

# 第 3 課：數據表示 (Data Representation)

## 3.1 數字系統與表示法

- 數據單位： 基本單位是位元 (bit/b)，可為 0 或 1。字節 (byte/B) 等於 8 位元。

表 3.1 數據單位列表

單位	記號	數值
位元	bit, b	數據的基本單位。它可以是 0 或 1。
字節/位元組	B	$1 \text{ B} = 8 \text{ bits}$
千字節	KB	$1 \text{ KB} = 2^{10} \text{ B} = 1024 \text{ B}$
百萬字節	MB	$1 \text{ MB} = 2^{10} \text{ KB} = 1024 \text{ KB}$
十億字節	GB	$1 \text{ GB} = 2^{10} \text{ MB} = 1024 \text{ MB}$
萬億字節	TB	$1 \text{ TB} = 2^{10} \text{ GB} = 1024 \text{ GB}$

- 位元速率單位： 例如 Mbps (每秒百萬位元)。

表 3.2 位元速率單位列表

單位	記號	數值
每秒位元	bps	每秒傳輸 1 位元的數據。
每秒千位元	kbps, Kb/s	$1 \text{ kbps} = 1000 \text{ bps}$
每秒百萬位元	Mbps, Mbit/s, Mb/s	$1 \text{ Mbps} = 1000 \text{ kbps}$
每秒十億位元	Gbps, Gbit/s, Gb/s	$1 \text{ Gbps} = 1000 \text{ Mbps}$
每秒萬億位元	Tbps, Tbit/s, Tb/s	$1 \text{ Tbps} = 1000 \text{ Gbps}$

- 數字系統： 電腦設備使用的是二進制 (基數 2, 0和1)，另有十進制 (基數 10) 和十六進制 (基數 16, 使用 0–9 和 A–F)。

表 3.7 十進制、二進制和十六進制的對照

二進制	0000	0001	0010	0011	0100	0101	0110	0111
十進制	0	1	2	3	4	5	6	7
十六進制	0	1	2	3	4	5	6	7

二進制	1000	1001	1010	1011	1100	1101	1110	1111
十進制	8	9	10	11	12	13	14	15
十六進制	8	9	A	B	C	D	E	F

## 二進制轉換

### 1. 二進制轉十進制 (Binary to Decimal)

概念：權重法。將二進制數的每一位數字乘以  $2^n$ ，其中  $n$  是該位數字的權重（從右邊的個位數開始，權重為  $0, 1, 2, 3, \dots$ ），然後將所有結果相加。

詳細步驟：

1. 從二進制數的最右邊一位（個位數）開始，將其位置的權重定為  $2^0$ 。
2. 向左移動一位，權重依序增加  $2^1, 2^2, 2^3, \dots$ 。
3. 將每一位的數字（0 或 1）與其對應的權重相乘。
4. 將所有乘積的結果相加，即為十進制數。

例子： $10110_2$

- $1 \times 2^4 = 16$
- $0 \times 2^3 = 0$
- $1 \times 2^2 = 4$
- $1 \times 2^1 = 2$
- $0 \times 2^0 = 0$
- 總和： $16 + 0 + 4 + 2 + 0 = 22_{10}$

## 2. 十進制轉二進制 (Decimal to Binary)

概念：短除法（除以 2 取餘數法）。將十進制數連續除以 2，直到商為 0，然後將每次得到的餘數從下往上排列。

詳細步驟：

1. 將十進制數除以 2，得到商和餘數。
2. 將上一步的商作為新的被除數，繼續除以 2，記錄新的商和餘數。
3. 重複此過程，直到商為 0 為止。
4. 將所有的餘數從最後一個（最底下的）餘數開始，依序寫到第一個（最頂上的）餘數，即為二進制數。

計算方法 (範例： $25_{10}$ )：

1.  $25 \div 2 = 12 \dots 1$  (LSB – 最低位)
2.  $12 \div 2 = 6 \dots 0$
3.  $6 \div 2 = 3 \dots 0$
4.  $3 \div 2 = 1 \dots 1$
5.  $1 \div 2 = 0 \dots 1$  (MSB – 最高位)

- 餘數由下往上：11001
- 因此， $25_{10} = 11001_2$ 。

表 3.3 十進制數字轉為 3 位元的二進制數字

十進制	0	1	2	3	4	5	6	7
二進制	000	001	010	011	100	101	110	111

### 3. 十六進制轉十進制 (Hexadecimal to Decimal)

概念：權重法。將十六進制數的每一位數字乘以  $16^n$ ，其中  $n$  是該位數字的權重（從右邊的個位數開始，權重為  $0, 1, 2, 3, \dots$ ），然後將所有結果相加。

詳細步驟：

1. 將十六進制數中的字母 (A–F) 轉換為對應的十進制數 (10–15)。
2. 從最右邊一位開始，將其位置的權重定為  $16^0$ 。
3. 向左移動一位，權重依序增加  $16^1, 16^2, 16^3, \dots$ 。
4. 將每一位的十進制數與其對應的權重相乘。
5. 將所有乘積的結果相加，即為十進制數。

計算方法 (範例： $A5_{16}$ )：

- $A = 10_{10}, 5 = 5_{10}$
- 右邊第一位 (5) :  $5 \times 16^0 = 5 \times 1 = 5$
- 左邊第二位 (A) :  $10 \times 16^1 = 10 \times 16 = 160$
- 總和：  $160 + 5 = 165_{10}$
- 因此， $A5_{16} = 165_{10}$ 。

## 4. 十進制轉十六進制 (Decimal to Hexadecimal)

概念：短除法（除以 16 取餘數法）。將十進制數連續除以 16，直到商為 0，然後將每次得到的餘數轉換為十六進制數字，並從下往上排列。

詳細步驟：

1. 將十進制數除以 16，得到商和餘數。
2. 將上一步的商作為新的被除數，繼續除以 16，記錄新的商和餘數。
3. 重複此過程，直到商為 0 為止。
4. 將所有餘數（0–15）轉換為對應的十六進制數字（0–9, A–F）。
5. 將轉換後的十六進制數字從最後一個（最底下的）餘數開始，依序寫到第一個（最頂上的）餘數，即為十六進制數。

計算方法（範例： $450_{10}$ ）：

$$1. 450 \div 16 = 28 \dots 2$$

$$2. 28 \div 16 = 1 \dots 12$$

$$3. 1 \div 16 = 0 \dots 1$$

4. 餘數轉換為十六進制：

- $2 \rightarrow 2$
  - $12 \rightarrow C$
  - $1 \rightarrow 1$
- 
- 餘數由下往上： $1C2$
  - 因此， $450_{10} = 1C2_{16}$ 。

## 樣式數量

- 樣式數量 (Patterns):  $b$  個位元可表示  $2^b$  個獨特的樣式。所需的位元數是  $\log_2 P$ ，其中  $P$  是所需的樣式數。
- 整數表示：
  - 無符號整數 (Unsigned integer): 指沒有負號的整數（正數和零）。
  - 有符號整數 (Signed integer): 帶有負號的整數（正數、負數和零）。
  - 二進制補碼 (Two's complement): 是一種以位元樣式表示有符號整數的方法，通常用於電腦運算。整數  $X$  的二進制補碼是  $-X$  的二進制補碼表示。
- 溢出誤差 (Overflow Error): 當計算結果超出位元數可表示的範圍時，就會發生。對於有符號整數而言，僅在兩個正數相加或兩個負數相加時才會出現溢出誤差。

表 3.8 三位元二進制補碼數字

十進制	-4	-3	-2	-1	0	1	2	3
補碼	100	101	110	111	000	001	010	011

表 3.9 有符號整數補碼數字/補碼

十進制	-4	-3	-2	-1	0	1	2	3	超出範圍
補碼數字	100	101	110	111	000	001			

## 3.2 字符編碼 (Character Encoding)

字符編碼系統用於表示數字、字母和符號。

- ASCII (美國信息交換標準碼): 英語常用，通常使用 7 位元表示一個字符。
- 大五碼 (Big-5) 和國標碼 (Guobiao, GB): 大五碼通常用於繁體中文，國標碼用於簡體中文。兩者皆使用 2 字節 (16 位元) 來儲存一個字符。
- 統一碼/萬國碼 (Unicode): 具備表示多種語言的能力，是可變長度編碼，通常使用 1 到 4 個字節來表示一個字符。

## 3.3 條碼編碼 (Barcode Encoding)

- 條碼 (Barcode): 由不同寬度的條形和空格組成，用於表示產品編號的數字。優勢是減少輸入錯誤和輸入時間。條碼大多只能從兩個角度掃描，即  $0^\circ$  或  $180^\circ$ 。
- 二維碼 (Quick Response code, QR code): 是一種二維條碼，可以從任何方向掃描，具有更高的糾正錯誤能力，通常可恢復達 30% 的數據。二維碼可表示統一碼字符，可代表的數據量上限不少於 1000 個字符。

## 3.4 模擬及數碼數據 (Analog and Digital Data)

- 模擬數據 (Analog data): 以物理方式用連續值表示的數據，例如聲波和圖畫。
- 數碼數據 (Digital data): 以 0 和 1 等離散符號表示的數據。
- 數碼化 (Digitalisation): 將模擬數據轉換為數碼數據。
  - 離散化 (Discretisation): 在固定時間間隔內對訊號值進行取樣。
  - 量化 (Quantisation): 離散化後，取樣的數值會被四捨五入為一組特定的離散值，例如整數。

## 3.5 多媒體 (Multimedia)

多媒體檔案可儲存各種檔案格式，它們具有不同的屬性。

- **壓縮 (Compression):** 用於減少檔案大小以節省儲存空間和傳輸頻寬。
  - **無損壓縮 (Lossless compression):** 解壓縮後可完全恢復原始檔案（例如 PNG、GIF、FLAC、ZIP、RAR、7Z）。
  - **有損壓縮 (Lossy compression):** 數據會有所損失，解壓縮後的檔案與原始檔案不同（例如 JPG、MP3）。

### 圖像格式：

表 3.24 點陣圖和向量圖之間的比較

種類	優勝之處
點陣圖	<ul style="list-style-type: none"><li>– 寫實數碼照片通常為點陣圖。</li><li>– 可以用常見的圖像編輯軟件（如 Microsoft 小畫家）編輯。</li></ul>
向量圖	<ul style="list-style-type: none"><li>– 其檔案多數較小。</li><li>– 縮放圖像時，它的邊緣不會顯得粗糙或變得模糊。</li></ul>

- **圖像屬性：** 色深（表示一個像素中顏色的位元數）、解像度/解析度（圖像中像素的總數）、長寬比。
- **未壓縮點陣圖檔案大小：** 色深  $\times$  解像度  $\times$  寬度（像素） $\times$  高度（像素）。

## 圖像檔案格式列表

檔案格式	類型	支援透明	支援動畫	壓縮	備註
BMP	點陣圖	✗	✗	沒有壓縮	檔案相對較大。通常支援 24 位元色深。
JPEG / JPG	點陣圖	✗	✗	有損壓縮	檔案相對較小。通常支援 24 位元色深。
GIF	點陣圖	✓	✓	無損壓縮	通常支援 8 位元色深。
PNG	點陣圖	✓	✗	無損壓縮	支援 48 位元色深甚至更高。APNG 是 PNG 的擴展，它支援動畫。
WebP	點陣圖	✓	✓	有損或無損壓縮	檔案相對較小。通常支援 24 位元色深。
TIFF / TIF	點陣圖	✓	✗	無損壓縮	TIFF 支援圖層和多頁圖像。
SVG	向量圖	✓	✓	沒有壓縮	支援無損壓縮。壓縮後的 SVG 稱為 SVGZ。

## |音頻格式：

- 音頻屬性：位元深度（位元數越多，音頻品質越高）、取樣頻率（每秒採集音頻樣本數）、聲道數。
- 未壓縮的音頻檔案大小：位元深度×取樣頻率×聲道數目×音頻時長。

### 音頻檔案格式列表

檔案格式	壓縮	備註
WAV / WAVE	未經壓縮	高音質。檔案相對較大。
MP3、AAC、OGG	有損壓縮	檔案相對較小。
WMA	有損或無損壓縮	音質與檔案大小取決於壓縮方式。
FLAC、ALAC	無損壓縮	高音質。檔案相對較大。
MIDI	未經壓縮	檔案相對較小。儲存音符或數碼取樣聲音的數據，而非人聲。

## 視頻格式：

- 視頻格式：視頻是圖像和音效的組合。
  - MP4 (MPEG-4)：有損壓縮，壓縮率高，廣泛用於串流服務。
  - AVI：未經壓縮，檔案相對較大。
  - 視頻屬性：色深、解像度、幀速率 (frames per second, fps) 和長度。
  - 未壓縮視頻檔案大小：色深 × 解像度 × 幀速率 × 時長 + 音軌大小。

表 3.30 視頻檔案格式列表

檔案 格式	編碼 – 語碼器 (codec)	可支援 的瀏覽 器	檔案大小	功能特點及應用方式
MP 4	MPEG-4	大多數	相對小、 壓縮率高	影片串流服務
MP 4	H.264 (AVC)	大多數	比 MPEG-4 小	串流直播、高清和全高清影片串流
MP 4	H.265 (HEVC)	很少	比 H.264 小	影像品質較高，提供解像度高於 2K 的 影片串流服務 (Amazon、Apple TV+、Disney+ 和 Netflix 正進行使用)
We bM	VP8	大多數 最新版 瀏覽器	比 H.264 小、壓縮 率更高	影片串流、串流直播

## 文本格式：

- 純文字 (Plain text)：例如 TXT。
- 格式化文本 (Formatted text)：例如 DOC/DOCX、PDF。PDF 檔案提供一致的排版樣式，適用於學歷證書或說明書等需要保持原始外觀的文件。

表 3.23 文字檔案格式列表

檔案格式	類型	備註
TXT	純文字	檔案大小相對較小。獲不同操作系統支援。
RTF	格式化文本	獲不同操作系統支援。可以包含圖像和超連結。
DOC /DO CX	格式化文本	首先在 Microsoft Word 建立。可以包含多媒體，包括圖像、音頻、視頻和超連結。DOC 檔案由 Microsoft Word 2003 或之前版本建立，DOCX 檔案則由 Microsoft Word 2007 或之後版本建立。
ODT	格式化文本	首先在 OpenOffice Writer 和 LibreOffice 建立。可以包含多媒體，包括圖像、音頻、視頻和超連結。
PAGES	格式化文本	首先在 Apple Pages 建立。可以包含多媒體，包括圖像、音頻、視頻和超連結。
PDF	格式化文本	首先在 Adobe Acrobat 建立，但通常從其他檔案格式匯出。可以包含多媒體，包括圖像、音頻、視頻和超連結。在不同設備和操作系統皆能保持一致的排版樣式和文字格式，且它是跨平台的格式。