# **Отчет по 2 лабораторной работе «Чтение и запись графических изображений в формате BMP»**

**Графическим форматом** называют порядок (структуру), согласно которому данные, описывающие изображение, записаны в файле. Типы форматов графических файлов определяются способом хранения и типом графических данных. Наиболее широко используются **растровый** (битовый), **векторный** и **метафайловый** форматы.

**Векторный формат** наиболее удобен для хранения изображений, которые можно разложить на простые геометрические фигуры (например, чертежи или текст). Векторные файлы содержат математические описания элементов изображения. Наиболее распространенные векторные форматы: AutoCADDXF и MicrosoftSYLK.

**Растровый формат** используется для хранения растровых данных. Файлы такого типа особенно хорошо подходят для хранения изображений реального мира, например, оцифрованных фотографий. Растровые файлы содержат битовую карту изображения и ее спецификацию. Наиболее распространенные растровые форматы: BMP, TIFF, GIF, PCX, JPEG.

**Битовые изображения**, как правило, выводятся на экран быстрее, как их внутренняя структура аналогична (до некоторой степени) структуре видеопамяти. Изображения, получаемые при помощи сканеров и цифровых видеокамер, получаются именно как битовые изображения.

К недостаткам битовых изображений можно отнести большой объем памяти, требующийся для их хранения (около 1 Мбайт в режиме True Color), невозможность масштабирования без потери качества изображения, а также сложность выделения и изменения отдельных объектов изображения.

**Метафайловый формат** позволяет хранить в одном файле и векторные, и растровые данные. Примером такого формата являются файлы CorelDRAW–CDR.

# **Формат BMP (Bitmap Picture):**

**BMP** — аппаратно-независимое побитовое изображение Windows - поддерживается любыми Windows-совместимыми программами. Структура файла BMP используется Windows для хранения растровых изображений. Например, в этом формате хранятся рисунки фона, пиктограммы и другие растровые изображения Windows. Формат сводит к минимуму вероятность ошибок или неправильной интерпретации растровых данных.

Формат .bmp представляет из себя несжатое (в основном) изображение, которое довольно легко читается и выводится в ОС Windows, в которой есть специальные функции, которые в этом помогают.

В операционной системе Windows используются два формата растровых (битовых) изображений –аппаратно-зависимый **DDB** (device dependent bitmap) и аппаратно-независимый **DIB** (device-independent bitmap).

Согласно определению, данному в документации к SDK, битовое изображение DDB есть набор бит в оперативной памяти, который может быть отображен на устройстве вывода (например, выведен на экран видеомонитора или распечатан на принтере). Внутренняя структура изображения DDB жестко привязана к аппаратным особенностям устройства вывода. Поэтому представление изображения DDB в оперативной памяти полностью зависит от устройства вывода.

Для описания растровых изображений в Windows используется следующая структура формата .bmp:

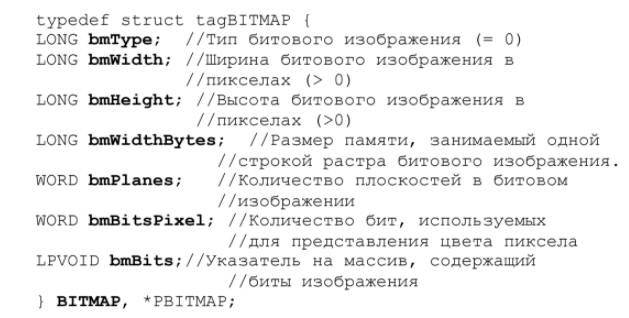


Рис.1 Структура bitmap

Каждый файл BMP содержит заголовок файла, заголовок изображения, растровые данные и карту цветов:

**− Заголовок файла** ВМР содержит информацию о типе и размере файла, а также о расположении в нем данных. После заголовка файла следует структура, задающая информацию о размере, цвете и сжатии изображения.

**− Сжатие.** Поле biCompression в BITMAPINFO определяет, какой тип сжатия используется в файле. Если значение поля равно BI\_RGB, то рисунок не сжимается. Если же значение поля равно BI\_RLE4 или BI\_RLE8, то рисунок использует метод сжатия групповым кодированием для изображений, соответственно, с 4-бит/пиксель или 8 бит/пиксель.

**− Карта цветов.** Значение поля biCirUsed определяет, будут ли использоваться растровым рисунком индексированные цвета. Если рисунок состоит из 1-, 4-, 8-битовых пикселей, то должна использоваться карта цветов. Если значение поля biCirUsed равно нулю, то применяется либо полная карта, либо 24-битовый цвет. Если поле не равно нулю, то его значение равно действительному количеству цветов, которые будут доступны программе-генератору графики или драйверам устройств. Поле biClrimportant определяет, сколько цветов в индексированной карте считаются важными для изображения рисунка. Если его значение равно нулю, то важными будут все цвета.

**− Растровые данные.** За картой цветов следуют растровые данные. Они хранятся в виде 1-, 4- , или 8-битовых индексов карты цветов или в виде буквенных 24-битовых данных системы RGB. Пиксели в растре записываются, начиная с нижнего левого угла изображения, и читаются слева направо и снизу-вверх.

Данный формат невозможно сжать, за исключением версии формата с 4- и 8-битовым цветом. Из-за этого формат .bmp не возможен для работы в Сети. Лучше всего переводить формат .bmp в другие форматы.

Формат bmp-файлов представлен на рис. 1.2:

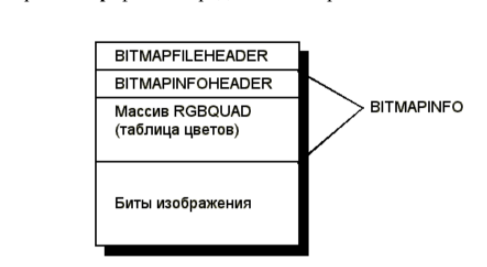


Рис.1.2 Структура файлов .bmp

Файл, содержащий битовое изображение, начинается со структуры **BITMAPFILEHEADER.** Эта структура описывает тип файла и его размер, а также смещение области битов изображения. Сразу после структуры BITMAPFILEHEADER в файле следует структура **BITMAPINFO**, которая содержит описание изображения и таблицу цветов.

Описание изображения (размеры изображения, метод компрессии, размер таблицы цветов и т. д.) находится в структуре **BITMAPINFOHEADER.** В некоторых случаях (не всегда) в файле может присутствовать палитра–таблица цветов (как массив структур RGBQUAD), присутствующих в изображении. Палитра отсутствует, если число бит на пиксел равно 24. Также палитра не нужна и для некоторых цветовых форматов 16 и 32 бит на пиксел.

После палитры (если она есть) в файле BMP записывается растр в виде битового (а точнее, байтового массива). В битовом массиве последовательно записываются байты строк растра. Количество байт в строке должно быть кратно четырем, поэтому если количество пикселов по горизонтали не соответствует такому условию, то справа в каждую строку дописывается некоторое число битов (выравнивание строк на границу двойного слова). Точное значение смещения битов изображения находится в структуре **BITMAPFILEHEADER**.