

Unidad Interdisciplinaria de Ingeniería y Tecnologías Avanzadas

Instituto Politécnico Nacional

Multimedia

Profesor: Sierra Romero Noe

Estudiante: Diaz Dominguez Luis Angel

2023640230

1. 1. Introducción

En este reporte se analiza la composición binaria del archivo **example001.bmp**.

2. Objetivo

El objetivo es identificar la forma en que se almacenan los metadatos y los valores cromáticos en un archivo de mapa de bits (DIB) de **24 bits**, utilizando herramientas de edición hexadecimal para contrastar la teoría con los datos reales del archivo.

3. Especificaciones del archivo (Header)

El encabezado inicial del archivo, conocido como **BITMAPFILEHEADER**, consta de **14 bytes** que definen la identidad del archivo.

4. Firma del archivo (Signature):

En el *offset 0000h* se encuentran los bytes **42 4D**, los cuales corresponden a los caracteres ASCII '**BM**'. Esto confirma que el sistema operativo reconoce correctamente el archivo como un **Bitmap**.



00000000	42	4D	36	03	00	00	00	00	00	00	36	00	00	00	28	00	BM6.
----------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	------

Tamaño del archivo:

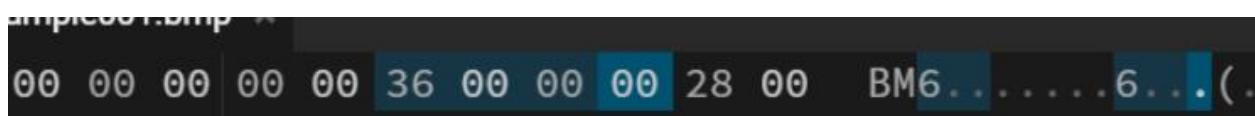
En el *offset 0002h* se encuentra el valor hexadecimal **36 03 00 00**. Al interpretarse en formato **Little-endian** (del byte menos significativo al más significativo), se obtiene **00 00 03 36**, lo cual equivale a **822 bytes** en representación decimal.



Sin título	example001.bmp	example001.bmp	00000000	42	4D	36	03	00	00	00	00	00	36	00	00	00	28	00	BM6.....6...()
------------	----------------	----------------	----------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----------------

Punto de inicio de los datos (DataOffset):

En el *offset 000Ah* se localiza el valor hexadecimal **36 00 00 00**. Interpretado en formato **Little-endian**, este valor corresponde a **0x36**, lo que indica que los metadatos finalizan y que los datos reales de la imagen comienzan a partir del **byte 54**.



example001.bmp	00	00	00	00	00	36	00	00	00	28	00	BM6.....6...()
----------------	----	----	----	----	----	----	----	----	----	----	----	----------------

3. Especificaciones de la imagen (InfoHeader)

El bloque de información, conocido como **BITMAPINFOHEADER**, consta de **40 bytes** y define las propiedades físicas de la matriz de píxeles.

- **Dimensiones:**

En los *offsets 0012h* (ancho) y **0016h** (alto) se encuentra el valor hexadecimal **10 00 00 00**. Al interpretarse en formato **Little-endian**, el valor **0x10** equivale a **16** en decimal, lo que confirma que la imagen tiene una resolución de **16 × 16 píxeles**.

Información de archivo			-Sin título-	example001.bmp
Nombre de archivo	example001.bmp		00000000	42 4D 36 03 00 00 00
Tipo	Sin signo (+)	Con signo (±)		
Entero de 8 bits	16	16	00000010	00 00 10 00 00 00 00
Entero de 16 bits	16	16	00000020	00 00 00 03 00 00 00
Entero de 24 bits	16	16	00000030	00 00 00 00 00 00 00
Entero de 32 bits	16	16	00000040	FF FF 00 FF 00 00 00
Entero de 64 bits (+)	68719476752		00000050	FF 00 00 FF 00 00 00
			00000060	FF FF FF FF FF FF 00
			00000070	FF FF 00 FF 00 00 00
			00000080	FF 00 00 FF 00 00 00
			00000090	FF FF FF FF FF FF 00
			000000A0	FF FF 00 FF 00 00 00
			000000B0	FF 00 00 FF 00 00 00
			000000C0	FF FF FF FF FF FF 00

Profundidad de color:

En el **offset 001Ch** se observa el valor hexadecimal **18 00**. Interpretado en formato **Little-endian**, este valor corresponde a **24 bits por píxel**, lo que implica que cada píxel utiliza **3 bytes** para representar los canales de color (RGB).

Información de archivo			-Sin título-	example001.bmp
Nombre de archivo	example001.bmp		00000000	42 4D 36 03 00 00 00
Tipo	Sin signo (+)	Con signo (±)		
Entero de 8 bits	24	24	00000010	00 00 18 00 00 00 00
Entero de 16 bits	24	24	00000020	00 00 00 03 00 00 00
Entero de 24 bits	24	24	00000030	00 00 00 00 00 00 FF
Entero de 32 bits	24	24	00000040	FF FF 00 FF 00 00 FF
Entero de 64 bits (+)	216172782113783832		00000050	FF 00 00 FF 00 00 FF
Entero de 64 bits (±)	216172782113783832		00000060	FF FF FF FF FF FF FF
Float, P. 16 bits	0.000001430511		00000070	FF 00 00 FF 00 00 FF
Float, P. 32 bits	3.3631153e-44		00000080	FF 00 00 FF 00 00 FF
Float, P. 64 bits	3.1315120625140267e-204		00000090	FF 00 00 FF 00 00 FF
			000000A0	FF 00 00 FF 00 00 FF
			000000B0	FF 00 00 FF 00 00 FF
			000000C0	FF FF FF FF FF FF FF
			000000D0	FF FF 00 FF 00 00 FF
			000000E0	FF 00 00 FF 00 00 FF
			000000F0	FF FF FF FF FF FF FF
			00000100	FF 00 FF 00 00 FF 00

4. Análisis de la carga útil (Pixel Data)

En un archivo BMP, el barrido de los datos de imagen se realiza de forma inversa: los píxeles se almacenan de **izquierda a derecha**, pero comenzando desde la **parte inferior hacia la superior** de la imagen. Además, los canales de color se guardan en el orden **BGR** (Azul-Verde-Rojo).

Validación de colores en el editor hexadecimal:

Al inspeccionar los datos a partir del **byte 54 (0x36)**, se identifican secuencias constantes que corresponden a colores específicos.

- **Blanco:**

Representado por la secuencia hexadecimal **FF FF FF**, lo que indica que los tres canales de color se encuentran en su valor máximo.

Rojo:

Representado por la secuencia hexadecimal **00 00 FF**, donde únicamente el **tercer byte** se encuentra activo, correspondiente al canal **rojo**, de acuerdo con el orden de almacenamiento **BGR**.

00 00 FF 00

Verde:

Representado por la secuencia hexadecimal **00 FF 00**, donde el **canal central** (verde) se encuentra activo, conforme al orden de almacenamiento **BGR**.

Azul:

Representado por la secuencia hexadecimal **FF 00 00**, donde el **primer byte** corresponde al canal **azul**, de acuerdo con el orden de almacenamiento **BGR**.

00 FF 00 00 FF 00 00
00 FF FF FF FF FF FF

Negro:

Representado por la secuencia hexadecimal **00 00 00**, lo que indica la **ausencia de información cromática** en los tres canales de color.

5. Cálculos de comprobación

Para verificar la consistencia del análisis, se realiza la siguiente validación matemática, originalmente planteada en clase:

1. **Píxeles totales:**

$$16 \times 16 = 256 \text{ píxeles.}$$

2. **Bytes de la imagen (Payload):**

$$256 \text{ píxeles} \times 3 \text{ bytes/píxel} = 768 \text{ bytes.}$$

3. **Tamaño total del archivo:**

$$768 \text{ bytes (datos)} + 54 \text{ bytes (metadatos)} = 822 \text{ bytes.}$$

Dado que la longitud de cada fila es de **48 bytes**, valor que es múltiplo de **4**, el archivo no requiere **bytes de relleno (padding)**.

6. Conclusión

El análisis realizado permitió constatar que la representación visual del archivo **example001.bmp** es una traducción directa de una matriz de valores hexadecimales. Asimismo, se validó que el tamaño del archivo, la resolución y la codificación cromática cumplen correctamente con los estándares técnicos del formato **BMP**, de acuerdo con la documentación y los conceptos teóricos analizados.

Si quieras, puedo:

- darle un tono todavía más **académico**,
- ajustarlo al formato que te pidan en la escuela,
- o revisar todo el documento completo por coherencia y estilo 