

Supplemental Information Regarding KeyLayoutConverter

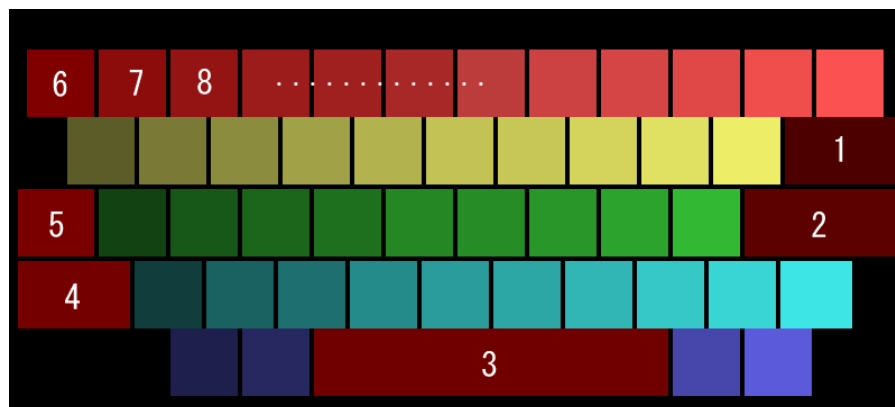
Obtain the kbd file that stores the following array by loading the tga file containing the keyboard layout into KeyLayoutConverter.

```
struct
{
    u8  x;                // Button's upper left x-coordinate
    u8  y;                // Button's upper left y-coordinate
    u8  width;            // Button width
    u8  height;           // Button height
    s16 group_offset;     // Button group information
    u16 id;               // Scan code
};
```

(The storage order for the array is insignificant.) Since the kbd file contains no header information, the number of elements in the array can be computed from the file size.

The x, y, width, and height elements mean exactly what would be expected.

The id element is the scan code for a PC keyboard. Search the HSB color coordinate systems of non-black rectangles in the tga file, and assign numbers in order of increasing value starting from the rectangle with the smallest value. (See the following diagram.)



group_offset is used to better deal with the keys that are not rectangular (such as the Japanese Enter key), while KeyLayoutConverter finds rectangles of the same color and assigns them a value. When group_offset is zero, there are no groups. When group_offset is a non-zero value, elements of arrays that are offset by group_offset units are in the same group and are assigned the same id. Based on this information, coordinates are obtained by looking through a group, and processes, such as showing the keyboard image as being pressed, are performed.

The kbdmap file exists to convert from scan codes to UTF-16. The kbdmap file is a text file in UTF-16LE-BOM format to convert scan codes to character codes. Scan code 0 has been assigned as the state when a button is not being pressed, so scan codes start from 1. The first character in the file corresponds to scan code 1, followed by scan codes 2, 3, 4, etc. Conversion from a scan code to a character code uses the following formula.

```
character_code = content_of_kbdmap[scancode];
```

Since the scan codes start at 1, it may seem that the array index should be `scancode-1`. However, a 2-byte BOM is at the beginning of the `kbdmap` file, which results in that portion being skipped, so everything works out correctly. The data density loses 2 bytes, but this specification is used because files with BOM are easily handled in a text editor.

Special keys such as Backspace or Enter are placed in regions with smaller scan codes and the number of these special keys is written to the source code. If the scan code value is smaller than the number of special keys, a special process is performed to convert the scan code to a character code. If the scan code value is not smaller, the number of special keys is subtracted from the scan code, and the conversion is performed on the result.

Be aware that the color sorting process is imperfect, and sometimes different colors are handled as if they were identical. (This problem may be caused by errors during the KeyLayoutConverter conversion from RGB to HSB color space.) This problem can be dealt with by carefully selecting the colors to be used.

June 30, 2005