                     FORMAT FOR BITMAP FILES
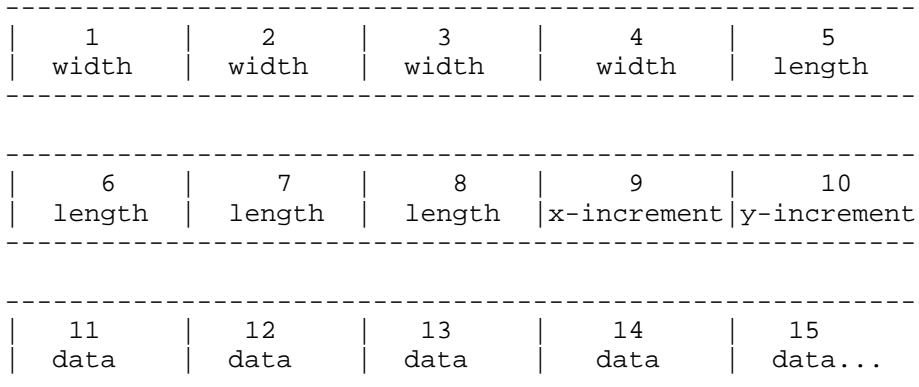
This note describes a proposed format for storing simple bitmaps (one
bit per pixel) (one bit per pixel) These files may be very large and the
intent is to use this format for short term storage and passing data
between closely coupled programs.   The data in the file should be
stored in 8-bit bytes (octets). Bitmaps may be any size.

The first 4 octets of the file gives the width of each line (x
direction), and the next 4 octets should give the number of lines of
the display (length, y direction). After this is one octet for the x
increment and one octet for the y increment.   Following these 10
octets is the bitmap itself.   The length and width fields are
stored most significant octet first.

The x and y increment octets tell how much space is between pixels.
For an ordinary display, both these would be one.

Each line of the display should be scanned from left to right. The
lines should start at the top and work down. Each line in the bitmap
should end on an octet boundary. If the width of the display is not
divisable by 8, the rest of the last octet should be filled with
zeros on the right.

Below is a representation of a bitmap file (each square is one
octet):

        | 1 | 2 | 3 | 4 | 5 |
        | width | width | width | width | length |

        | 6 | 7 | 8 | 9 | 10 |
        | length | length | length |x-increment|y-increment|

        | 11 | 12 | 13 | 14 | 15
        | data | data | data | data | data...

For example, bitmaps from the RAPICOM 450 can be in Fine Detail,
Quality, or Express Mode.   In Fine Detail mode the x-increment and
y-increment would be 1, for Quality mode, the x-increment would be 1
and the y-increment would be 2, and for Express mode, the x-increment

would be 1 and the y-increment would be 3. For these bitmaps it is
intended that each scan line be repeated y-increment times when they
are displayed.

Alan R. Katz