2B-4

グラフィック・ディスプレイIBM 3179-Gにおける Area Fill のアルゴリズムについて。

森徹，吉川博文
日本アイ・ビー・エム株式会社 大和研究所 標準端末開発

"Area" is one of output primitives to draw graphic picture. It's relatively difficult to implement it on a low-cost display terminal because it requires some hardware assist to achieve reasonable speed. IBM 3179-G Graphic Color Display Station supports the area primitive as well as other primitives such as "Line" and "Marker". We developed a special method to draw area very fast on IBM 3179-G.

An area which is used on IBM 3179-G is defined as follows :

- One or more closed figures jointly define the area boundary.
- Any connected region with an odd number of line crossings from infinity is shaded with a shading pattern.
- The boundary lines defining each closed figure can optionally be drawn.
- If two areas are defined that are adjacent to each other, the shading patterns used in the two areas should not overlap at the boundary, and neither should there be any gaps between the patterns.

To implement the drawing of area properly on IBM 3179-G, we developed a special method and implemented it on hardware logic with microcode assistance to achieve high-speed drawing. 3179-G has another mono-plane APA buffer, called Work-APA buffer, for the sole purpose of filling area in addition to essential tri-plane APA buffer, called RGB-APA buffer, from which graphic picture is refreshed to screen.

The procedure of the area fill is summarized as follows putting the focus on the function split between hardware and microcode.

1. IBM 3179-G microcode defines the boundary of area on Work-APA buffer with hardware assistance. If the boundary of area is also to drawn — we call such an area, 'Area with Boundary' —, the microcode draws the boundary of area also on RGB-APA buffer.
2. IBM 3179-G hardware, by scanning the Work-APA buffer from left to right, starts filling area from the point it crosses the boundary odd times and ends filling area at the point it crosses the boundary even times. Now the area on Work-APA buffer is solid-filled.
3. For Area with Boundary, the microcode, assisted by hardware, erases the boundary lines of solid-filled area on Work-APA buffer. This is to avoid the overlap of the boundary line and the interior of area at stage 5.
4. The microcode sets shading pattern, color and mix attributes to the hardware. The mix attribute is used to specify how the area is mixed

---

Implementation of Area-Fill on IBM 3179-G
Thoru MORI, Hirofumi YOSHIKAWA
IBM, Japan

with the figures already drawn on RGB-APA buffer, i.e. OR'ed, overpainted, exclusive OR'ed, etc..

5. The hardware transfers the solid-filled area on Work-APA buffer to the RGB-APA buffer using those attributes.

It's rather complecated to define the boundary of area on Work-APA buffer at stage 1. Simple odd-even method doesn't work at top vertex where two lines meet so that there is only one pel in the scan line. It cannot be used also for the horizontal line and the diagonal line of slight slope where there are multiple pels on the same scan line. We have to find some rules which eliminates such problems. Besides, if another area having at least one boundary line in common with this area is drawn, the overlap of shading will occur at the common boundary. To allow neither overlap nor gap between the two adjacent areas, another rules should be established to define how the boundary of area is included for all areas drawn on IBM3179-G.

Thorough much study and experiment, we've concluded that the following five rules are necessary and sufficient when the boundary of area is drawn on Work-APA at stage 1.

1. Draw in exclusive OR.
2. Don't draw horizontal line.
3. Don't draw start pel.
4. Draw line upward.
5. For the line of slight slope which results in multiple pels in the same scan line, draw only the pel nearest to the start point per scan line.

The functions to realize this area-draw method is split between hardware and microcode so that the functions performed by hardware are really suitable to be done by hardware. For example, To get the Area shaded on RGB-APA buffer with shading pattern specified at stage 4, it's only AND'ing the shading pattern with the solid-filled area made on Work-APA buffer at stage 2. For anothe example, the filling area by even-odd method at stage 2 is achieved just by replacing each pel with the result of exclusive-OR ing the pel with its left pel from left to right on each scan line on Work-APA buffer. Drawing thousands of pels would take unbearable time if it had been carried out only by microcode, but IBM3179-G can shade any area only in 10-200 milliseconds by the assist of these hardware.