

บทที่ 2

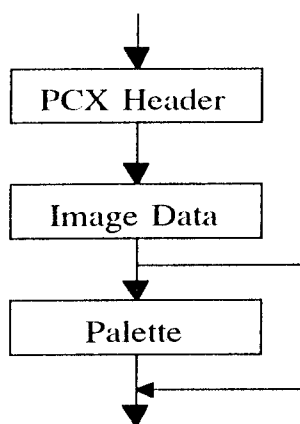
แฟ้มข้อมูลภาพ

2.1 ส่วนประกอบของแฟ้มข้อมูลแบบ PCX

ส่วนประกอบของ FILE.PCX จะประกอบด้วย

- Header
- Image Data
- palette

ดังแสดงในรูป 2.1



รูปที่ 2.1 ลักษณะทั่วไปของ Data Stream ของ PCX format

- HEADER จะเป็นส่วนที่ใช้บอกลักษณะของรูปภาพ ว่ามีขนาดความกว้างเท่าไร ยาวเท่าไร มี FORMAT ชนิดใด มีสีเท่าไร

- ข้อมูลภาพ จะเป็นส่วนที่เก็บข้อมูลของภาพ โดยจะมีการเข้ารหัส ซึ่ง FILE.PCX จะมีการบีบข้อมูลแบบ RLE (RUN LENGTH ENCODING) โดยมีหลักในการบีบ คือข้อมูลภาพที่ซ้ำกันจะมีตัวนำหน้าข้อมูลตัวที่ซ้ำ โดยตัวนำจะมีค่าได้ไม่เกิน 63 ถ้าซ้ำกันเกิน 63 ตัวจะต้องแบ่งข้อมูลเก็บทีละ 63 ตัว แต่ถ้าข้อมูลไม่ซ้ำก็จะเก็บลง FILE.PCX เลย

- แพลเลตต์ เป็นตัวกำหนดค่าสี คือในการเก็บภาพแต่ละภาพไม่สามารถที่จะเก็บค่าสีทั้งหมดได้ อย่างเช่น ถ้าแม่สีแต่ละสีมีระดับอยู่ 64 ระดับสี ซึ่งเมื่อนำมาผสมกันก็จะได้สีทั้งหมด $64 \times 64 \times 64 = 262144$ สี ซึ่งในภาพแต่ละภาพจะมีการใช้สีไม่หมดทุกสี ดังนั้น แพลเลตต์ก็จะเก็บเฉพาะสีที่ภาพนั้นจะใช้เท่านั้น ซึ่งถ้ามีการเปลี่ยนข้อมูลในส่วนนี้ก็จะทำให้สีในภาพมีการเปลี่ยนแปลง โดยที่ไม่ได้มีการเปลี่ยนแปลงข้อมูลภาพ ตัวอย่างเช่น การที่สีของท้องฟ้าค่อยๆ สว่างขึ้น ก็เกิดจากการที่มีการเปลี่ยนแปลงตารางสีในส่วนของ แพลเลตต์

2.2 โครงสร้างหลักของไฟล์ .PCX

ไฟล์ .PCX เริ่มต้นด้วยข้อมูล 128 ไบต์ ซึ่งจะทำการเก็บข้อมูลต่างๆ เพื่อนำไปใช้ในการ RESTORE ภาพ ดังรูปที่ 2.2

ข้อมูล COLOR MAP ถูกใช้ในการแทนค่าเป็น PALETTE REGISTER 16 ตัว โดยที่ PALETTE REGISTER 1 ตัว ของระบบแสดงผลอีจีเอ จะบรรจุข้อมูล 6 บิตเท่านั้น โดยแต่ละ 2 บิต จะแทนสีหลักคือ แดง, น้ำเงิน, เขียว ดังรูปที่ 2.3 ตัวอักษรใหญ่จะแทนความเข้มของสี 75%, ตัวอักษรเล็กแทน 25% แต่ละสีหลักจะมี 4 ระดับของการแสดงสีคือ 0%, 25%, 75%, 100% (ค่าบิตของตัวอักษรตัวเล็กและตัวอักษรตัวใหญ่เป็น '1' หมด) ดังรูปที่ 2.4

เช่น ค่ารีจิสเตอร์เป็น '00100001' หมายถึงแสดงสีแดงที่มีความเข้ม 75% และแสดงสีน้ำเงินที่มีความเข้ม 25% ค่ารีจิสเตอร์เป็น '00100101' หมายถึงแสดงสีแดงที่มีความเข้ม 100% แสดงสีน้ำเงินที่มีความเข้ม 25% ค่ารีจิสเตอร์เป็น '00000000' หมายถึงไม่แสดงสีใดเลย เป็นต้น

ข้อมูล COLOR MAP ในตอนต้นไฟล์ (HEADER FILE) ประกอบด้วยข้อมูล 16 ชุด ชุดละ 3 ไบต์ ไบต์ที่ 1 ของแต่ละชุดจำนวน 3 ไบต์ย่อยนั้นๆ เป็นค่าสำหรับแทนสีแดงโดยถูกนำมาสร้างเป็นเลข 4 จำนวน ตั้งแต่ 0 - 3 (00,01,10,11) และตัวเลขนี้จะต้องคูณกับ 85 เพื่อจะทำการเก็บลงไฟล์ในส่วนเฮดเดอร์ไฟล์ ในกรณีเดียวกันสีเขียวคือไบต์ที่ 2 และสีน้ำเงินคือไบต์ที่ 3 วิธีการนี้จะถูกทำซ้ำๆ กัน 16 รอบ เพื่อจะได้ค่าของสีแดง น้ำเงิน เขียว จนครบ 16 ชุด

ไบต์	ขนาด (BYTE)	ชื่อข้อมูล	รายละเอียด
0	1	Password	ค่า 'OaH' = ไฟล์ .PCX
1	1	Version	เก็บค่า Version (รุ่น) ของ PC Paintbrush ดังนี้ 0 = Version 2.5 2 = Version 2.8 with Palette 3 = Version 2.8 without Palette 5 = Version 3.0
2	1	Encoding	ค่า = 1
3	1	Bits per Pixel	จำนวนของบิตที่ใช้แสดง 1 Pixel จาก 1 Plane (ระนาบสี) 1 = EGA,VGA, or HERC 4 = CGA
4	8	Windows - Dimensions	ค่า integer 4 ค่า (ค่าละ 2 Byte) ให้ค่ามุมบน ซ้าย (x1,y1) และมุมล่างขวา (x2,y2) ของภาพ กำหนดให้รูปแบบ x1,y1,x2,y2
12	2	Horizontal - Resolution	ความละเอียดของการแสดงภาพใน แนวนอน 640 = EGA, VGA 320 = CGA 720 = HERCULES
14	2	Vertical - Resolution	ความละเอียดของการแสดงภาพใน แนวตั้ง 480 = VGA 350 = EGA 200 = CGA 348 = HERC
16	48	Color Map	ข้อมูล COLOR PALETTE
64	1	Reserved	
65	1	No. of Plane	จำนวนของ PLANES ที่ใช้แสดงภาพ
66	1	Byte per Line	จำนวนไบต์ต่อการสแกน 1 บรรทัด
68	2	Palette Info	How to interpret the palette
70	8	Maximum x value	Special for fractal files
78	8	Maximum x value	Special for fractal files
86	8	Maximum x value	Special for fractal files
94	8	Maximum x value	Special for fractal files
102	8	P Value	Special for fractal files
110	8	Q Value	Special for fractal files
118	10	Not used	

รูปที่ 2.2 โครงสร้างไฟล์ .PCX

หมายเหตุ ข้อมูลไบต์ที่ 70 - 127 ใช้เฉพาะ Fractal Programming

2.3 การแทนความหมาย PALETTE

ถ้า PALETTE REGISTER มีค่า 'XX-100001' แล้ว (ใน PALETTE ใดๆ (0-15))

บิตที่ 2 เป็น '0' และ บิตที่ 5 เป็น '1' จะได้สีแดง คือ '10' = 2

บิตที่ 1 เป็น '0' และ บิตที่ 4 เป็น '0' จะได้สีเขียว คือ '00' = 0

บิตที่ 0 เป็น '1' และ บิตที่ 3 เป็น '0' จะได้สีน้ำเงิน คือ '01' = 1

ดังนั้นค่าที่จะเก็บใน COLOR MAP ของ PALETTE ใดๆ (0 - 15) คือ (*85)

RED	GREEN	BLUE
$2*85=170$	$0*85= 0$	$1*85=85$

หรือถ้า PALETTE REGISTER มีค่า 'XX111111' แล้ว ค่าที่จะเก็บคือ

RED	GREEN	BLUE
255	255	255

จอวีจีเอมีการควบคุมสีที่แตกต่างกับอีจีเอคือ แต่ละ PALETTE REGISTER จะเก็บลำดับของ COLOR REGISTER ไว้ 1 ตัว และ COLOR REGISTER ไข่เนื้อที่ 6 บิต ดังนั้นค่าสีที่เป็นไปได้มี 64 ค่า (หมายถึง SHADES ของสี) ในวีจีเอสามารถอ่านค่า (ค่า 6 บิตของสีแดง น้ำเงิน เขียว) ข้อมูลใน PALETTE REGISTER ใน COLOR REGISTER เพื่อสร้าง COLOR MAP ขึ้นมา โดยอ่านแต่ละ PALETTE REGISTER และนำไปแอดเดรส COLOR REGISTER เพื่ออ่าน COLOR REGISTER อีกที, ทำการคูณสีแดง,น้ำเงิน,เขียว ด้วย 4 และเก็บผลลงในกลุ่ม 3 ไบต์ ของ PALETTE นั้น ในวีจีเอจะมี 256 สีที่แตกต่างกัน การใช้งานเหมือนกับการแสดง 16 สี ข้อมูลใน PALETTE จะยาวกว่า 16 สี ข้อมูลนี้จะอยู่ที่ส่วนท้ายของไฟล์ .PCX การเข้าถึงข้อมูล ต้องตรวจไบต์ตรงตำแหน่งเวอร์ชันว่าเป็น 5 (เวอร์ชัน 3.0) หรือไม่และข้อมูล PALETTE จะต้องนับถอยหลังจากท้ายไฟล์ไป 769 ไบต์ ในไบต์นั้น ต้องเป็น OCH (12 DECIMAL) แล้วข้อมูลที่ท้ายไฟล์จะเป็นข้อมูล PALETTE 256 สี

2.4 การอ่านข้อมูลเก็บลงไฟล์

ภาพที่จะอ่านจะถูกอ่านตามแนวของจอภาพ ในแนวนอนจากซ้ายไปขวาเริ่มที่ PIXEL ที่ตำแหน่งบน, ซ้าย ไปทางขวาจนสุดภาพแล้วจึงอ่านในแถวถัดไปโดยในอีจีเอและวีจีเอ ซึ่งมี หน่วยความจำหลาย ๆ PLANE ดังนั้นใน 1 แถว (SCAN LINE) จะอ่านข้อมูลในทุก PLANE เริ่มที่ PLANE 0, 1, 2 และ 3 ตามลำดับจนครบ แล้วจึงอ่านแถวถัดไปจนหมดภาพที่ต้องการเก็บ

ไบต์ที่	Palette	ไบต์ที่	Palette
16,17,18	0	40,41,42	8
19,20,21	1	43,44,45	9
22,23,24	2	46,47,48	10
25,26,27	3	49,50,51	11
28,29,30	4	52,53,54	12
31,32,33	5	55,56,57	13
34,35,36	6	58,59,60	14
37,38,39	7	61,62,63	15

รูปที่ 2.3 แสดงตำแหน่งในการเก็บสี จำนวน 3 ไบต์ คือ แดง,เขียว,น้ำเงิน

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
-	-	R	G	B	r	g	b
X	X	75%	75%	75%	25%	25%	25%

รูปที่ 2.4 ระดับของการแสดงสี

2.5 การเข้ารหัสไฟล์ .PCX

การเข้ารหัสของ ไฟล์ .PCX เรียกว่า RUN-LENGTH ENCODING (RLE) โดยมีวิธีการดังนี้ ถ้าไบต์ใดๆ มีค่าไม่เหมือนกับไบต์อื่นๆ และถ้าค่า 2 บิตบนไม่เท่ากับ“11” ไบต์นั้นจะถูกเก็บลงไฟล์ เลข นอกนั้นจะมีตัวนับ (COUNTER) อยู่คอยนับว่าเหมือนกันกี่ไบต์ แต่ต้องไม่เกิน 63 ถ้าเกินให้นำค่าตัวนับที่ได้ไป OR กับ ‘COh’ แล้วเก็บค่าตัวนับนั้นลงไฟล์ แล้วตามด้วยค่าของไบต์นั้นๆ จากนั้นจึงเริ่มนับใหม่เป็น 1 ต่อไป ถ้าในกรณีของไบต์เดียวที่มีค่าบิตบนเป็น ‘11’ ให้เขียนตัวนับ=1 (OR with COh = ‘CIh’) ลงไปในไฟล์และตามด้วยค่าไบต์นั้น ดังแสดงเป็นรูปที่ 2.5

เงื่อนไข	การกระทำ
ค่าที่เหมือนไบต์อื่น	นับ Counter + 1
ค่า < 'COh' ไม่เหมือนไบต์อื่น	เก็บลงไฟล์เลย
ค่า > 'COh' ไม่เหมือนไบต์อื่น	เก็บ 'CIh' ลงไฟล์ตามด้วยค่าไบต์นั้น
Counter > 63	Counter = 1, เก็บค่า 'FFh' และค่าไบต์ลงไฟล์

รูปที่ 2.5 การเข้ารหัสไฟล์ PCX

2.6 การถอดรหัสไฟล์ .PCX

การถอดรหัสใช้วิธีการตรงข้ามกับการเข้ารหัส พอสรุปได้ดังนี้

อ่านข้อมูลมา 1 ไบต์แล้วตรวจดูว่ามากกว่า 'COh' หรือไม่ (ค่า 2 บิตบนเป็น 11) ถ้าใช่ค่านี้จะเป็นค่าตัวนับทันที (XOR ด้วย COh จะได้ค่าจริงที่น้อยกว่า 63 ออกมา) ซึ่งไบต์ต่อไปก็จะเป็นค่าของไบต์ (VALUE) แล้วทำการขยายข้อมูลออกมาตามจำนวนตัวนับนั้นๆ แล้วนำเก็บในหน่วยความจำ ถ้าค่าไบต์ที่อ่านน้อยกว่า COh จะทำการเก็บข้อมูลไบต์นั้นตัวเดียวลงหน่วยความจำที่มีอยู่ได้เลยโดยไม่ต้องขยายข้อมูลทำเช่นนี้จนจบไฟล์ก็จะได้ข้อมูลของไฟล์ทั้งหมด

อัลกอริทึม การถอดรหัส

```

loop until EOF
read byte n
if byte n > 'COh'
    read next byte n+1
    decode byte n+1 for byte n
end.
```

ตัวอย่างไฟล์ .PCX บางส่วนดังรูปที่ 2.6 โดยแสดงเป็นเลขฐาน 16 เป็นเฮกเตอร์ 128 ไบต์ ค่าของแต่ละไบต์ส่วนใหญ่จะมากกว่า COh แล้วตามด้วยค่าที่มากกว่า COh อีกตัวหนึ่ง ซึ่งต่อกันเป็นคู่ แต่ถ้าค่าที่น้อยกว่า COh จะอยู่เดี่ยวๆ เพราะจะถูกเขียนลงไฟล์เป็นไบต์เดี่ยว ค่าที่มากกว่า COh เมื่อถูก XOR ด้วย COh แล้วจะมีค่าไม่เกิน 3Fh จากรูปที่ 2.6 เมื่อเทียบกับรูปที่ 2.2 จะเห็นว่า ไบต์แรก ที่ตำแหน่ง i1, j1 เป็น 0Ah คือ บอกว่าเป็นไฟล์ .PCX ไบต์ต่อมาคือ 05h บอกว่าเป็น PC Paintbrush Version 3.0, ไบต์ต่อมาคือ 01h บอกว่าเป็นการ encoding แบบ RLE, 01h ถัดมาบอก จำนวนบิตที่ใช้แสดง 1 pixel จาก 1 Plane ถ้าเป็น 01h ใช้กับ EGA ,VGA ,หรือ HERC แต่ถ้าเป็น 04h ใช้กับ CGA

ที่ตำแหน่ง i1, j2 4 ไบต์นี้บอกจุดเริ่มต้นของภาพ (มุมบนซ้ายสุดของภาพ) ในที่นี้เป็น 00000000h
 คือตำแหน่ง (0,0) ที่ i1, j3 เป็น 4 ไบต์บอกจุดสิ้นสุดของภาพ (มุมขวาล่างสุดของภาพ) ในที่นี้
 เป็น 7F023200h คือตำแหน่ง (32514,12800) ที่ตำแหน่ง i1, j4 บอกความละเอียดของการแสดงภาพ
 ในแนวนอน และแนวตั้ง ตามลำดับ ในที่นี้คือ 64006400h ระดับ EGA และ VGA จากนั้น 48 ไบต์
 ถัดไปจะเป็น Color map คือข้อมูล Color Palette พิจารณาที่ Palette 0 คือ 0000000h ที่ i1, j5 ไบต์
 แรกคือ 00h เก็บข้อมูลสีแดง บอกว่า แสดงสีแดง 0% ไบต์ต่อมาคือ 00h เก็บข้อมูลสีเขียว บอกว่า
 แสดงสีเขียว 0% และไบต์ที่ 3 คือ 00h เก็บข้อมูลของสีน้ำเงิน บอกว่าแสดงสีน้ำเงิน 0% ต่อมาเป็น
 Palette 1

	j1	j2	j3	j4	j5	j6
i1	0A050101	00000000	7F023200	64006400	00000000	00FF00FF
i2	0000FFFF	FF0000FF	00FFFFFF	00FFFFFF	00000000	00FF00FF
i3	0000FFFF	FF0000FF	00FFFFFF	00FFFFFF	00045000	00000000
i4	00000000	00000000	00000000	00000000	00000000	00000000
i5	00000000	00000000	00000000	00000000	00000000	00000000
i6	00000000	00000000	D2FF7FFF	FFFFFFFF	FFE0FFFF	00D100D2
i7	FF7FFFFFFF	FFFFFFFF	D0FFFF00	D100D2FF	7FFFFFFF	FFFFFFFFE0
i8	FFFF00D1	00D2FF7F	FFFFFFFF	FFFFE0FF	FF00D100	CFFFC1ED
i9	C2FF7FFF	FFFFFFFF	FFE0FFFF	00D100D2	FFC1DDCF	FF7FFFFFFF
i10	CCFFC1FD	C1D6FFFF	FFFFD2FF	FF00D100	C1FFBEC1	F76FCEFF
i11	7FFFFFFB	FFFFD0FF	B5BFCBFF	C1FDFFFF	C1FFFF00	D100C1FF
i12	C1F7A0C1	EAADBFCC	FF7FFEFF	C1EFC1E5	BFFFFFFCE	FFC1DDC1
i13	C4C1EB6F	FFFFCCFF	FF00D100	C1FF7E8D	36125BC1	EFC5FFC1
i14	F7ADC1DF	BDC2FF7F	FEFFBE94	C1C115BF	FFFFCCFF	C1FE536C
i15	949FC7FF	C1FEC1FF	C1F7FFFF	C1FFFF00	D100C1FF	...more

รูปที่ 2.6 ตัวอย่างข้อมูลในไฟล์ .PCX บางส่วน

คือ 0000FFh ไบต์แรกคือ 00h แสดงสีแดง 0% ไบต์ต่อมาคือ 00h แสดงสีเขียว 0% ไบต์ต่อมาคือ
 FFh แสดงสีน้ำเงิน 100% (ดูรายละเอียดในหัวข้อ 1.3 และ 1.4) แล้วทำซ้ำจนครบ 16 Palette จากนั้น
 พิจารณาไบต์ที่ 65 ที่ตำแหน่งไบต์ที่ 2 ของ i3, j5 จะบอกจำนวนของ Plane ที่ใช้แสดงภาพ ในที่นี้คือ

04h คือมี 4 Plane ไบต์ต่อมาเป็น 50h คือบอกว่ามีจำนวน 80 ไบต์ต่อการ scan 1 บรรทัด 2 ไบต์ ถัดมา บอกชนิดของ Palette ถ้าเป็น 1 คือ grayscale, 2 คือ color และสุดท้ายไบต์ที่ 70 ถึง 127 ไม่ได้ใช้ (ใช้เฉพาะ Fractal Programming) ส่วนที่เหลือเป็นส่วนของ Data