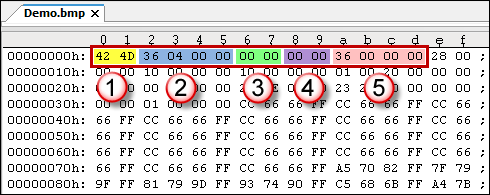
BMP格式簡介

BMP格式簡單的說就是「**File Header + Info Header + (optional palette) + Raw Data**」，不過我遇到的例子都是「**File Header + Info Header + Raw Data**」比較多，範例也將用此格式為例。

【**FILE HEADER 實例圖解**】**14 bytes**

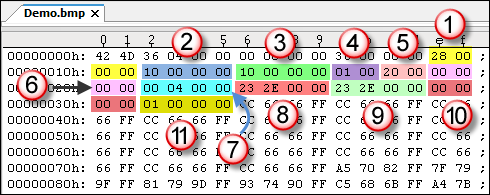
typedef struct {   
/\* type : Magic identifier,一般為BM(0x42,0x4d) \*/   
unsigned short int type;   
unsigned int size;/\* File size in bytes,全部的檔案大小 \*/   
unsigned short int reserved1, reserved2; /\* 保留欄位 \*/   
unsigned int offset;/\* Offset to image data, bytes \*/   
} FILEHEADER;



1. **type**：2 bytes，一般都是'**B**' (0x42)、'**M**' (0x4D)
2. **size**：4 bytes，記錄該BMP檔的大小，0x436 = 1078 bytes
3. **reserved1**：保留欄位，2 bytes
4. **reserved2**：保留欄位，2 bytes
5. **offset**：4 bytes，0x36 = 54 bytes

【**INFO HEADER 實例圖解**】**40 bytes**

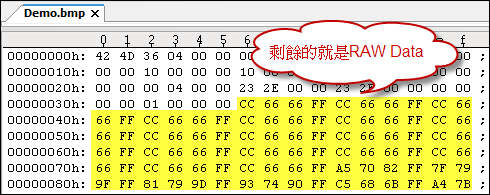
typedef struct {   
unsigned int size;/\* Info Header size in bytes \*/   
int width,height;/\* Width and height of image \*/   
unsigned short int planes;/\* Number of colour planes \*/   
unsigned short int bits; /\* Bits per pixel \*/   
unsigned int compression; /\* Compression type \*/   
unsigned int imagesize; /\* Image size in bytes \*/   
int xresolution,yresolution; /\* Pixels per meter \*/   
unsigned int ncolours; /\* Number of colours \*/   
unsigned int importantcolours; /\* Important colours \*/   
} INFOHEADER;



1. **size**：4 bytes，0x28 = 40 bytes，表示Info Header的大長度總共 40 bytes
2. **width**：4 bytes，0x10 = 16，圖檔寬度為16 pixel
3. **height**：4 bytes，0x10 = 16，圖檔高度為16 pixel
4. **planes**：2 bytes，0x01 = 1，位元圖層數為1
5. **bits**：2 bytes，0x20 = 32，每個pixel需要32bits
6. **compression**：4 bytes，0代表不壓縮
7. **imagesize**：4 bytes，0x400 = 1024 bytes，點陣圖的資料大小為1024 bytes
8. **xresolution**：4 bytes，水平解析度
9. **yresolution**：4 bytes，垂直解析度
10. **ncolours**：4 bytes，點陣圖使用的調色盤顏色數
11. **importantcolours**：4 bytes，重要的顏色數

【**RAW DATA 實例圖解**】

剛剛的File Header共14bytes，Info Header為40bytes，「imagesize」 = 1024 bytes，所以「**14 + 40 + 1024 = 1078**」，即等於File Header中「size」的大小。下圖我只擷取部分的資料，反正全部的檔案，減去Header檔54位元組，剩下的就是點陣圖的資料。



在Info Header中的「bits」為32 bits，故四個位元組一組，若24 bits，則三個位元組一組，例子中的長寬各為16，以「Z」字型來看，一列則為16組，即16 X 4 = 64 bytes。注意的是，圖中是以A、B、C ~ …的讀取順序來解說，但實際上程式所讀取到的通常會是反過來的，即… ~ C、B、A。另外，下圖是以「BGRA」排列。

