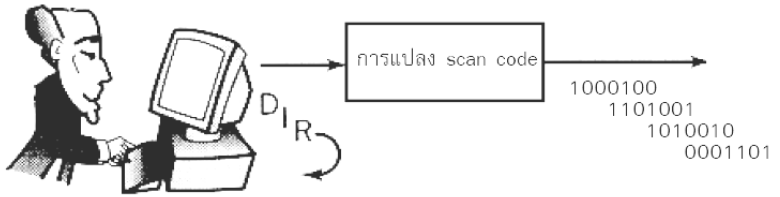
ระบบคอมพิวเตอร์และสถาปัตยกรรม (Computer System and Architecture)

บทที่ 2
ข้อมูล (Data)

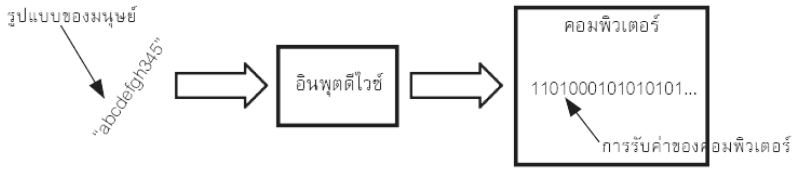


ข้อมูล

ทำงานของคอมพิวเตอร์มี 3 ขั้นตอนหลักคือ การรับข้อมูลเข้า หรืออินพุต (Input) การดำเนินการ (Process) และการนำ ข้อมูลออก หรือเอาต์พุต (Output)

พื้นฐานข้อมูล

ข้อมูลไม่ว่าจะเป็นคาแรกเตอร์ รูปภาพ เสียง หรือข้อมูลในรูปแบบ ต่าง ๆ จะต้องสามารถนำเข้าสู่คอมพิวเตอร์ และแปลงให้อยู่ในรูปแบบที่เหมาะสมด้วยอุปกรณ์อินพุตที่ให้คอมพิวเตอร์สามารถดำเนินการ จัดเก็บ หรือนำไปใช้ในระบบคอมพิวเตอร์ได้



ฟอร์แมตมาตรฐานของข้อมูล

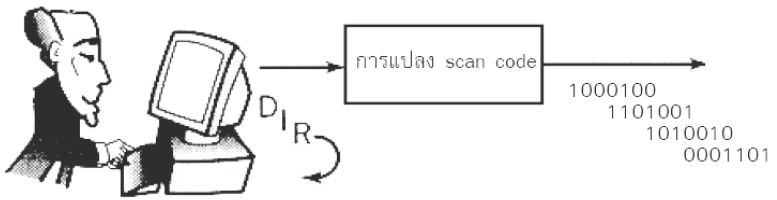
ฟอร์แมตข้อมูล	มาตรฐาน
คาแรกเตอร์	Unicode, ASCII, EBCDIC
รูปภาพ (บิตแมพ)	GIF (Graphical Image Format), TIFF (Tagged Image File Format), PNG (Portable Network Graphics)
รูปภาพ (ออปเจ็กต์)	PostScript, JPEG (Joint Photographic Experts Group). SWF (Macromedia Flash), SVG (Scalable Vector Graphics)
กราฟิกและฟอนต์	PostScript, TrueType
เสียง	WAV (Waveform Audio), AVI (Audio Visual Interleaved), MP3 MPEG (Moving Picture Experts Group)
(audio layer) 3, เพจ	MIDI (Musical Instrument Digital Interface), WMA PDF (Acrobat Portable Document Format), HTML XML (eXtension Markup Language)
ภาพเคลื่อนไหว	Quicktime, MPEG-2 (Moving Picture Experts Group -2), RealVideo, WMV

Character

- ASCII
- ASCII กับคาแรกเตอร์ภาษาไทย มาตรฐาน สมอ.
- EBCDIC
- Unicode

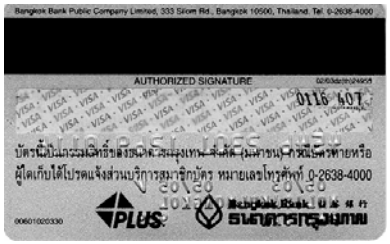
การใส่ตัวอักษรจากแป้นพิมพ์

- ตัวอักษรที่ใส่ผ่านแป้นพิมพ์จะถูกแปลง scan code แล้วส่งเข้าไป
- สมมุติว่าผู้ใช้งานคีย์ตัวอักษร 3 ตัว “D”, “I”, “R” แล้วกดคีย์ Enter คอมพิวเตอร์จะแปลง scan code 4 ตัวเป็นโค้ด ASCII ฐานสองเป็น 1000100, 1001001, 1010010, 0001101



การใส่ตัวอักษรจากแหล่งอื่น

- OCR (Optical Character Recognition)
- Barcode Reader
- Magnetic Stripe Reader
- Voice Input



เลขจำนวนเต็ม

- ในระบบเลขฐานสอง สามารถแสดงค่าโดยใช้เลขเพียง 2 ตัวคือ 0 และ 1 โดยอาจจะใช้เครื่องหมาย + และ - รวมทั้งแทนจำนวนทศนิยมได้ เช่น

  1000011 = 35 หรือ -1011.0110 = -11.375

- แต่ในคอมพิวเตอร์จะรู้จักเฉพาะเลข 0 และ 1 ไม่สามารถใช้ประโยชน์จากจุดได้ แต่สามารถใช้เครื่องหมายลบโดยใช้เงื่อนไขเพิ่มเติม การเก็บข้อมูลจะเป็นการเก็บค่าในรีจิสเตอร์ เช่น ถ้าเป็นขนาด 8 บิต จะได้

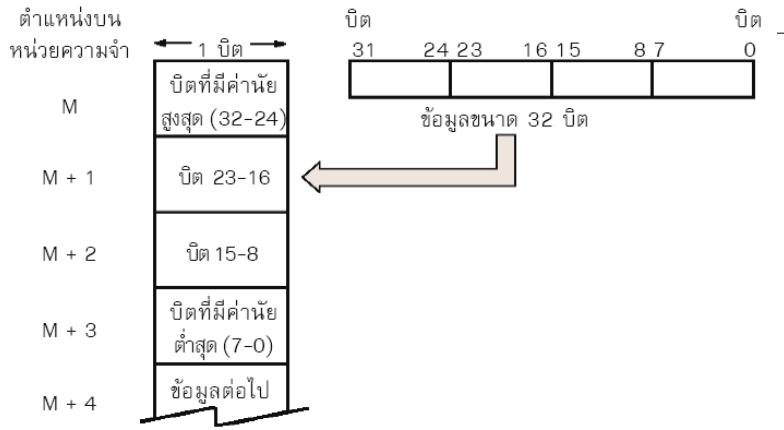
  00000000 = 0 00000001 = 1
  01000001 = 65 10000001 = 129
  10000011 = 131 11111111 = 255

เลขจำนวนเต็มไม่มีเครื่องหมาย

- การเก็บข้อมูลในคอมพิวเตอร์ที่เป็นเลขจำนวนเต็มไม่มีเครื่องหมายจะเก็บเลขฐานสอง ตามจำนวนบิตที่กำหนด เช่น

  8 บิต = 256 ค่า (0-255)
  16 บิต = 65,536 (0-65,535)
  16 บิต = 4,294,967,296 (0-4,294,967,295)

เลขจำนวนเต็มไม่มีเครื่องหมาย



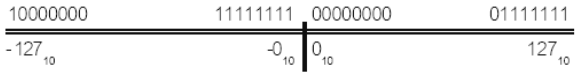
เลขจำนวนเต็มไม่มีเครื่องหมาย

เลขจำนวนเต็มแบบ BCD

จำนวนบิต	ขอบเขต BCD		ขอบเขต Binary	
4	0 - 9	1 digit	0 - 15	1 + digit
8	0 - 99	2 digit	0 - 255	2 + digit
12	0 - 999	3 digit	0 - 4,095	3 + digit
16	0 - 9,999	4 digit	0 - 65,535	4 + digit
20	0 - 99,999	5 digit	0 - 1 Million	6 digit
24	0 - 999,999	6 digit	0 - 16 Million	7 + digit
32	0 - 99,999,999	8 digit	0 - 4 Billion	9 + digit
64	$0-(10^{16}-1)$	16 digit	0 - 16 Billion	19 + digit

เลขจำนวนเต็มที่มีเครื่องหมาย

- **Sign-and-magnitude :** คล้ายเลขจำนวนเต็มทั่วไป แต่บิตซ้ายสุดเป็นบิตเครื่องหมาย
  เลข 0 หมายถึงจำนวนเต็มบวก และเลข 1 เป็นจำนวนเต็มลบ
  00110101 = +53
  10110101 = -53
- **1's complement** : เกิดจากนำเลขจำนวนเต็มฐานสองจำนวนนั้นลบออกจากค่าที่เป็น 1 ทุกบิต เช่น 1's complement ของ 10110101 คือ
  11111111-10110101 = 01001010 หรือ สลับค่าทุกบิตเป็นค่าตรงข้าม
  10110101 ---> 01001010



เลขจำนวนเต็มที่มีเครื่องหมาย

- **2's complement** : เกิดจากบวก 1 เข้ากับบิตขวาสุดของ 1's complement เช่นหาค่า 2's complement ของ 10110100 คือ
  ค่า 1's complement ของ **10110100** = **01001011**
  ค่า 2's complement ของ **10110100** = **01001011+1**
  = 01001100

เลขจำนวนเต็มที่มีเครื่องหมาย

- **2's complement :**

128	64	32	16	8	4	2	1
0	0	1	0	1	0	0	1

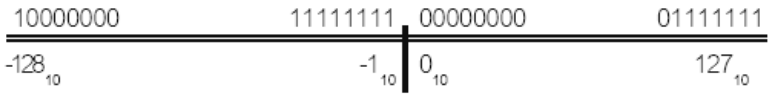
ค่า value box ของจำนวน + ขนาด 8 บิต

ค่า 00101001 จะเท่ากับ $32+8+1 = 41)_{10}$

-128	64	32	16	8	4	2	1
1	1	0	0	1	0	1	1

ค่า value box ของจำนวน - ขนาด 8 บิต

มีค่า = -128+64+8+2+1 = -53



โครงสร้างการแสดงค่า 2's complement

เลขจำนวนเต็มที่มีเครื่องหมาย

- **Overflow และตัวทด :**

(+4)+(+2)
0100
0010
0110 = (+7) ← ผลลัพธ์ถูก
(ก)

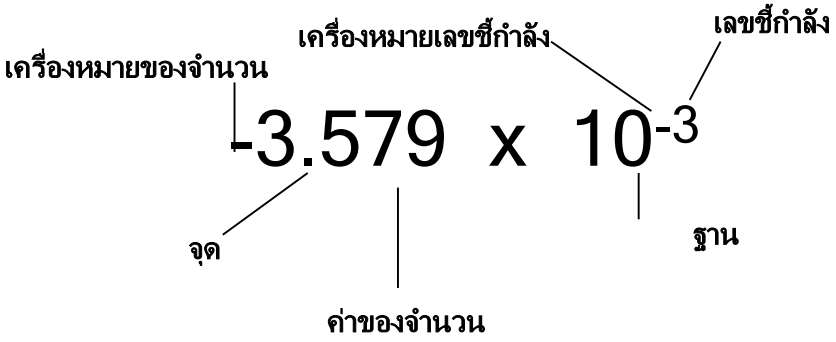
(+4)+(+6)
0100
0110
1010 = (-6) ← Overflow, ผลลัพธ์ผิด
(ข)

(-4)+(-2)
1100
1110
1 1010 = (-6) ← ผลลัพธ์ถูก
(ค)

(-4)+(-6)
1100
1010
1 0110 = (+3) ← Overflow, ผลลัพธ์ผิด
(ง)

เลขทศนิยม

- **พื้นฐานเลขทศนิยม :**



เลขทศนิยม

- **เลขทศนิยม** : จำนวนเต็มที่เป็นมาตรฐาน 1 word ประกอบด้วยเลข 7 ตัวและเครื่องหมายอีก 1 ตัว รูปแบบจะเป็น

SMMMMMMM

โดยที่ S คือ Sign ซึ่งเป็นบิตที่เก็บเครื่องหมาย

M คือ Mantissa ซึ่งเป็นบิตค่าของจำนวนเต็ม

รูปแบบนี้สามารถเก็บค่าจำนวนเต็มในช่วง

$-9{,}999{,}999 < I < 99{,}999{,}999$

เลขทศนิยม

- **เลขทศนิยมที่มีเครื่องหมาย** : จะมีบิตเครื่องหมาย 2 บิต รูปแบบจะเป็น

SEEMMMMM

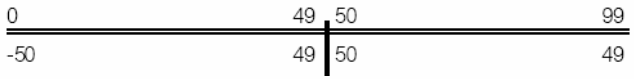
โดยที่ S คือ Sign ซึ่งเป็นบิตที่เก็บเครื่องหมาย

E คือ Exponent ซึ่งเป็นบิตที่เก็บเลขชี้กำลัง

M คือ Mantissa ซึ่งเป็นบิตค่าของจำนวนทศนิยม

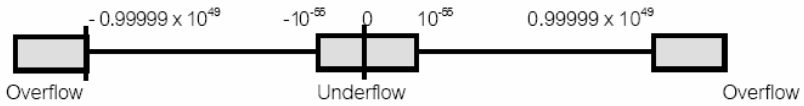
เลขทศนิยม

- **Excess-50** : กำหนดเป็น 100 ช่วง โดยแบ่งครึ่งหนึ่งเป็นบวก อีกครึ่งหนึ่งเป็นลบ รูปแบบจะเป็น



$0.00001 \times 10^{-50} < \text{number} < 0.99999 \times 10^{+49}$

- **Overflow และ Underflow** :



เลขทศนิยม

- **การทำ Normalization** : เป็นการกำหนดรูปแบบของเลขทศนิยม เพื่อให้เกิดความแม่นยำ โดยการเลื่อนจำนวนไปทางซ้ายโดยการเพิ่มเลขชี้กำลังจนกว่า 0 ที่นำหน้าจำนวนนั้นถูกกำจัดออกไป รูปแบบมาตรฐานจะเป็น

$$.MMMMM \times 10^{EE}$$

เช่น $123.4567 = 123.4567 \times 10^{0}$

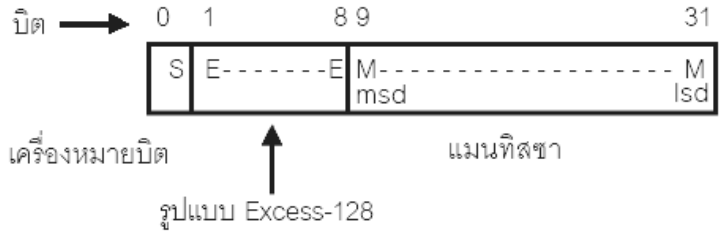
$= 0.1234567 \times 10^{3}$

$= 0.12345 \times 10^{3}$

$= 05312345$

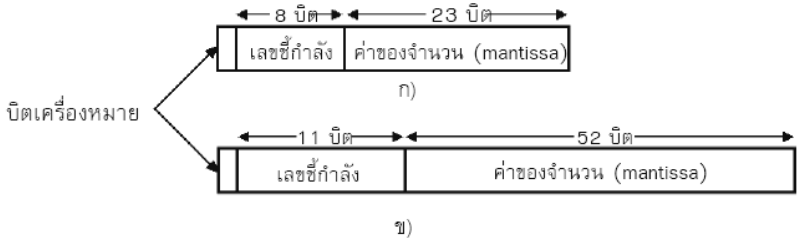
เลขทศนิยม

- **เลขทศนิยมในคอมพิวเตอร์** : นำรูปแบบเลขทศนิยมที่มีเครื่องหมายมาประยุกต์ใช้งาน ถ้าใช้ขนาด 32 บิต จะใช้บิตเครื่องหมาย 1 บิต, เลขชี้กำลัง 8 บิต (ใช้ excess-128 ในช่วง 10^{-128} ถึง 10^{+127}) และค่าของจำนวนใช้ 23 บิต (ระหว่าง 10^{-38} ถึง 10^{+38}) รูปแบบจะเป็น



เลขทศนิยม

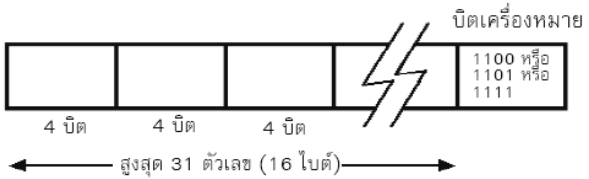
- **เลขทศนิยมตามมาตรฐาน IEEE754** : มาตรฐานการเก็บจำนวนทศนิยมในคอมพิวเตอร์ รูปแบบจะเป็น



รูปแบบตามมาตรฐาน IEEE754 ก) 32 บิต ข) 64 บิต

เลขทศนิยม

- **Packed Decimal** : แต่ละเลขฐานสิบจะเก็บในรูปแบบ BCD ที่ใช้ 2 ตัวเลขเป็น 1 ไบต์ ตัวเลขที่มีค่านัยสำคัญสูงสุดจะเก็บไว้ก่อนในบิตที่มีค่ามากของไบต์แรก เครื่องหมายถูกเก็บอยู่ในบิตที่มีค่าต่ำของไบต์สุดท้าย สามารถเก็บค่าได้ถึง 31 ตัวเลขค่าเลขฐานสอง 1100 และ 1101 ใช้แสดงเครื่องหมาย + และ - ตามลำดับ ส่วนค่า 1111 ใช้เพื่อกำหนดว่าจำนวนนั้นไม่มีเครื่องหมาย รูปแบบจะเป็น

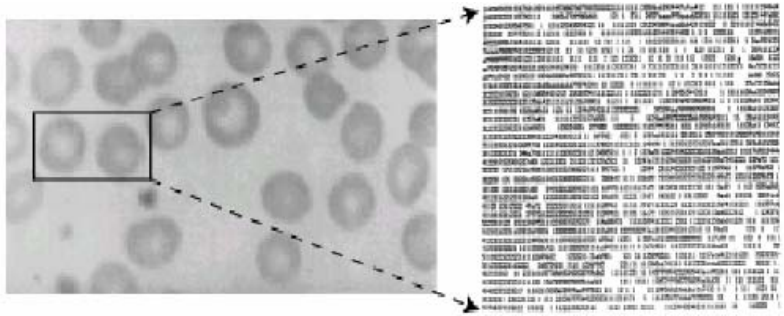


ข้อมูลรูปภาพ

- Raster image --> Bitmap, GIF, JPG
- Vector image --> Graphical object, Object image
- SVG (Scalable Vector Graphics)
- Macromedia Flash

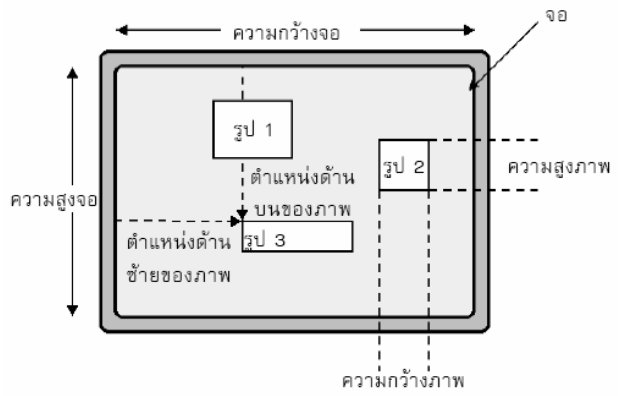
ข้อมูลรูปภาพ

- รูปภาพแบบBitmap



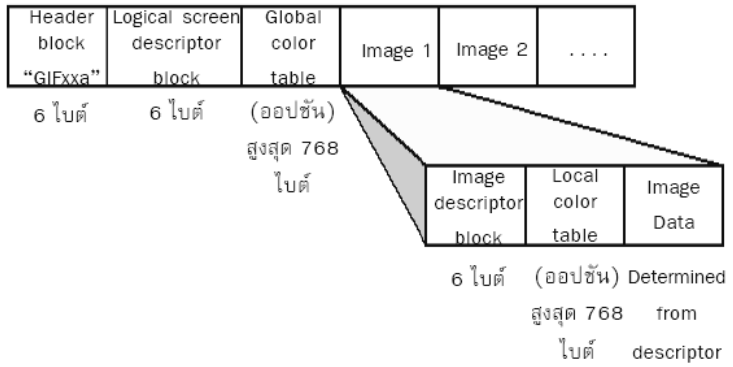
ข้อมูลรูปภาพ

- รูปภาพแบบ GIF



ข้อมูลรูปภาพ

- รูปภาพแบบ GIF



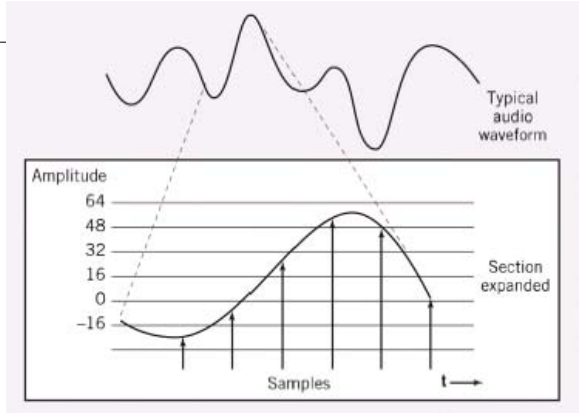
ข้อมูลรูปภาพ

- รูปภาพแบบ Object

ข้อมูลรูปภาพ

- รูปภาพแบบ Postscript
- การแสดงCharacterเป็นรูปภาพ
- รูปภาพแบบวิดีโอ

ข้อมูลเสียง



การแปลงสัญญาณอนาล็อกของคลื่นเสียงเป็นดิจิตอล

ข้อมูลที่ถูกบีบอัด

- Lossless
- Lossy

0 5 5 7 4 7 0 0 0 0 1 4 7 4 7 9 1 0 0 0 0 0 6 6 8 2 7 4 7 7 4 7 0 0 0 7 4 7 0 0 7 4 7 1 2 3 0 0 0 6...

0 1 5 5 7 4 7 0 4 1 4 7 4 7 9 1 0 5 6 6 8 2 7 4 7 7 4 7 0 3 7 4 7 0 2 7 4 7 1 2 3 0 3 6...

0 1 5 5 Z 0 4 1 4 Z 9 1 0 5 6 6 8 2 Z Z 0 3 Z 0 2 Z 1 2 3 0 3 6...