成如下多項式：

  $1234.567_8 = 1 \times 8^3 + 2 \times 8^2 + 3 \times 8^1 + 4 \times 8^0 +$

  $5 \times 8^{-1} + 6 \times 8^{-2} + 7 \times 8^{-3}$

- $56789A.BC_{16}$是一個十六進位數字，我們可將它表示成如下多項式：

  $56789A.BC_{16} = 5 \times 16^5 + 6 \times 16^4 + 7 \times 16^3 +$

  $8 \times 16^2 + 9 \times 16^1 + 10 \times 16^0 +$

  $11 \times 16^{-1} + 12 \times 16^{-2}$

二進位系統

- 二進位系統 (binary system) 是以0、1等兩個數字做為計數的基底。
- 為簡化，我們通常將二進位數字1000和十進位數字8寫成1000_2和8_{10} (或寫成$1000_2 = 8_{10}$)。

八進位系統

- 八進位系統 (octal system) 是以0、1、2～7等八個數字做為計數的基底。

十六進位系統

- 十六進位系統 (hexadecimal system) 是以0、1、2～9、A、B、C、D、E、F等十六個數字做為計數的基底。

十進位	二進位	八進位	十六進位	十進位	二進位	八進位	十六進位
0	0000	0	0	16	10000	20	10
1	0001	1	1	17	10001	21	11
2	0010	2	2	18	10010	22	12
3	0011	3	3	19	10011	23	13
4	0100	4	4	20	10100	24	14
5	0101	5	5	21	10101	25	15
6	0110	6	6	22	10110	26	16
7	0111	7	7	23	10111	27	17
8	1000	10	8	24	11000	30	18
9	1001	11	9	25	11001	31	19
10	1010	12	A	26	11010	32	1A
11	1011	13	B	27	11011	33	1B
12	1100	14	C	28	11100	34	1C
13	1101	15	D	29	11101	35	1D
14	1110	16	E	30	11110	36	1E
15	1111	17	F	31	11111	37	1F

表2.3二、八、十、十六進位對照表

將二、八、十六進位數字轉換成十進位數字

$$
\begin{aligned}
5621.78_{10} &= (5 \times 1000) + (6 \times 100) + (2 \times 10) + \\
&\quad (1 \times 1) + (7 \times 0.1) + (8 \times 0.01) \\
&= (5 \times 10^{3}) + (6 \times 10^{2}) + (2 \times 10^{1}) + \\
&\quad (1 \times 10^{0}) + (7 \times 10^{-1}) + (8 \times 10^{-2})
\end{aligned}
$$

將二、八、十六進位數字轉換成十進位數字

$$51763.2_8 = (5 \text{ x } 8^4) + (1 \text{ x } 8^3) + (7 \text{ x } 8^2) +$$
$$(6 \text{ x } 8^1) + (3 \text{ x } 8^0) + (2 \text{ x } 8^{-1})$$
$$= (5 \text{ x } 4096) + (1 \text{ x } 512) + (7 \text{ x } 64) +$$
$$(6 \text{ x } 8) + (3 \text{ x } 1) + (2 \text{ x } 0.125)$$
$$= 20480_{10} + 512_{10} + 448_{10} + 48_{10} +$$
$$3_{10} + 0.25_{10}$$
$$= 21491.25_{10}$$

將二、八、十六進位數字轉換成十進位數字

$$F2A9.C_{16} = (F \times 16^{3}) + (2 \times 16^{2}) + (A \times 16^{1}) + (9 \times 16^{0}) + (C \times 16^{-1})$$

$$= (15 \times 4096) + (2 \times 256) + (10 \times 16) + (9 \times 1) + (12 \times 0.0625)$$

$$= 61440_{10} + 512_{10} + 160_{10} + 9_{10} + 0.75_{10}$$

$$= 62121.75_{10}$$

將二、八、十六進位數字轉換成十進位數字

$$10110.0011_2 = (1 \text{ x } 2^4) + (0 \text{ x } 2^3) + (1 \text{ x } 2^2) +$$
$$(1 \text{ x } 2^1) + (0 \text{ x } 2^0) + (0 \text{ x } 2^{-1}) +$$
$$(0 \text{ x } 2^{-2}) + (1 + 2^{-3}) +(1 + 2^{-4})$$
$$= (1 \text{ x } 16) + (0 \text{ x } 8) + (1 \text{ x } 4) +$$
$$(1 \text{ x } 2) + (0 \text{ x } 1) + (0 \text{ x } 0.5) +$$
$$(0 \text{ x } 0.25) + (1 \text{ x } 0.125) +$$
$$(1\text{x } 0.0625)$$
$$= 16_{10} + 4_{10} + 2_{10} + 0.125_{10} + 0.0625_{10}$$
$$= 22.1875_{10}$$

將十進位數字轉換成二、八、十六進位數字

將十進位數字59.75_{10}轉換成二進位數字：

(1) $59.75_{10} = 59_{10} + 0.75_{10}$

(2) 找出整數部分的二進位表示法

2	59	1 (59除以2的餘數)
2	29	1 (29除以2的餘數)
2	14	0 (14除以2的餘數)
2	7	1 (7除以2的餘數)
2	3	1 (3除以2的餘數)
	1	1 (最大有效字元)

商數小於除數時停止，依反方向寫下餘數得到$59_{10} = 111011_2$

將十進位數字轉換成二、八、十六進位數字

（3）找出小數部分的二進位表示法

$$
\begin{array}{rl}
 & 0.75 \leftarrow \text{取得小數部分乘以2} \\
 & \times\ 2 \\
\hline
\text{小數點右邊第一位} \rightarrow & 1.50 \\
 & 0.50 \leftarrow \text{取得小數部分乘以2} \\
 & \times\ 2 \\
\hline
\text{小數點右邊第二位} \rightarrow & 1.00 \leftarrow \text{小數部分等於0時停止}
\end{array}
$$

依序寫下乘以2之積數的整數部分得到$0.75_{10} = 0.11_2$

（4）將整數部分及小數部分的二進位表示法合併得到$59.75_{10} = 111011.11_2$

將十進位數字轉換成二、八、十六進位數字

將十進位數字5176.3125_{10}轉換成八進位數字：

（1）$5176.3125_{10} = 5176_{10} + 0.3125_{10}$

（2）找出整數部分的八進位表示法

8	5176	0 (5176除以8的餘數)
8	647	7 (647除以8的餘數)
8	80	0 (80除以8的餘數)
8	10	2 (10除以8的餘數)
	1	1 (最大有效數字)

商數小於除數時停止，依反方向寫下餘數得到$5176_{10} = 12070_8$

將十進位數字轉換成二、八、十六進位數字

(3) 找出小數部分的八進位表示法

$$
\begin{array}{r}
0.3125 \\
\times \quad\quad 8 \\
\hline
2.5000 \\
0.5000 \\
\hline
\times \quad\quad 8 \\
4.0000
\end{array}
$$

0.3125 ← 取得小數部分乘以8

小數點右邊第一位 → 2.5000

0.5000 ← 取得小數部分乘以8

小數點右邊第二位 → 4.0000

依序寫下乘以8之積數的整數部分得到$0.3125_{10} = 0.24_8$

(4) 將整數部分及小數部分的八進位表示法合併，得到 $5176.3125_{10} = 12070.24_8$。

將十進位數字轉換成二、八、十六進位數字

將十進位數字4877.610轉換成十六進位數字 ：

（1）4877.610 = 487710 + 0.610

（2）找出整數部分的十六進位表示法

16	4877	13 (4877除以16的餘數)
16	304	0 (304除以16的餘數)
16	19	3 (19除以16的餘數)
	1	1 (最大有效數字)

商數小於除數時停止，依反方向寫下餘數得到

487710 = 130D8

將十進位數字轉換成二、八、十六進位數字

（3）找出小數部分的十六進位表示法

0.6　　取得小數部分乘以16

x 16

小數點右邊第一位 9.6

0.6　　出現循環時停止

(從小數點右邊第一位開始)

依序寫下乘以16之積數的整數部分，得到0.610 = 0.916

（4）將整數部分及小數部分的十六進位表示法合併，得到

4877.610 = 130D.916

將八或十六進位數字轉換成二進位數字

$$5\ 7\ 6\ 2.\ 1\ 3_8 = 101\ 111\ 110\ 010.\ 001\ 011_2$$

$$E\ 8\ C\ 4.\ B_{16} = 1110\ 1000\ 1100\ 0100.\ 1011_2$$

將二進位數字轉換成八或十六進位數字

$$011\ 010\ 111.101\ 100_2 = 3\ 2\ 7.\ 5\ 4_8$$

整數部分每三個數字一組，不足三個的
就在左邊補上0
小數部分每三個數字一組，不足三個的
就在右邊補上0

將二進位數字轉換成八或十六進位數字

$0010\ 1101\ 0111\ 1010.\ 1111\ 0010_2 = 2\ D\ 7\ A.\ F\ 2_{16}$

整數部分每四個數字一組，不足四個的就在左邊補上0

小數部分每四個數字一組，不足四個的就在右邊補上0

數值表示法

- 帶符號大小
- 1's補數
- 2's補數

帶符號大小

- 假設使用n位元來表示正負整數，那麼最左邊的位元(MSD) 是整數的正負符號，0表示正數，1表示負數，剩下的n - 1位元才是整數的數值大小，正整數的範圍為$0 \sim 2^{n-1}-1$，負整數的範圍為 $-(2^{n-1}-1) \sim 0$。

1's補數

- 假設使用n位元來表示正負整數，最左邊的位元 (MSD) 是整數正負符號，0表示正數，1表示負數，剩下n - 1位元才是整數數值大小，正整數範圍為$0 \sim 2^{n-1}-1$，負整數範圍為 $-(2^{n-1}-1) \sim 0$。

2’s補數

- 假設使用n位元來表示正負整數，那麼最左邊的位元 (MSD) 是整數的正負符號，0表示正數，1表示負數，剩下的n - 1位元才是整數的數值大小，正整數範圍為$0 \sim 2^{n-1}-1$，負整數範圍為$-2^{n-1} \sim 0$。

數值表示法

十進位	帶符號大小	1's補數	2's補數	十進位	帶符號大小	1's補數	2's補數
+8	無	無	無	-8	無	無	1000
+7	0111	0111	0111	-7	1111	1000	1001
+6	0110	0110	0110	-6	1110	1001	1010
+5	0101	0101	0101	-5	1101	1010	1011
+4	0100	0100	0100	-4	1100	1011	1100
+3	0011	0011	0011	-3	1011	1100	1101
+2	0010	0010	0010	-2	1010	1101	1110
+1	0001	0001	0001	-1	1001	1110	1111
+0	0000	0000	0000	-0	1000	1111	0000

表2.4不同數值表示法對照表

補數的推廣

- $(K - 1)$'s補數：對於N_K的每位數字均以 $(K - 1)$ 減去該數字，便能求出N_K的 $(K - 1)$ 's補數為 $((K - 1 - d_{p-1})(K - 1 - d_{p-2})\cdots(K - 1 - d_1)(K - 1 - d_0).(K -1 - d_{-1})\cdots(K - 1 - d_{-q}))_K$；或者，您也可以套用公式 $(K_p -K_{-q}) - N_K$。
- K's補數：先求出N_K的 $(K - 1)$ 's補數，再加上K_{-q}，也就是 $((K - 1 - d_{p-1})(K - 1 - d_{p-2})\cdots(K - 1 - d_1)(K - 1 - d_0).(K -1 - d_{-1})\cdots(K - 1 - d_{-q}))_K + K_{-q}$；或者，您也可以套用公式 $K_p - N_K$。

數值算術運算

- 加法
- 減法
- 乘法
- 除法

加法

- 範例：$111010_2 + 11011_2$

（1）

$$
\begin{array}{r}
00111010 \\
+ \quad 00011011 \\
\hline
\end{array}
$$

（2）

$$
\begin{array}{r}
1 \\
00111010 \\
+ \quad 00011011 \\
\hline
01
\end{array}
$$

加法

（3）

$$\begin{array}{rr} & 1 \\ & 00111010 \\ + & 00011011 \\ \hline & 0101 \end{array}$$

（4）

$$\begin{array}{rr} & 1 \\ & 00111010 \\ + & 00011011 \\ \hline & 10101 \end{array}$$

加法

（5）

$$
\begin{array}{rr}
 & 1 \\
 & 00111010 \\
+ & 00011011 \\
\hline
 & 010101
\end{array}
$$

（6）

$$
\begin{array}{rr}
 & 00111010 \\
+ & 00011011 \\
\hline
 & 01010101
\end{array}
$$

減法

（1） 00001010
- 00000011

（2） -1
00001010
- 00000011

1

減法

（3）   -1

```
      -1
  00001010
- 00000011
----------
        11
```

（4）   -1

```
     -1
  00001010
- 00000011
----------
       111
```

減法

（5） 00001010

$$
\begin{array}{r}
00001010 \\
-\quad 00000011 \\
\hline
00000111
\end{array}
$$

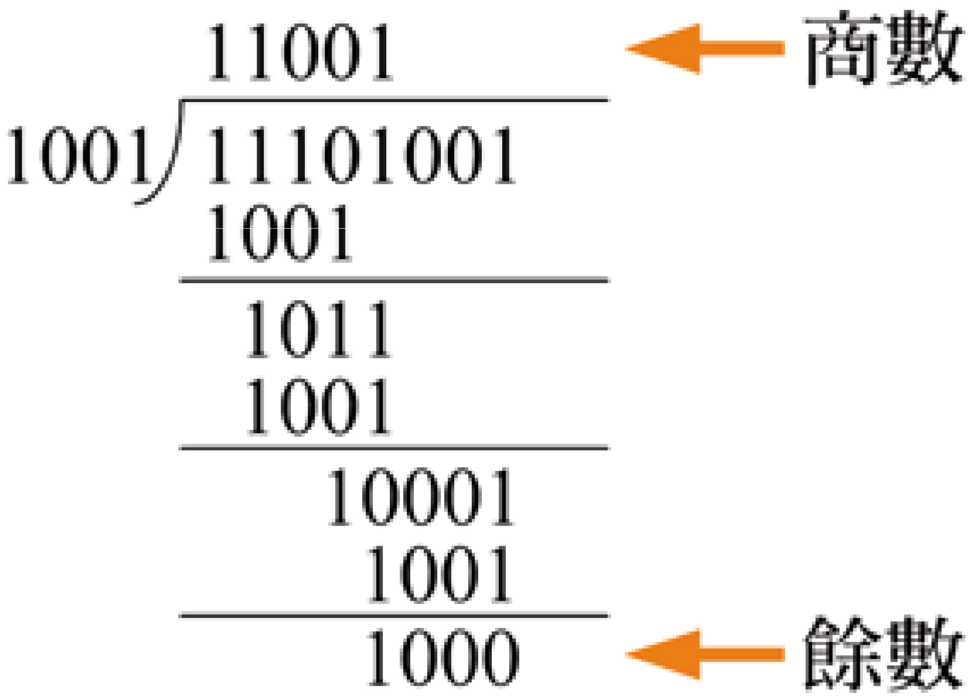
乘法

- 範例：11012 x 10112

```
        1101
   x    1011
------------
        1101
       1101
      0000
     1101
------------
    10011111
```

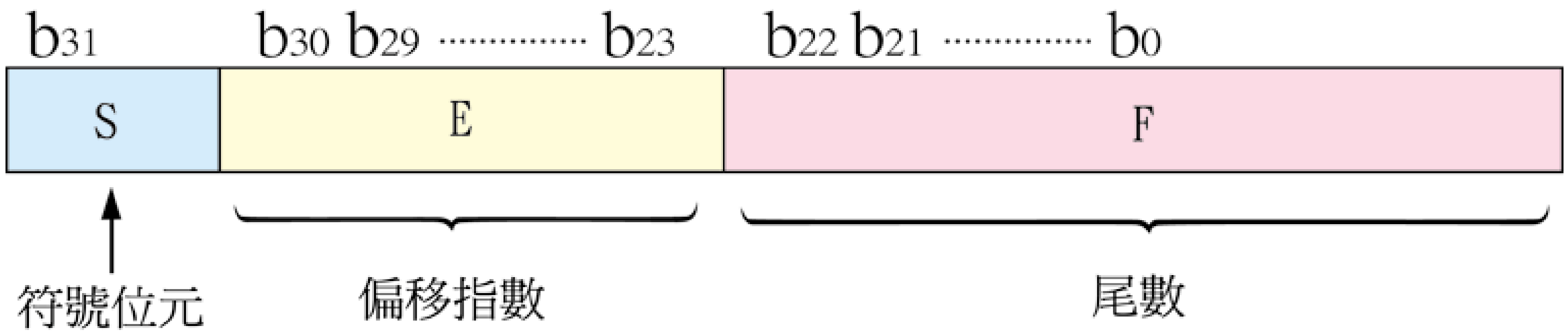
除法

- 範例：111010012 ÷ 10012



數碼系統

- IEEE 754 Single格式
  - 符號位元 (sign bit)
  - 偏移指數 (biased exponent)
  - 尾數 (mantissa)、分數 (fraction)



文字表示法

- ASCII
- ASCII-8
- EBCDIC
- 中文編碼系統；如BIG5 、王安碼、CCCII碼，簡體中文編碼系統以國標碼GB或漢字碼HZ為主。
- Unicode

常見的文字檔格式

- TXT
- DOC/DOCX
- PDF

圖形表示法

- 點陣圖：點陣圖放大時，容易出現鋸齒狀。
- 向量圖：能夠依照任意比例放大、縮小、旋轉及傾斜，而不會出現鋸齒狀。

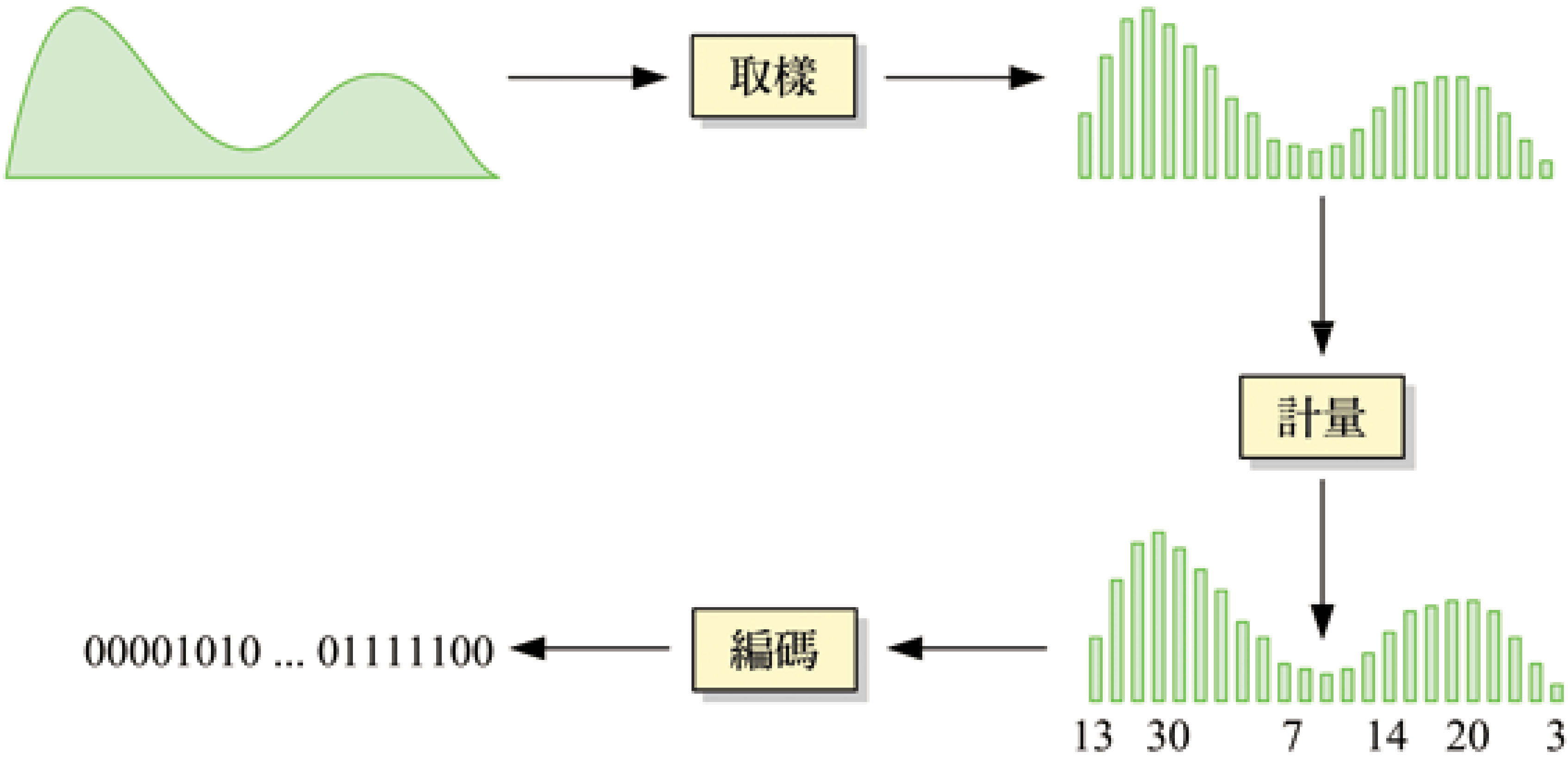
點陣圖

- 水平解析度V.S.垂直解析度
- 圖形尺寸
- 色彩深度
- 常見的點陣圖檔格式
  - BMP
  - JPEG
  - GIF
  - PNG
  - TIFF
  - PSD

向量圖

- 常見的向量圖檔格式
  - EPS
  - DXF/DWG
  - WMF

聲音表示法



聲音表示法

- 常見的聲音檔格式
  - WAV
  - MP3
  - MIDI
  - Real Audio
  - WMA
  - CD-AUDIO
  - Dolby Digital
  - DTS

視訊表示法

- 主要的電視系統視訊標準：
  - NTSC (national television standards committee)
  - PAL (phase alteration line)
  - SECAM (sequential color and memory)
  - HDTV (high definition TV)

視訊表示法

- 常見的視訊檔格式
  - AVI
  - MPEG
  - Quick Time
  - Real Video
  - WMV

資料壓縮

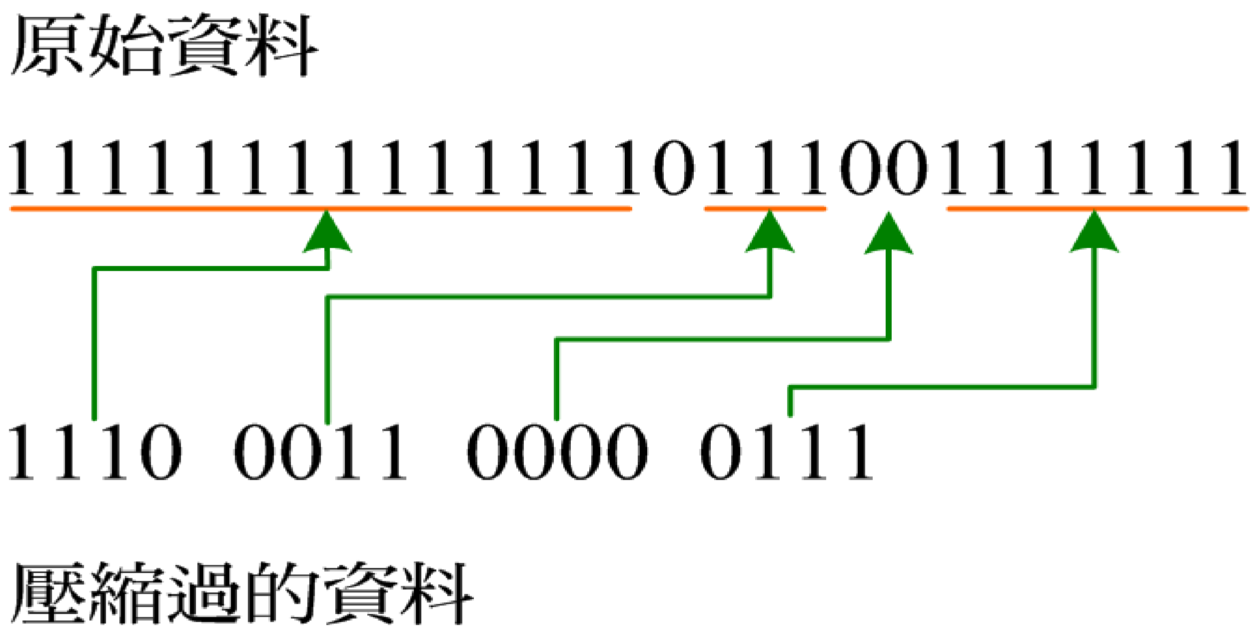
- 非失真壓縮，例如變動長度編碼 (run length encoding)、霍夫曼碼 (Huffman coding)、Lempel-Ziv編碼等。
- 失真壓縮，例如JPEG可用來壓縮圖形、照片，MPEG可用來壓縮影片，MP3可用來壓縮聲音。

資料壓縮

- 常見的壓縮技術
  - JPEG
  - GIF
  - PNG
  - MP3
  - MPEG

變動長度編碼

- 原理是記錄符號出現的次數，例如：



霍夫曼碼

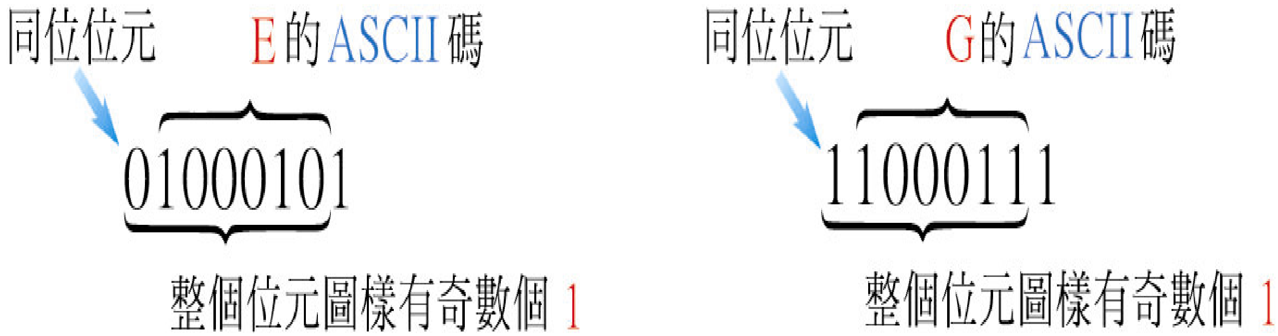
- 編碼步驟如下：
  - 找出所有符號的出現頻率。
  - 將頻率最低的兩者相加得出另一個頻率。
  - 重覆步驟2不斷將頻率最低的兩者相加，直到只剩下一個頻率為止。
  - 根據合併的關係分別配置0和1，形成一個編碼樹 。

誤差與錯誤檢查

- 固有誤差 (inherent error)
- 捨棄誤差 (round-off error)

同位元檢查

- 在資料位元傳送出前，先加上一個同位位元(通常加在最前面)，然後送出，待接收到這些位元圖樣後，檢查看看是否有奇數個1或偶數個1。
- 分成奇同位檢查和偶同位檢查。



循環冗餘碼 (CRC)

- 讓發訊端與收訊端事先協調一個生成多項式，然後發訊端在將資料位元傳送出去之前，先將資料位元除以生成多項式，再將得到的餘數(即CRC碼) 放在資料位元的後面一起傳送出去。

循環冗餘碼 (CRC)

假設資料位元為1100101011110，生成多項式為$X^3 + 1$ (1001)，試求取CRC碼及加上CRC碼後的完整訊息：

1. 由於生成多項式$X^3 + 1$ (1001) 的冪次為3，故先在資料位元1100101011110的後面加上三個0，得到被除數為1100101011110000。

循環冗餘碼 (CRC)

2. 以長除法求取110010101110000除以生成多項式$X^3 + 1$ (1001)的餘數：

```
              101101000101
       ┌─────────────────────
  1001 │ 110010101110000
         1001
         ─────────────────
          01110
           1001
         ─────────────────
           01011
            1001
         ─────────────────
             001001
               1001
         ─────────────────
               00001100
                   1001
         ─────────────────
                   001100
                     1001
         ─────────────────
                       11   ← 餘數
```

3. CRC碼為餘數11，故完整訊息為110010101110011。

錯誤更正碼 (ECC)

- 當錯誤更正碼的**漢明距離**大於等於D時，只要發生錯誤的位元不超過D - 1個，系統都能夠偵測出來，而只要發生錯誤的位元不超過 (D - 1) / 2個，系統都能夠加以更正。