

DirectFB 1.0.0-rc1 (2006-10-1) Quick Review

— Renesas の提案がメインラインされた日本語対応拡張 —

Hisao Munakata ([munakata.hisao\(AT\)renesas.com](mailto:munakata.hisao(AT)renesas.com))
株式会社 ルネサスソリューションズ

Open Source 開発コミュニティとの連携モデル提案

■ Step_1 : Open Source で公開されたコードベースを利用する

- CPU アーキテクチャポータビリティ
- ハードウェアアクセラレータ用のカスタムドライバーの開発
- ハードウェアの構造に合わせた実装の最適化の検討(バッファ構造など)

■ Step_2 : 製品要求仕様に合わせてコードを拡張する

- オープンソースの仕様が自分の実装ターゲットにフィットしているか調査する
- 必要に応じてオープンソースの仕様を拡張(改造)する[必要がある]

☆ 一般性のある課題についてはマスターコードに戻した方が皆が Happy では？

■ Step_3 : 拡張したコードを開発プロジェクトに還元(メインライニング)する

- コミュニティのメンテナーと“会話”する(出来れば一度は会えると後が早い)
- 本体機能を壊さない拡張仕様を提案していけばメインライニングの可能性がある

ルネサスからの提案の趣旨

— DirectFB を 日本の地デジ (ISDB-T) に使えるようにしたい —

■ ARIB に規定された日本語フォント 等への対応

- ① AYUV 色空間のサポート ← オリジナルは ARGB サポートのみ
- ② 2bpp (4階調) フォント対応 ← オリジナルは 1bpp or 8bpp サポートのみ
- ③ EUC/SJIS エンコーディング対応 ← オリジナルは UTF-8 サポートのみ
- ④ ARIB Font 指定への対応
- ⑤ 領域指定クリップ (スイッチングプレーンのハードウェア実装)

■ 低価格ソリューション (= 省メモリー) 対応

- ⑥ フォントキャッシュサイズ制限の追加
- ⑦ シングルバッファ、複数プレーンの同期フリップ機構 (Group Flip)
- ⑧ 合成シナリオカスタマイズ (未使用の合成用バッファをスキップする機能)

■ 組み込みCPU 対応

- ⑨ (カラーキー参照時の) 32bit バウンダリーの自動アライメント調整機能

<http://www.directfb.org/index.php?path=Main%2FNews> CELF Technical Jamboree #11

API [1.0.0-rc1]

- Added '2' bit packed alpha pixel format DSPF_A4, e.g. for fonts.

☆ ARIB4階調フォント対応 (②)

API | IDirectFBDisplayLayer [1.0.0-rc1]

- Added **SetClipRegions()** that, if supported by hardware, sets the clipping regions that are used to enable or disable visibility of parts of the layer. The number of regions is stated in the display layer description. The layer will be shown only in these regions or except in these regions.

☆ ARIB 動静切り替えプレーン対応 (⑤)

API | IDirectFBFont [1.0.0-rc1]

- Added **EnumEncodings()** enumerating all provided encodings, also see Fonts section below.
- Added **FindEncoding()** to look up an encoding directly by its name.
- Added **SetEncoding()** for choosing the encoding for local interface methods and as a default for the surfaces.

☆ ARIB文字コード対応 (④)

API | IDirectFBSurface [1.0.0-rc1]

- Added **SetEncoding()** to choose an encoding for the text routines, will be overwritten by **SetFont()** which takes the default encoding of the new font, also see Fonts section below.

☆ ARIB文字コード対応 (④)

Fonts [1.0.0-rc1]

- Added support for **other encodings than UTF8**. Encodings are (or can be) provided with each font implementation. This model reduces code sharing slightly, but allows higher efficiency via optimized combination of decoding and translation.
- Every encoding just has a name and an ID, where the **name can be chosen freely**, except for DTEID_UTF8 which is always available and has the name "UTF8".
- **Implemented** UTF8 and "Latin1" encoding in FT2 **font loader**. Nice demonstration how different encodings can be used, while still having a single glyph cache, which is no longer based on character codes, but on their raw indices.

☆ ARIB文字コード対応 (③④)

CELF Technical Jamboree #11

Graphics Core [1.0.0-rc1]

- Implemented **font cache limit**, currently hardcoded to five rows, each row stores a number of glyphs. When the maximum number is reached it will reuse **the least recently used row**, kicking out all glyphs that have been on it.

☆ 省メモリ対策 (⑥)

(日本語のような文字種が膨大なフォントに対する対策)

Software Renderer [1.0.0-rc1]

- Fix **unaligned 32 bit accesses** in color keying code.

☆ 組み込み系 CPU (SuperH, ARM等) に対する最適化 (⑨)

API [0.9.25]

- Added DSPF_AYUV, a 32bit packed AYUV format for graphics, being the counterpart of ARGB in the YUV color space.

☆ ISDB-T 放送仕様 への対応 (①)

Graphics Core [0.9.25]

- Fix negative line width handling in fill_tri()

☆ バグを見つけてフィードバックした

提案中の機能

CLUT to CLUT テーブル変換機能 (SetIndexTranslation)

So the actual application could look like this:

```
int indices[4] = { -1, 20, 21, 22 };

surface->SetBlittingFlags( surface,
DSBLIT_INDEX_TRANSLATION );
surface->SetIndexTranslation( surface, &indices, 4 );

surface->SetFont( surface, font2bpp );
surface->DrawString( surface, ... );
```

- ☆ 階調色が指定されている（自動計算できない）ARIBフォントカラー指定への対応
指定された4階調色に対応した 本体のカラーテーブルを参照するための API 拡張

gfxdriver 内での実装最適化 — API には反映されていない内容 —

- Grouped Flip:
(シングルバッファ設定におけるプレーン間 同期Flip (⑦))

☆ ARIBプレーン合成における省メモリ対策

- プレーン使用状況に応じた合成処理の最適化 (⑧)

☆ 省バス幅 & 省メモリ対策