

---

# SH7262/SH7264 グループ

R01AN0613JJ0103

Rev. 1.03

## ビデオディスプレイコントローラ 3 $\alpha$ ブレンド窓機能の使用例

---

2011.03.23

### 要旨

本アプリケーションノートは、SH7262/SH7264 のビデオディスプレイコントローラ 3 (VDC3) の  $\alpha$  ブレンド窓機能について説明します。

### 動作確認デバイス

SH7264

### 目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラム例.....	25
4. 参考ドキュメント.....	42

## 1. はじめに

### 1.1 仕様

ビデオディスプレイコントローラ 3 (VDC3) の  $\alpha$  ブレンド窓機能を使用して、入力映像とグラフィックス画像 (2 面) を重ね合わせて TFT-LCD パネルに表示します。

### 1.2 使用機能

- ビデオディスプレイコントローラ 3 (VDC3)
- 汎用入出力ポート

### 1.3 適用条件

マイコン	SH7262/SH7264
動作周波数	内部クロック : 144 MHz バスクロック : 72 MHz 周辺クロック : 36 MHz
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded Workshop Ver.4.07.00
C コンパイラ	ルネサスエレクトロニクス製 SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.03 Release00
コンパイルオプション	High-performance Embedded Workshop でのデフォルト設定 (-cpu=sh2afpu -fpu=single -object="\$(CONFIGDIR)¥\$(FILELEAF).obj" -debug -gbr=auto -chgincpath -errorpath -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1 -nologo)

### 1.4 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。合わせて参照してください。

- SH7262/SH7264 グループ 初期設定例
- SH7262/SH7264 グループ ビデオディスプレイコントローラ 3 TFT-LCD パネル表示例
- SH7262/SH7264 グループ ビデオディスプレイコントローラ 3 映像表示例
- SH7262/SH7264 グループ ビデオディスプレイコントローラ 3 映像録画例

### 1.5 "L"アクティブ端子 (信号) の表記について

端子名(信号名)末尾の # は "L" アクティブ端子(信号)であることを示します。

## 2. 応用例の説明

本応用例は、VDC3 の  $\alpha$  ブレンド窓機能を使用するための端子接続例と設定例を説明します。

### 2.1 VDC3 の動作概要

本アプリケーションノートでは、VDC3 の概要と  $\alpha$  ブレンド窓機能について説明します。

#### 2.1.1 概要

VDC3 には以下の 4 つの機能があります。なお、映像表示機能と映像録画機能は同時に使用することができません。

1. 映像表示機能：入力映像を縮小後、メモリにバッファし、パネル表示する機能
2. 映像録画機能：入力映像を大容量内蔵 RAM または SDRAM に設定フィールド数分を保存する機能
3. 入力映像にグラフィックス画像（2 面）を重ね合わせる機能
4. TFT-LCD パネル向けの制御信号出力機能

## 2.1.2 特長

表 1にVDC3 の特長を示します。

表1 VDC3 の特長

項目	機能
動作周波数	映像入力クロック : 27MHz パネルクロック : 4~36MHz (パネル仕様に依存)
入力映像規格	ITU-R BT.656 規格準拠 8bit 入力 (27MHz) ITU-R BT.601 規格準拠 8bit シリアル入力に対応 (27MHz)
映像録画機能	RGB565 形式にて 1/2 フィールドのレート (NTSC : 30fps、PAL : 25fps) で映像を保存
映像画質調整機能	コントラスト調整、ブライト調整
映像スケーリング処理	垂直 1/2、1/3、1/4 倍 水平 2/3、1/2、1/3、1/4 倍 PAL 対応のため、さらに、各スケール値を 6/7 倍にすることが可能
グラフィックス画像	2面 (レイヤ1、レイヤ2) RGB565 プログレッシブ形式 ( $\alpha$ : なし、R : 5bit、G : 6bit、B : 5bit の計 16bit) $\alpha$ RGB4444 プログレッシブ形式 ( $\alpha$ : 4bit、R : 4bit、G : 4bit、B : 4bit の計 16bit)
グラフィックス機能	<ul style="list-style-type: none"> <li>• <math>\alpha</math> ブレンド窓機能 : 設定した領域にて、入力映像、レイヤ1、レイヤ2 を透過率 <math>\alpha</math> に基づき Mixing (フェードイン、フェードアウト機能あり)</li> <li>• クロマキー機能 : 設定した RGB 色にて透過率 <math>\alpha</math> に基づき Mixing</li> <li>• ドット <math>\alpha</math> 機能 : <math>\alpha</math> RGB4444 形式のグラフィックス画像のとき透過率 <math>\alpha</math> に基づき Mixing</li> <li>• 上記機能の <math>\alpha</math> 値の優先順位は <math>\alpha</math> ブレンド窓 &gt; クロマキー &gt; ドット <math>\alpha</math></li> </ul>
出力映像サイズ	640pixel × 480line (VGA) サイズ 480pixel × 240line (WQVGA) サイズ 320pixel × 240line (QVGA) サイズ横長 240pixel × 320line (QVGA) サイズ縦長  【注】 入力映像の最大表示可能領域は、NTSC : 480pixel×240line、PAL : 480pixel×288line です。
出力映像形式	RGB565 プログレッシブ映像出力 (16bit のパラレル出力)
同期信号出力	TFT-LCD パネル向けの制御信号を出力します
割り込み出力	ライン割り込み出力 (任意のラインにて出力可能) BT.601、BT.656 映像入力の VSYNC 周期乱れ検出信号 フィールド書き込み完了信号 内部バッファのオーバ/アンダフロー検出信号

## 2.1.3 入出力端子

表 2にVDC3 の入出力端子を示します。

表2 VDC3 の入出力端子

端子名	入出力	名称	説明
DV_CLK	入力	映像入力クロック	BT.601、BT.656 のクロックを入力します。
DV_VSYNC	入力	VSYNC 入力	BT.601 の VSYNC 信号を入力します。
DV_HSYNC	入力	HSYNC 入力	BT.601 の HSYNC 信号を入力します。
DV_DATA7~0	入力	BT.601、BT.656 入力	BT.601、BT.656 のデータ信号を入力します。
LCD_CLK	出力	パネルクロック	パネルクロック出力端子です。
LCD_EXTCLK	入力	パネルクロックソース	パネルクロックソース入力端子です。
LCD_VSYNC	出力	パネル用 VSYNC 出力	パネル用垂直同期信号出力端子です。
LCD_HSYNC	出力	パネル用 HSYNC 出力	パネル用水平同期信号出力端子です。
LCD_DE	出力	パネル用データイネーブル出力	パネル用データイネーブル信号またはデータ開始位置パルス信号の出力端子です。
LCD_DATA15~0	出力	パネル用データ出力	パネル用データの出力端子です。 <small>MSB LSB          MSB LSB</small> [ 15 : 11 ] : Red[ 4 : 0 ] [ 10 : 5 ] : Green[ 5 : 0 ] [ 4 : 0 ] : Blue[ 4 : 0 ]
LCD_M_DISP	出力	パネル用制御信号	パネル用交流化信号

## 2.1.4 構成

図 1にVDC3の全体ブロック図を示します。各機能ブロックの概要は表 3を参照してください。

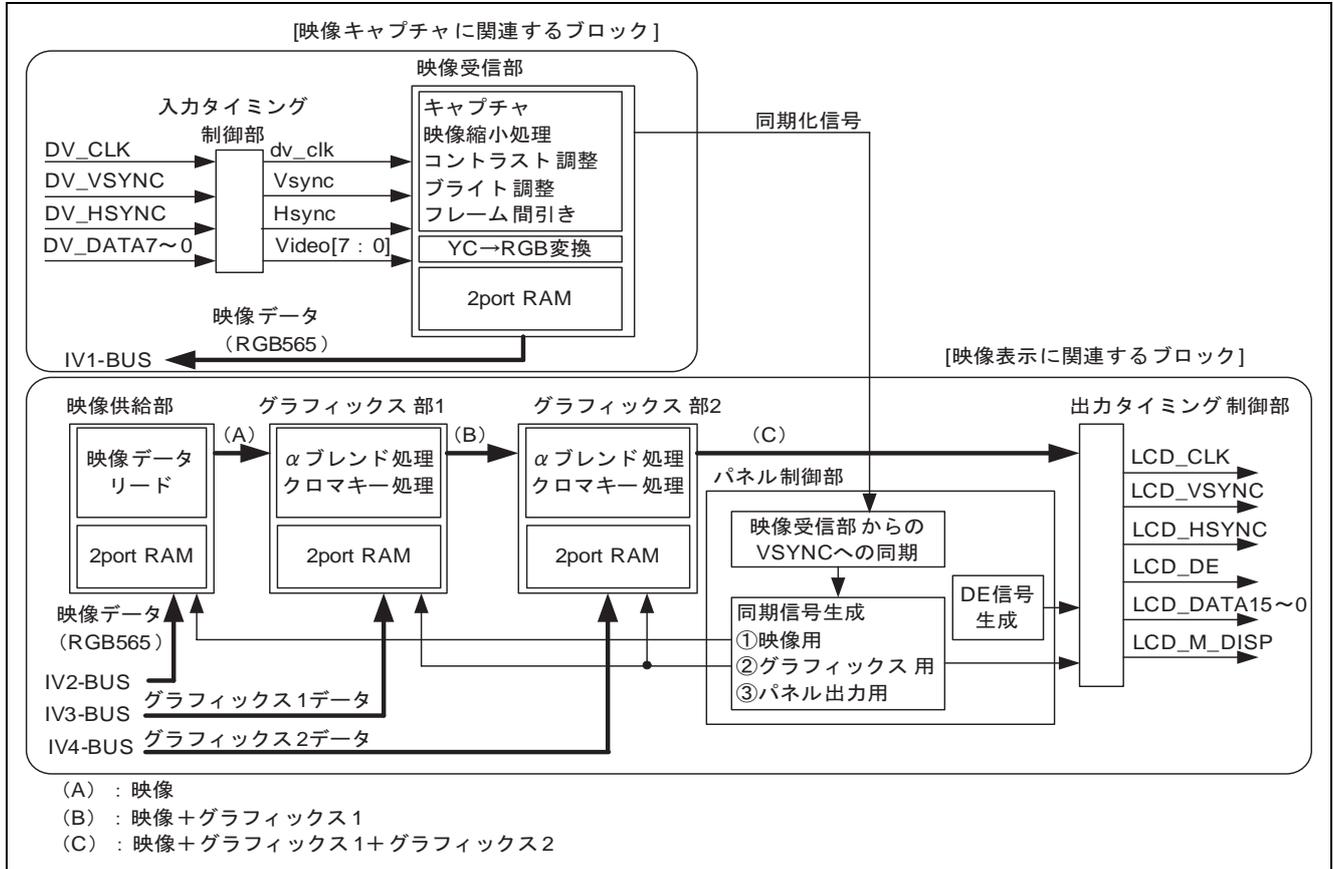


図1 VDC3 ブロック図

表3 VDC3 の機能ブロック

ブロック名	機能概要
入力タイミング制御部	入力同期信号のクロック立ち上がり／立ち下がりエッジのタイミング選択、極性を制御します。BT.601、BT.656 映像入力のクロック立ち上がり／立ち下がりエッジのタイミングを制御します。
映像受信部	(1) 入力映像をキャプチャし、スケーリング、コントラスト、ブライツ処理します。 (2) YC を RGB565 形式に変換しデータを IV1-BUS 経由で保存します。 (3) フィールド間引きを行い、RGB565 形式でデータを IV1-BUS 経由で保存します。
映像供給部	IV2-BUS 経由で映像データの読み出しを行います。
グラフィックス部 1	メモリに格納されたグラフィックス画像 (レイヤ 1) を IV3-BUS 経由で読み出し、映像供給部の映像とオーバーレイ処理を行い、グラフィックス部 2 に出力します。
グラフィックス部 2	メモリに格納されたグラフィックス画像 (レイヤ 2) を IV4-BUS 経由で読み出し、グラフィックス部 1 の出力とオーバーレイ処理を行い、出力タイミング制御部に出力します。
パネル制御部	パネル出力用の同期信号を生成します。
出力タイミング制御部	出力同期信号のクロック立ち上がり／立ち下がりエッジのタイミング選択、極性を制御します。RGB565 映像出力のクロック立ち上がり／立ち下がりエッジのタイミングを制御します。

### 2.1.5 VDC3 のグラフィックス機能

ここでは、グラフィックス機能の  $\alpha$  ブレンド窓機能、クロマキー機能およびドット  $\alpha$  機能について説明します。各機能は独立した透過率  $\alpha$  を持ちます。この透過率  $\alpha$  の値を制御することにより、カレントレイヤと下位レイヤをブレンドすることができます。

また、上記機能の  $\alpha$  値の優先度は、 $\alpha$  ブレンド窓機能 > クロマキー機能 > ドット  $\alpha$  機能です。有効にした機能の中で一番優先度の高い機能の  $\alpha$  値だけが有効になり、それ以外の  $\alpha$  値は無効になります。

#### (1) $\alpha$ ブレンド窓機能の概要

- $\alpha$  制御領域内にあるカレントレイヤと下位レイヤを  $\alpha$  値制御によりブレンドすることが可能
- 入力映像のフィールドレートに同期して、 $\alpha$  値を加算または減算することが可能（フェードイン/フェードアウト）

図 2 に  $\alpha$  ブレンド窓機能のイメージ図を示します。表 4 に  $\alpha$  ブレンド窓機能の  $\alpha$  値によるブレンド比率を示します。

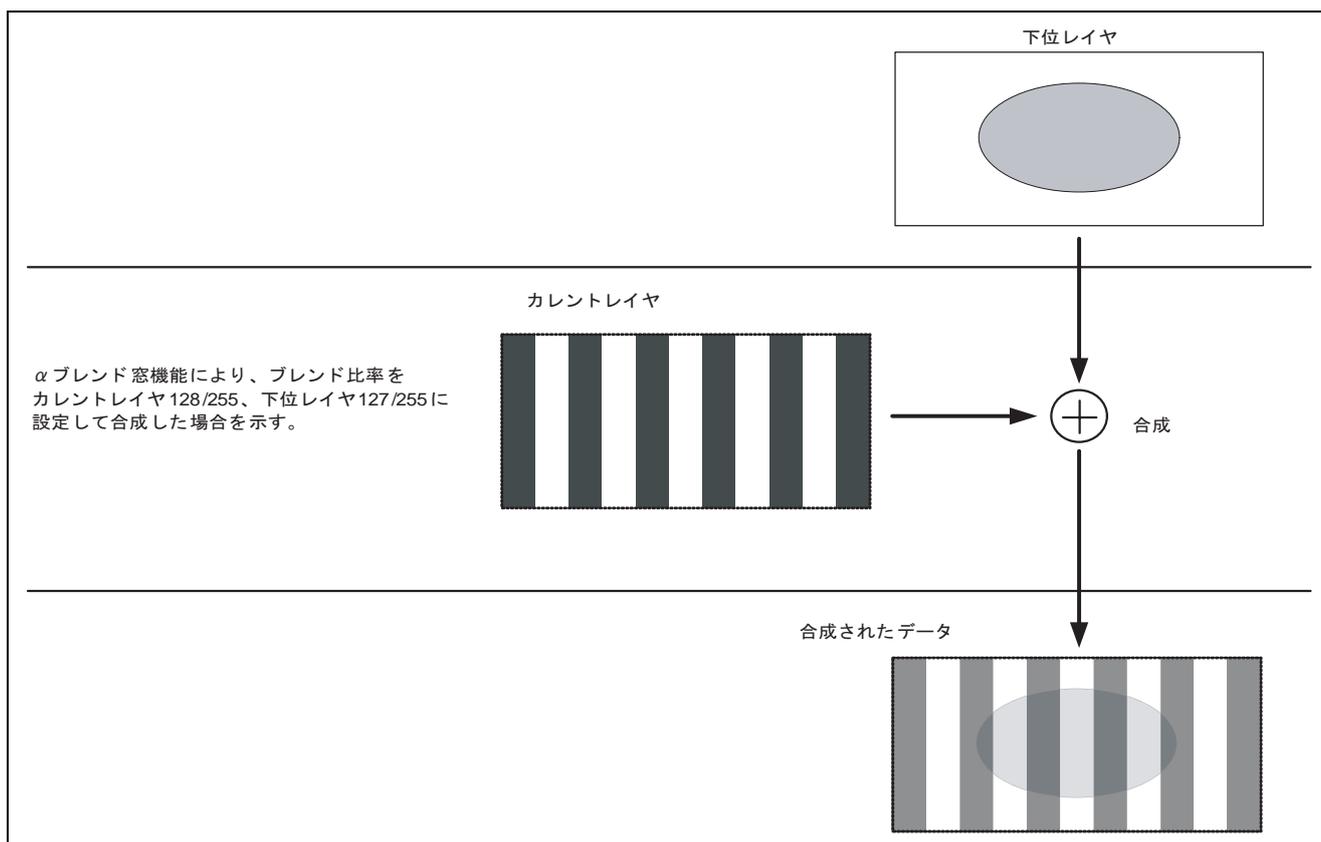


図2  $\alpha$  ブレンド窓機能のイメージ図

表4  $\alpha$  ブレンド窓機能の  $\alpha$  値によるブレンド比率

$\alpha$ 値	カレントレイヤ	下位レイヤ	表示される比率
255	255/255	0/255	<ul style="list-style-type: none"> <li>カレントレイヤ : 100 [ % ]</li> <li>下位レイヤ : 0 [ % ]</li> </ul>
254	254/255	1/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>254/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>1/255 \times 100</math> [ % ]</li> </ul>
253	253/255	2/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>253/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>2/255 \times 100</math> [ % ]</li> </ul>
.	.	.	.
.	.	.	.
.	.	.	.
1	1/255	254/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>1/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>254/255 \times 100</math> [ % ]</li> </ul>
0	0/255	255/255	<ul style="list-style-type: none"> <li>カレントレイヤ : 0 [ % ]</li> <li>下位レイヤ : 100 [ % ]</li> </ul>

## (2) クロマキー機能の概要

クロマキー機能とは、グラフィックス画像の中にクロマキー対象色と一致するピクセルが存在する場合、一致したピクセルデータに対して次の操作を行う機能です。

- クロマ色指定レジスタに設定している色情報を元に色置換を行います。
- 色置換後に、クロマ色指定レジスタの ALPHA[7:0]ビットによる $\alpha$ 制御を行います。

図3にクロマキー機能のイメージ図を示します。表5にクロマキー機能の $\alpha$ 値によるブレンド比率を示します。

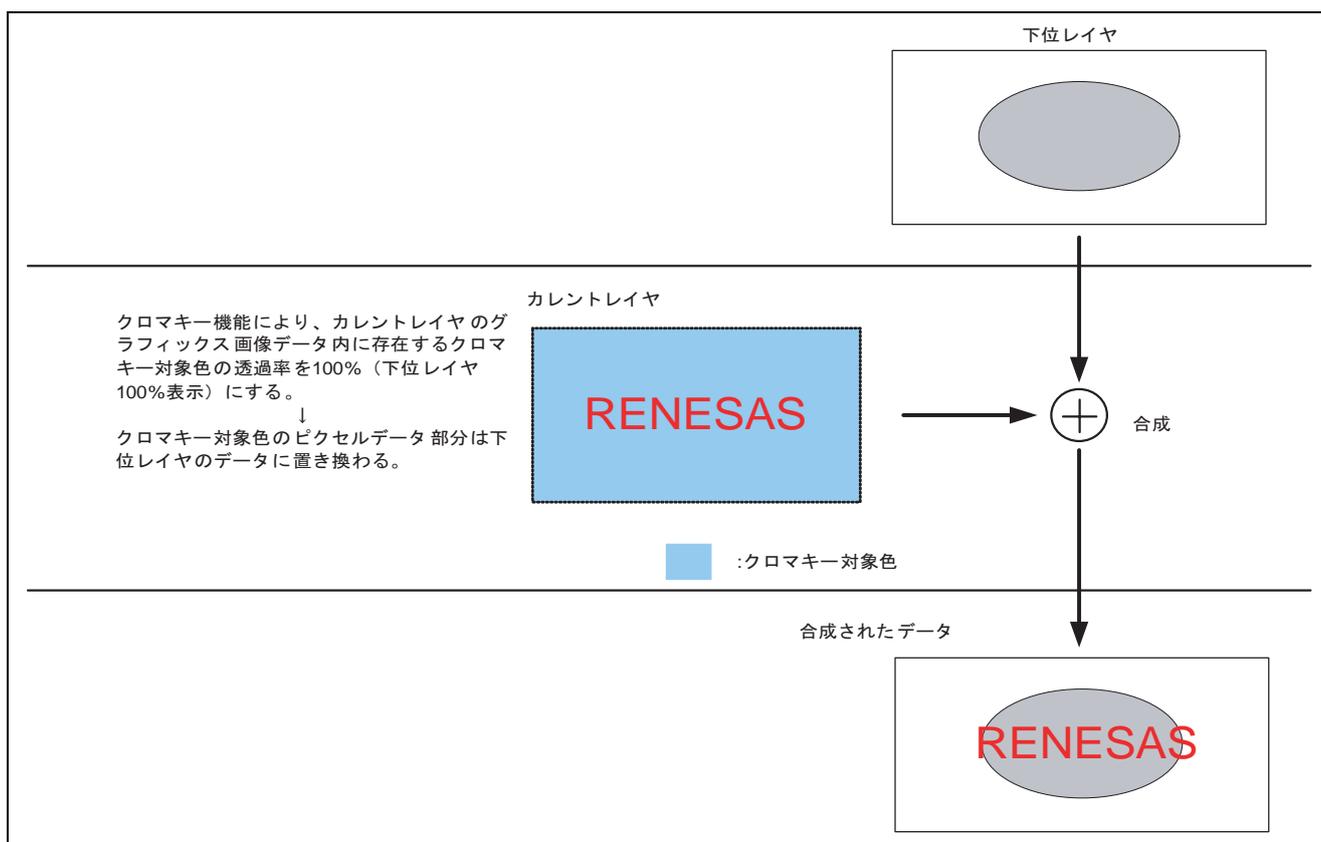


図3 クロマキー機能のイメージ図

表5 クロマキー機能の  $\alpha$  値によるブレンド比率

$\alpha$ 値	カレントレイヤ	下位レイヤ	表示される比率
255	255/255	0/255	<ul style="list-style-type: none"> <li>カレントレイヤ : 100 [ % ]</li> <li>下位レイヤ : 0 [ % ]</li> </ul>
254	254/255	1/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>254/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>1/255 \times 100</math> [ % ]</li> </ul>
253	253/255	2/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>253/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>2/255 \times 100</math> [ % ]</li> </ul>
.	.	.	.
.	.	.	.
.	.	.	.
1	1/255	254/255	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>1/255 \times 100</math> [ % ]</li> <li>下位レイヤ : <math>254/255 \times 100</math> [ % ]</li> </ul>
0	0/255	255/255	<ul style="list-style-type: none"> <li>カレントレイヤ : 0 [ % ]</li> <li>下位レイヤ : 100 [ % ]</li> </ul>

ビデオディスプレイコントローラ 3  $\alpha$  ブレンド窓機能の使用例(3) ドット  $\alpha$  機能の概要

ドット  $\alpha$  機能とは、グラフィックス画像のフォーマットを  $\alpha$  RGB4444 形式にした場合、透過率  $\alpha$  に基づき下位レイヤと合成する機能です。図 4 に  $\alpha$  RGB4444 形式グラフィックス画像のフォーマットを示します。 $\alpha$  RGB4444 形式グラフィックス画像データの上位 4 ビットを使用して  $\alpha$  値を設定します。表 6 にドット  $\alpha$  機能の  $\alpha$  値によるブレンド比率を示します。

グラフィックス画像のデータを  $\alpha$  RGB4444 形式にする場合は、バス制御レジスタの BUS\_FORMAT ビットを 1 に設定してください。

図4  $\alpha$  RGB4444 形式グラフィックス画像のフォーマット表6 ドット  $\alpha$  機能の  $\alpha$  値によるブレンド比率

$\alpha$ 値	カレントレイヤ	下位レイヤ	表示される比率
15	15/15	0/15	<ul style="list-style-type: none"> <li>カレントレイヤ : 100 [%]</li> <li>下位レイヤ : 0 [%]</li> </ul>
14	14/15	1/15	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>14/15 \times 100</math> [%]</li> <li>下位レイヤ : <math>1/15 \times 100</math> [%]</li> </ul>
13	13/15	2/15	.
.	.	.	.
.	.	.	.
2	2/15	13/15	.
1	1/15	14/15	<ul style="list-style-type: none"> <li>カレントレイヤ : <math>1/15 \times 100</math> [%]</li> <li>下位レイヤ : <math>14/15 \times 100</math> [%]</li> </ul>
0	0/15	15/15	<ul style="list-style-type: none"> <li>カレントレイヤ : 0 [%]</li> <li>下位レイヤ : 100 [%]</li> </ul>

## 2.2 応用例の回路接続例

本応用例では、 $\alpha$ ブレンド窓機能を使用して入力映像とグラフィックス画像を合成し、TFT-LCDパネルに出力します。図5にSH7264とデジタルビデオデコーダの接続例を示します。また、図6にSH7264とTFT-LCDパネルとの接続例を示します。

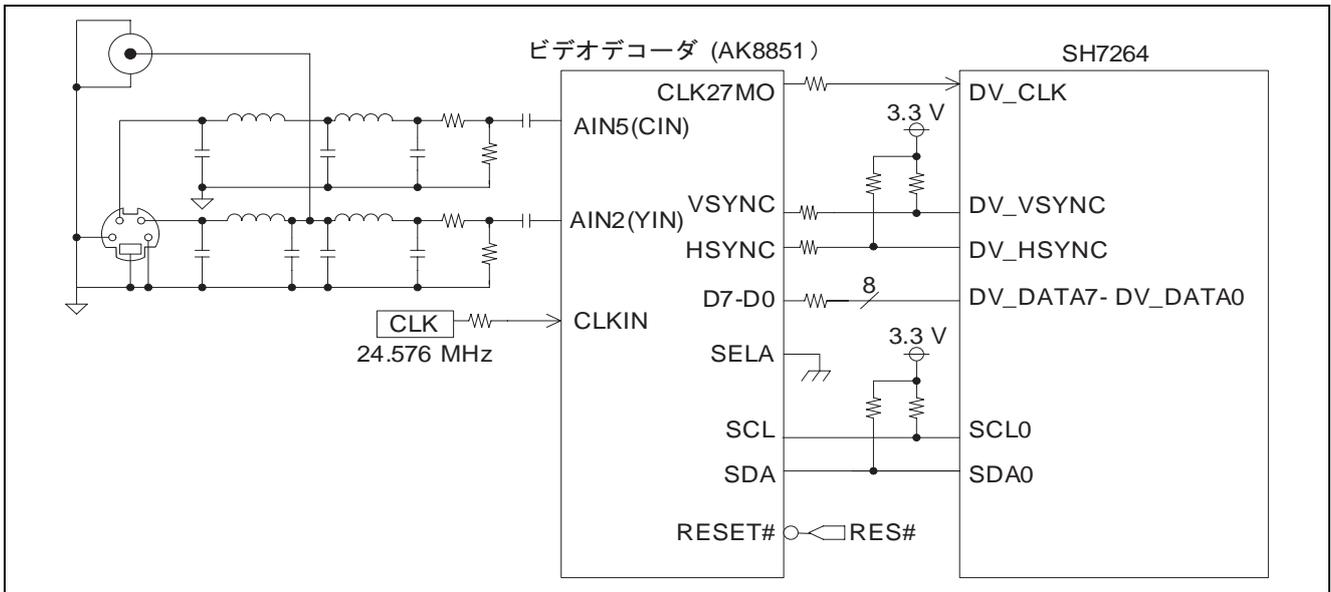


図5 デジタルビデオデコーダとの接続例

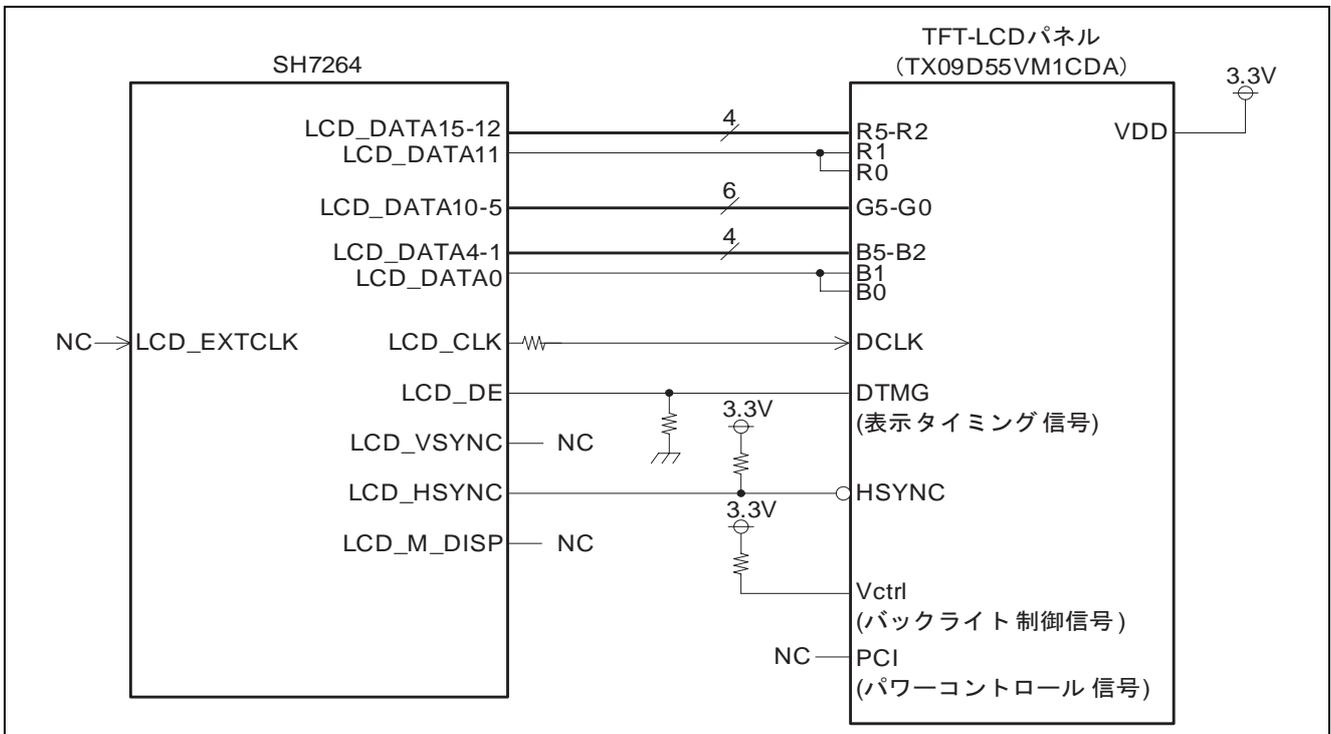


図6 TFT-LCD パネルとの接続例

## 2.3 参考プログラムの仕様

ここでは参考プログラムの仕様と各処理のフローチャートを説明します。

### 2.3.1 仕様

- 映像、グラフィックス1およびグラフィックス2を $\alpha$ ブレンド窓機能により重ね合わせた後、QVGAサイズ（縦320×横240）のTFT-LCDパネルに表示
- BT.656（NTSC）形式の映像を入力し、縦1/3、横1/3に縮小し、縦160×横240のサイズでパネル中央に表示
- グラフィックス画像1は、サイズは80×80、表示開始位置は（60、100）、黒の画像データ
- グラフィックス画像2は、サイズは80×80、表示開始位置は（100、140）、白の画像データ
- グラフィックス1の $\alpha$ 制御： $\alpha$ 値の初期値を255とし、1フレームレート毎に $\alpha$ 値を減算してフェードイン処理
- グラフィックス2の $\alpha$ 制御： $\alpha$ 値の初期値を0とし、1フレームレート毎に $\alpha$ 値を加算してフェードアウト処理
- クロマキー機能、ドット $\alpha$ 機能は無効

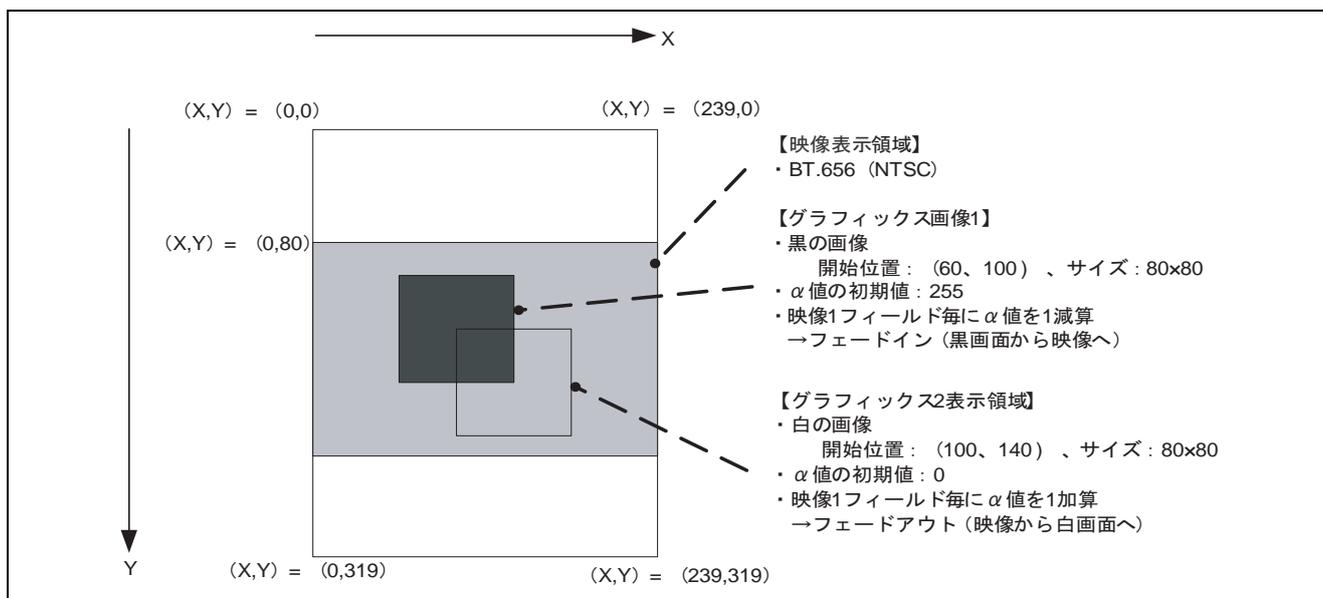


図7 参考プログラムの動作イメージ図

### 2.3.2 参考プログラムのメインフロー

図 8に参考プログラムのメインフローを示します。図 9から 図 17に示す初期化処理を実行して $\alpha$ ブレンドした映像と画像をTFT-LCDパネルに表示します。

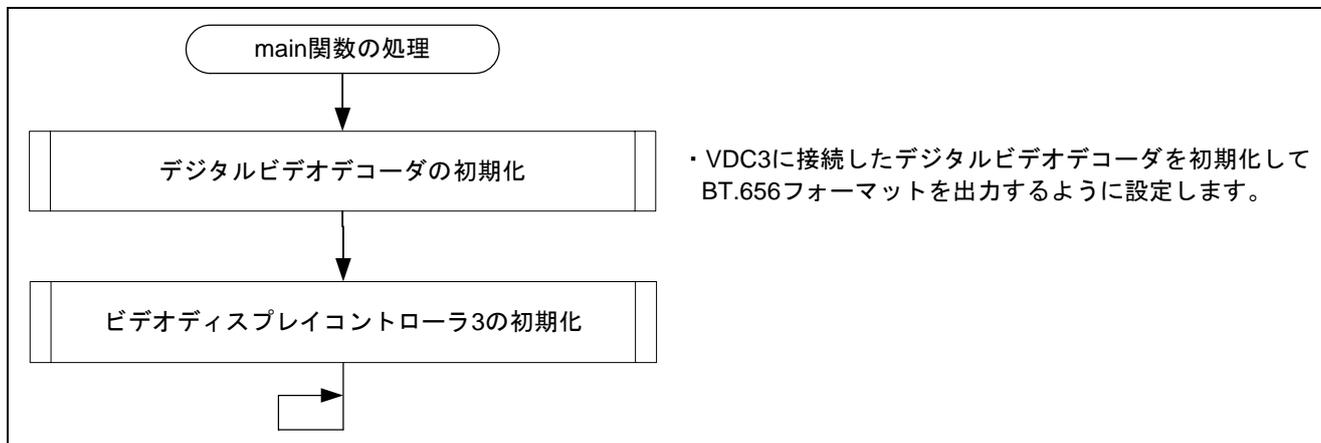


図8 参考プログラムのメインフロー

## 2.3.3 VDC3 の初期化

図 9にVDC3 の初期化フローを示します。

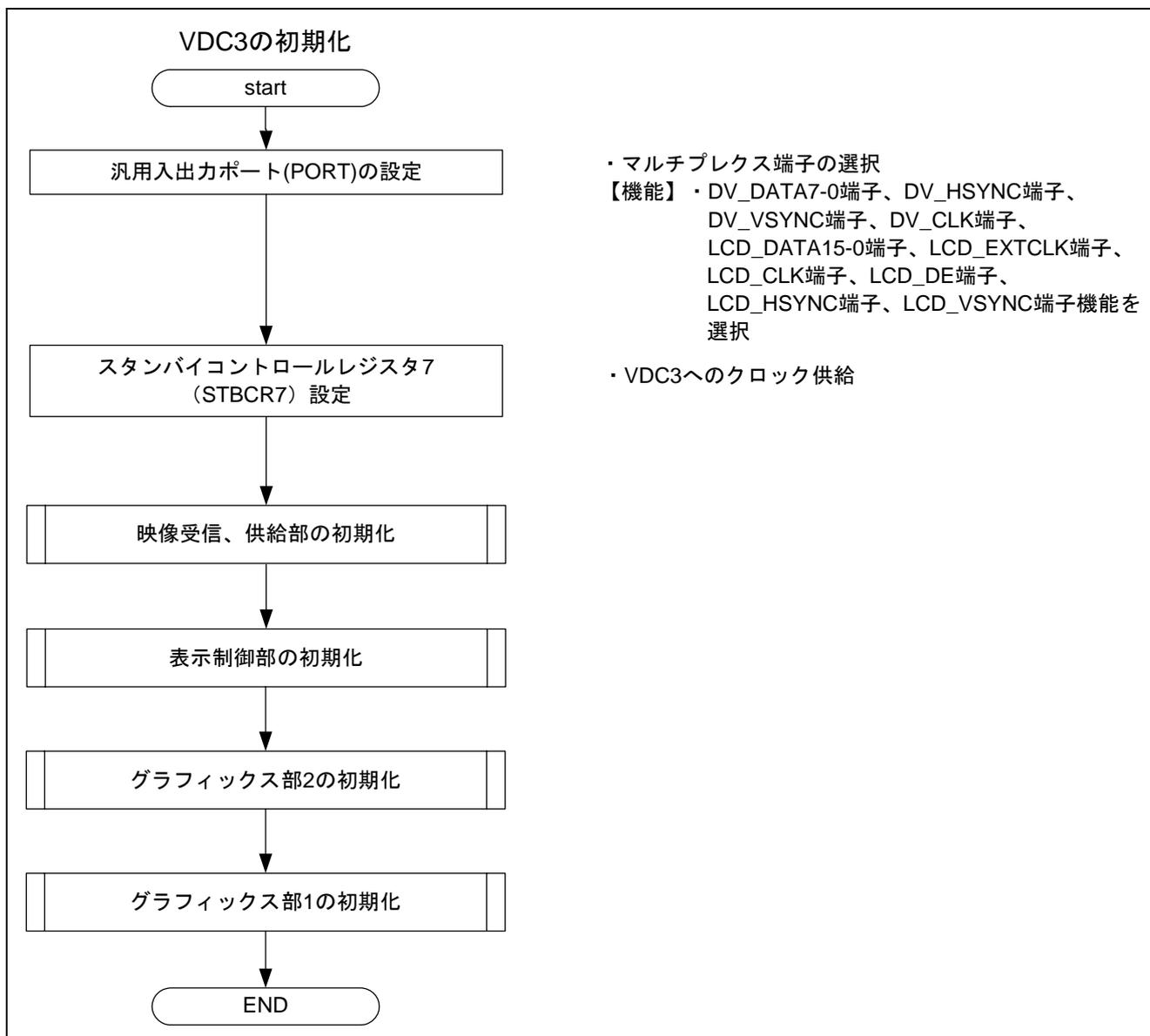


図9 VDC3 の初期化フロー

ビデオディスプレイコントローラ 3  $\alpha$  ブレンド窓機能の使用例

## 2.3.4 映像受信、供給部の初期化

図 10および図 11に映像受信、供給部の初期化フローを示します。

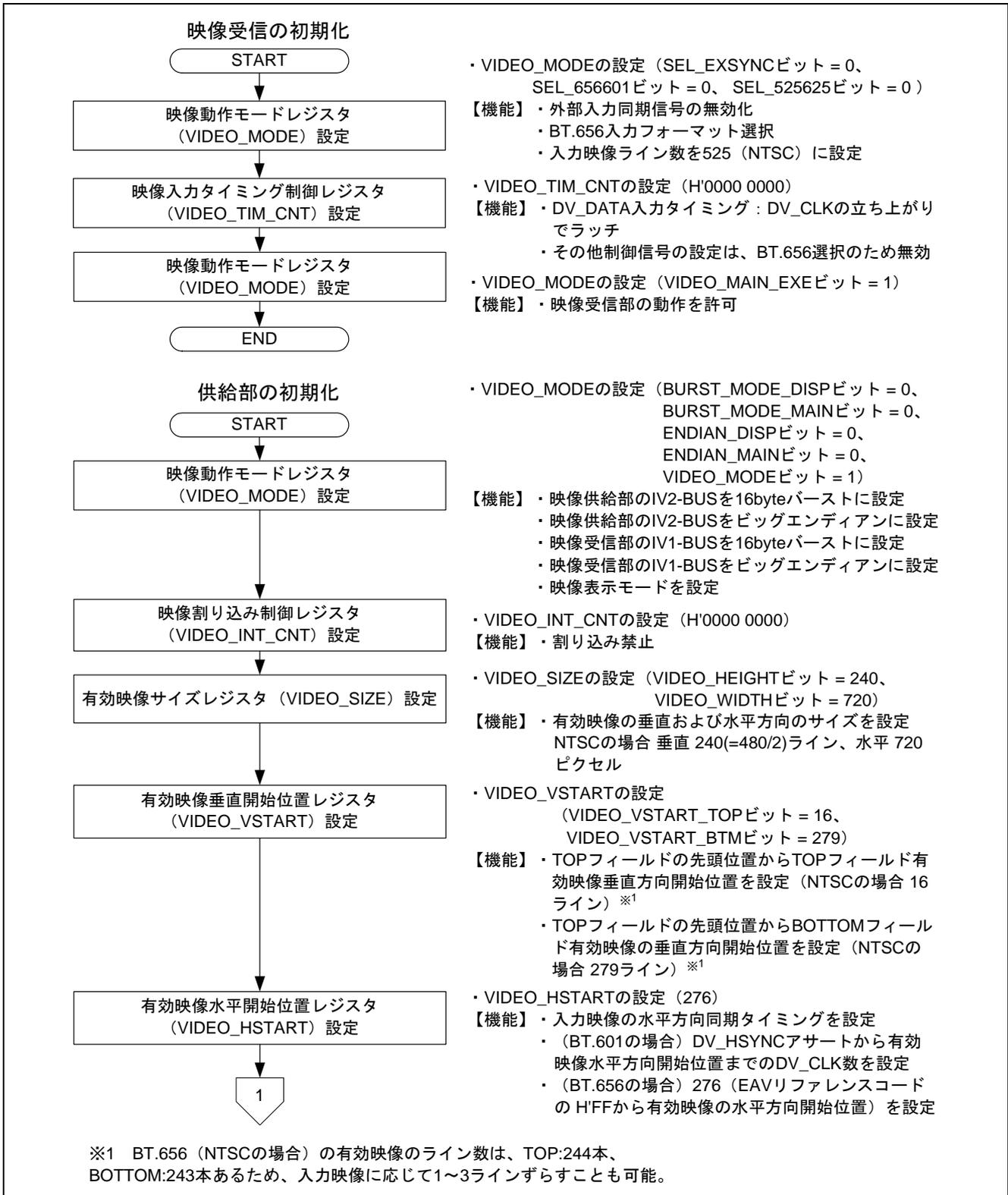


図10 映像受信、供給部の初期化フロー (1)

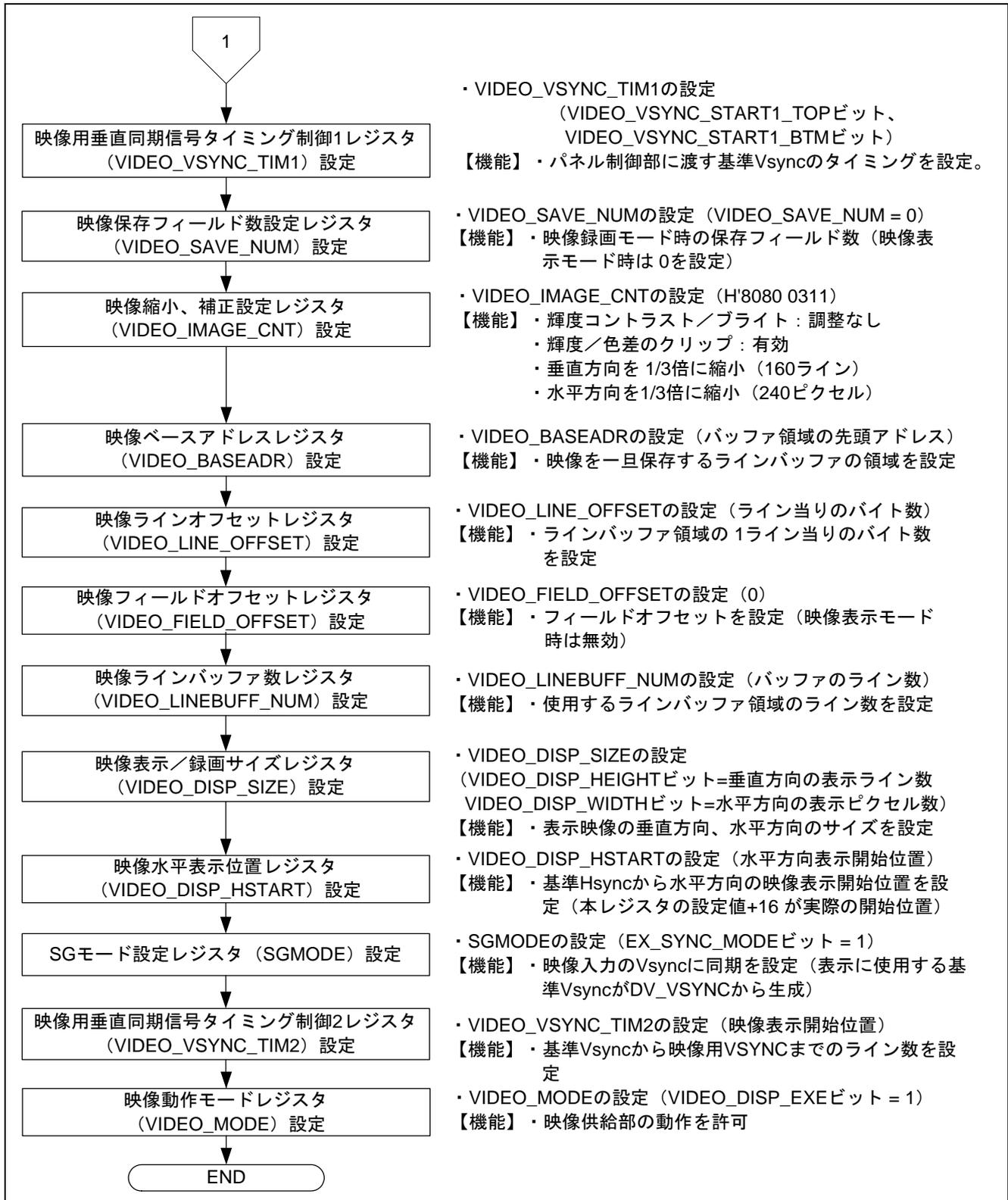
ビデオディスプレイコントローラ 3  $\alpha$  ブレンド窓機能の使用例

図11 映像受信、供給部の初期化フロー (2)

## 2.3.5 表示制御部の初期化

図 12および図 13に表示制御部の初期化フローを示します。

なお図 12および図 13の設定は、本応用例で使用するTFT-LCDパネルの仕様に合わせています。使用するTFT-LCDパネルの仕様に合わせた設定値を検討してください。

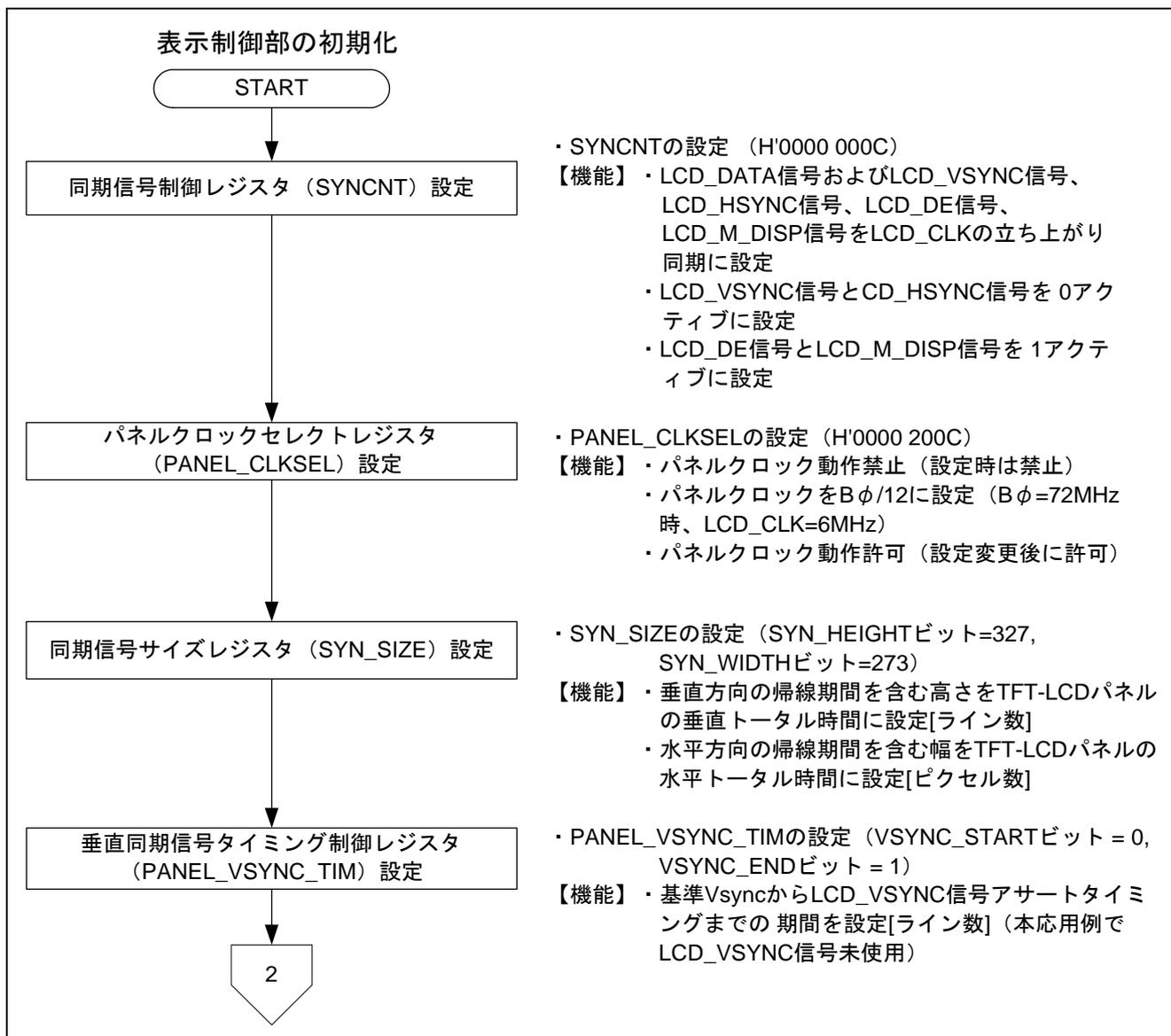


図12 表示制御部の初期化フロー (1)

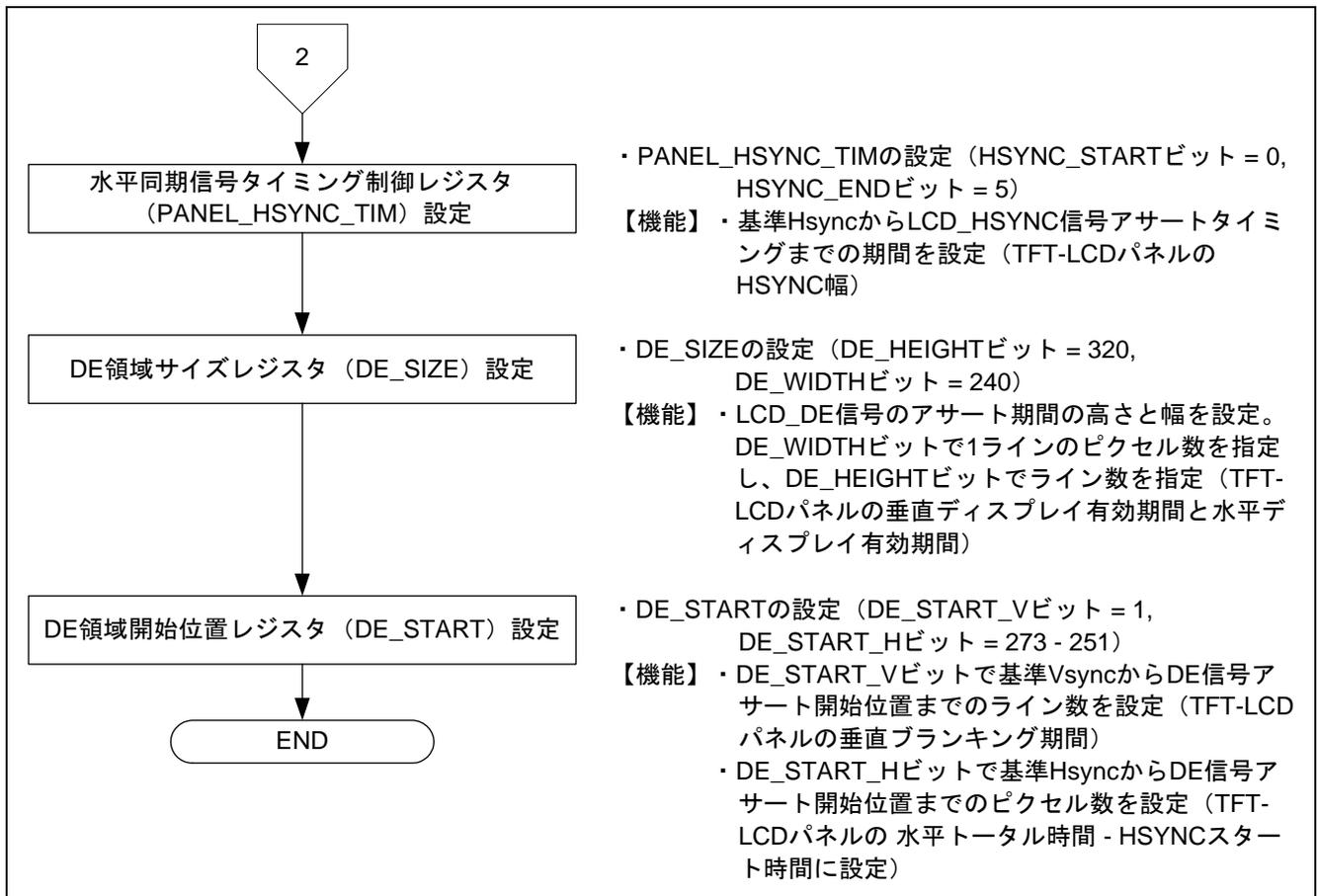


図13 表示制御部の初期化フロー (2)

## 2.3.6 グラフィックス部 1 の初期化

図 14および図 15にグラフィックス部 1 の初期化フローを示します。

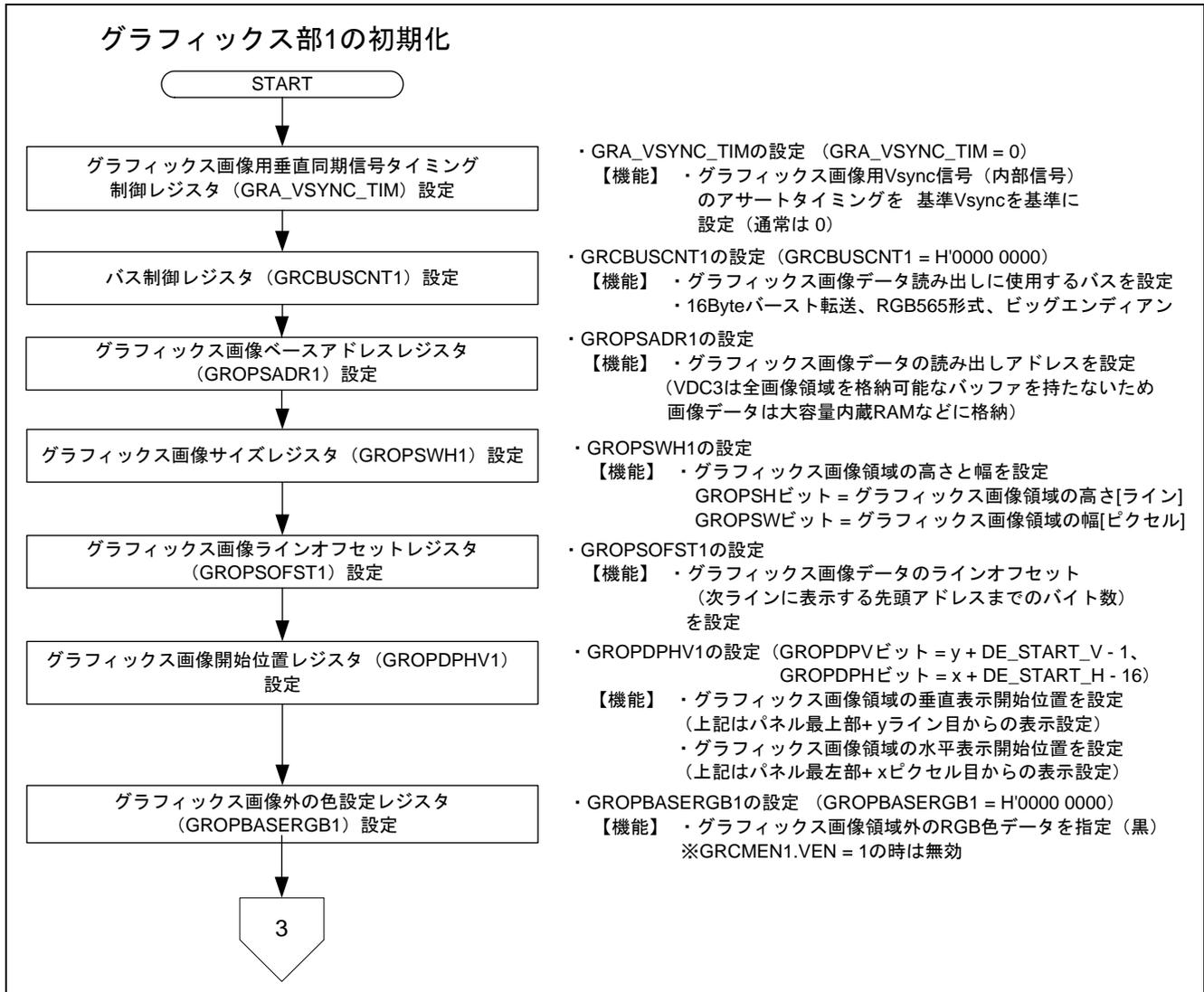


図14 グラフィックス部 1 の初期化フロー (1)

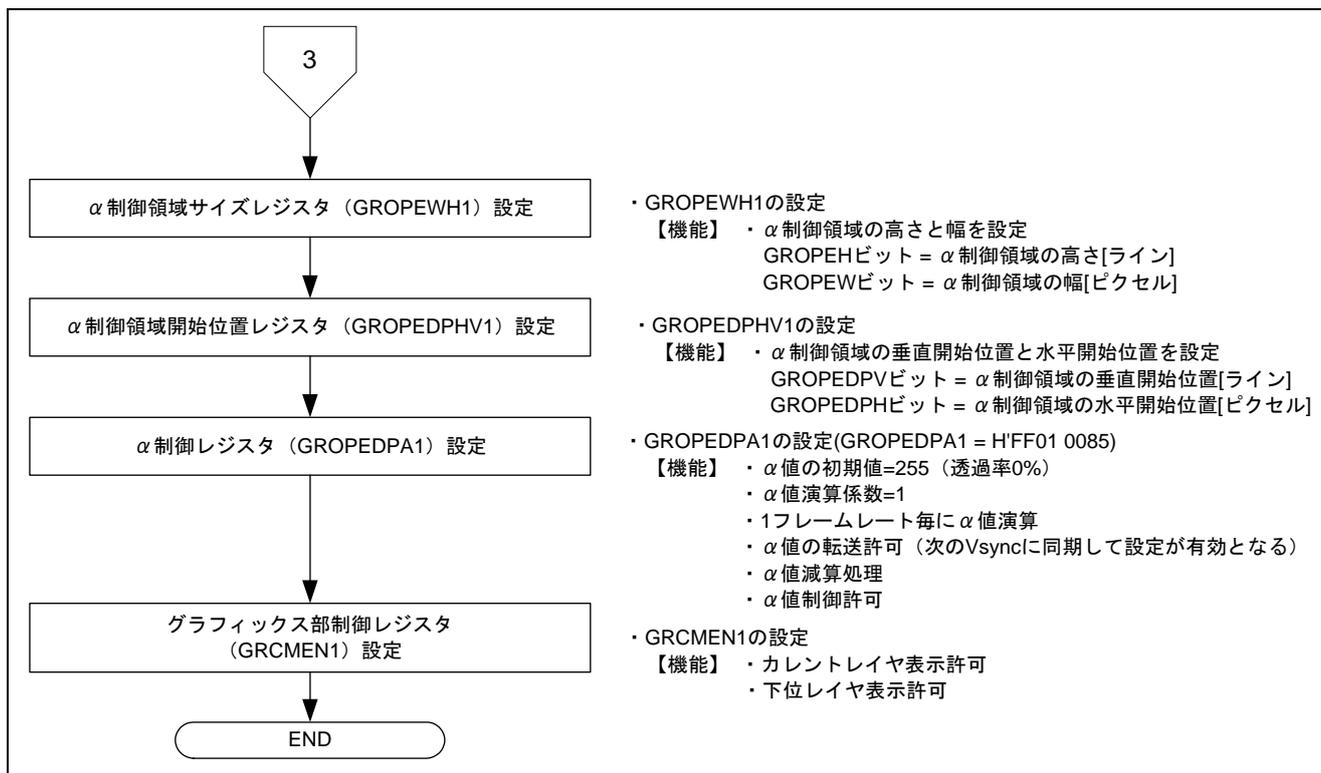


図15 グラフィックス部 1 の初期化フロー (2)

## 2.3.7 グラフィックス部 2 の初期化

図 16および図 17にグラフィックス部 2 の初期化フローを示します。

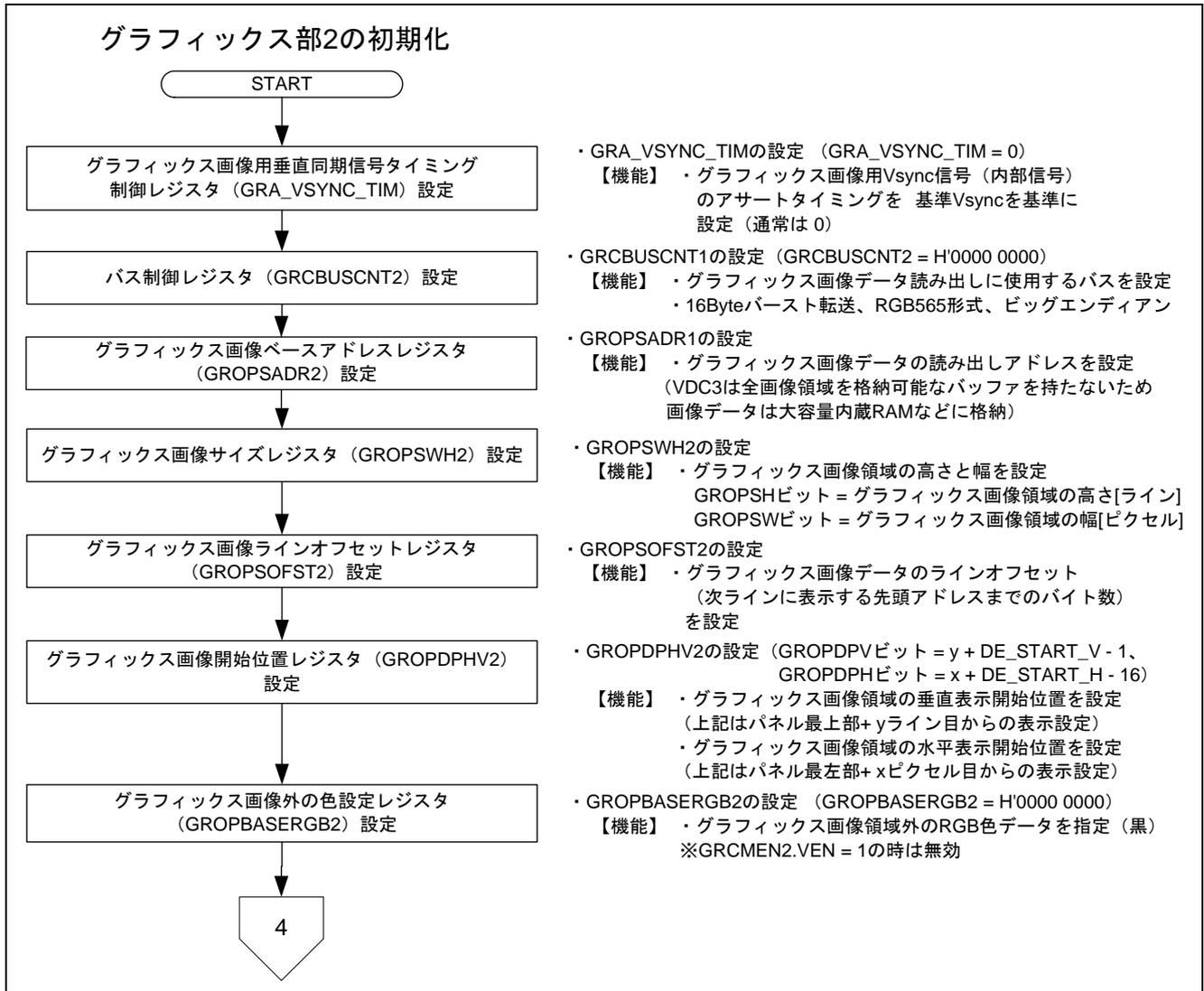


図16 グラフィックス部 2 の初期化フロー (1)

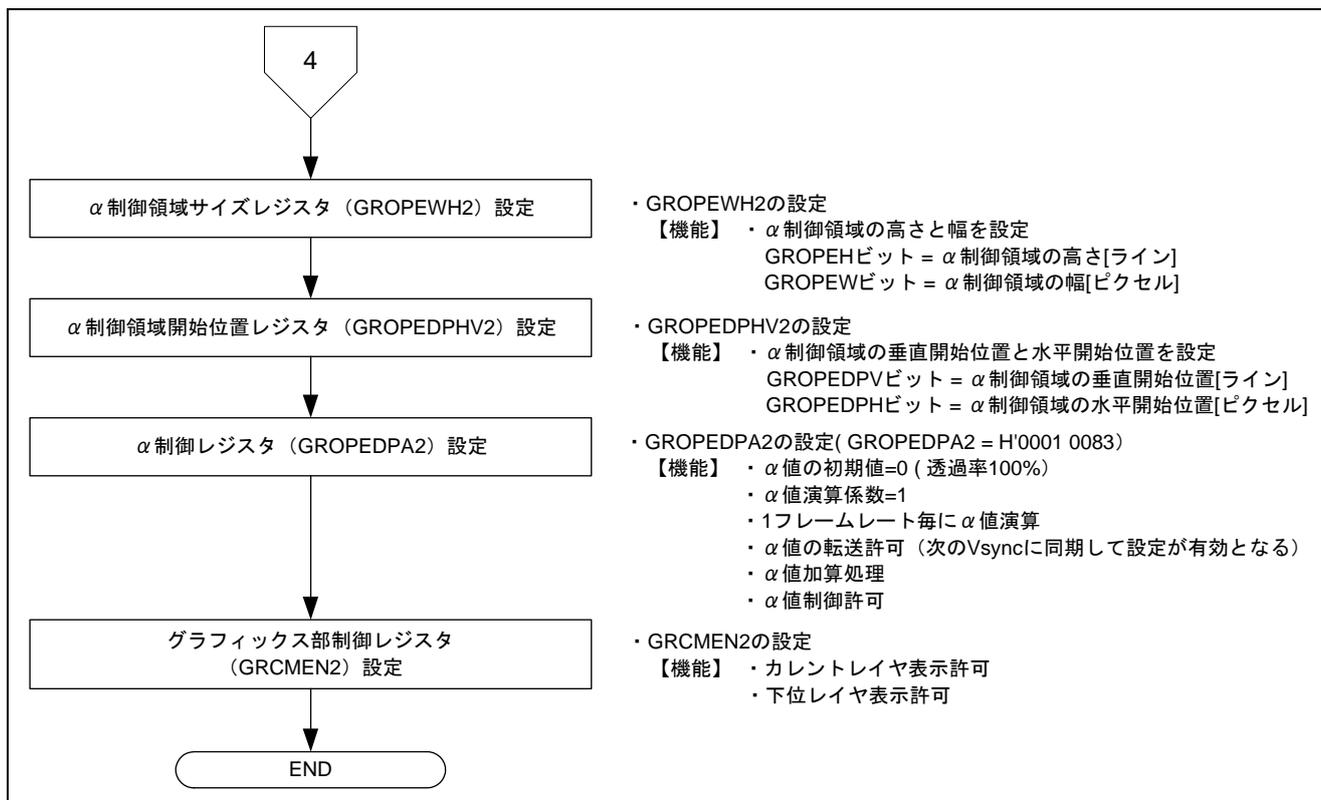
ビデオディスプレイコントローラ 3  $\alpha$  ブレンド窓機能の使用例

図17 グラフィックス部 2 の初期化フロー (2)

### 3. 参考プログラム例

#### 3.1 参考プログラムについての補足

SH7264 は、製品によって大容量内蔵 RAM の容量が 1MB または 640KB と異なるため、参考プログラムのセクション配置やレジスタの設定を一部変更する必要があります。そのため本アプリケーションノートでは 1MB 用と 640KB 用の 2 つのワークスペースを用意しています。

640KB 版はライトプロテクトを解除しなければ保持用内蔵 RAM へ書き込むことができないため、640KB 版のワークスペースは、システムコントロールレジスタ 5 (SYSCR5) にライトプロテクトの解除を設定しています。

使用する製品を確認した上で、対応するワークスペースを使用してください。

## 3.2 サンプルプログラムリスト "main.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 /* Copyright (C) 2009(2010,2011) Renesas Electronics Corporation. All Rights Reserved.*/
29 /*****
30 *   System Name : SH7264 Sample Program
31 *   File Name   : main.c
32 *   Abstract    : VDC3  $\alpha$  ブレンド窓機能の使用例
33 *   Version     : 2.00.00
34 *   Device      : SH7264
35 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *               : C/C++ compiler package for the SuperH RISC engine family
37 *               :                               (Ver.9.03 Release00).
38 *   OS          : None
39 *   H/W Platform: M3A-HS64G50(CPU board), M3A-HS64G02(Option board)
40 *   Description :
41 *****/
42 *   History     : Mar.12,2009 Ver.1.00.00
43 *               : Jan.14,2010 Ver.1.01.00 ヘッダファイルのフォーマット変更
44 *               : Feb.28,2011 Ver.2.00.00 VDC3 のコード修正に対応
45 *****/
46
47
```

### 3.3 サンプルプログラムリスト"main.c" (2)

```
48  /*****
49  Includes <System Includes> , "Project Includes"
50  *****/
51  #include <stdio.h>
52  #include "io_vdc3_alpha_blend.h"
53
54  /*****
55  Exported global variables and functions (to be accessed by other files)
56  *****/
57  /* ==== Global functions ==== */
58  void main(void);
59
60  /*****
61  * ID          :
62  * Outline     :  $\alpha$ ブレンド窓機能の使用例メイン
63  * Include     :
64  * Declaration : void main(void);
65  * Description : 本関数は、 $\alpha$ ブレンド窓機能の設定をします。
66  * Argument    : void
67  * Return Value : void
68  *****/
69  void main(void)
70  {
71      /* ==== デジタルビデオデコーダの初期化 ==== */
72      init_video_decoder();
73
74      /* ==== VDC3 の初期化 ==== */
75      io_vdc3_init();
76
77      while(1){
78          /* loop */
79      }
80
81  }
82
83  /* End of File */
```

## 3.4 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (1)

```
1  /*****
2  *   DISCLAIMER
3  *
4  *   This software is supplied by Renesas Electronics Corporation and is only
5  *   intended for use with Renesas products. No other uses are authorized.
6  *
7  *   This software is owned by Renesas Electronics Corporation and is protected under
8  *   all applicable laws, including copyright laws.
9  *
10 *   THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *   REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *   INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *   PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *   DISCLAIMED.
15 *
16 *   TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *   ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *   FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *   FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *   AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *   Renesas reserves the right, without notice, to make changes to this
23 *   software and to discontinue the availability of this software.
24 *   By using this software, you agree to the additional terms and
25 *   conditions found by accessing the following link:
26 *   http://www.renesas.com/disclaimer
27 *****/
28 /* Copyright (C) 2011 Renesas Electronics Corporation. All Rights Reserved.*/
29 /*****
30 *   System Name : SH7264 Sample Program
31 *   File Name   : io_vdc3_alpha_blend.c
32 *   Abstract    : VDC3  $\alpha$ ブレンド窓機能の使用例
33 *   Version     : 1.00.00
34 *   Device      : SH7264
35 *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *               : C/C++ compiler package for the SuperH RISC engine family
37 *               :                               (Ver.9.03 Release00).
38 *   OS          : None
39 *   H/W Platform: M3A-HS64G50(CPU board), M3A-HS64G02(Option board)
40 *   Description :
41 *****/
42 *   History     : Feb.28,2011 Ver.1.00.00 VDC3の既存サンプルコードを統合して作成
43 *****/
44
45
```

## 3.5 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (2)

```
46  /*****
47  Includes <System Includes> , "Project Includes"
48  *****/
49  #include "iodefine.h"
50  #include "io_vdc3_alpha_blend.h"
51
52  /*****
53  Exported global variables and functions (to be accessed by other files)
54  *****/
55  /* ==== Global functions ==== */
56  void io_vdc3_init(void);
57  void io_vdc3_change_buffer( int grphcs_no, unsigned short *buffer );
58
59  /* ==== Global variables ==== */
60  #pragma section GRPH1_BUFF /* バッファはキャッシュ無効空間の 128 バイトまたは 16 バイト境界に配置 */
61  unsigned short grph_buffer1[2][GRPHCS1_Y_SIZE][ (GRPHCS1_LINE_OFFSET / BYTES_PER_PIXEL) ];
62  #pragma section
63
64  #pragma section GRPH2_BUFF /* バッファはキャッシュ無効空間の 128 バイトまたは 16 バイト境界に配置 */
65  unsigned short grph_buffer2[2][GRPHCS2_Y_SIZE][ (GRPHCS2_LINE_OFFSET / BYTES_PER_PIXEL) ];
66  #pragma section
67
68  #pragma section VLINE_BUFF /* バッファはキャッシュ無効空間の 128 バイトまたは 16 バイト境界に配置 */
69  unsigned short video_line_buffer[VOUT_LINEBUF_NUM][ (VOUT_LINE_OFFSET / BYTES_PER_PIXEL) ];
70  #pragma section
71
72
73  /*****
74  Private global variables and functions
75  *****/
76  /* ==== Private fuctions ==== */
77  static void io_vdc3_init_video_in(void);
78  static void io_vdc3_init_video_out(void);
79  static void io_vdc3_init_grphcs1(void);
80  static void io_vdc3_init_grphcs2(void);
81  static void io_vdc3_init_disp(void);
82  static void io_vdc3_start(void);
83
```

## 3.6 サンプルプログラムリスト "io\_vdc3\_alpha\_blend.c" (3)

```
84  /*****
85  * ID      :
86  * Outline : VDC3 の初期化
87  * Include :
88  * Declaration : void io_vdc3_init(void);
89  * Description : 本関数は $\alpha$ ブレンド窓機能を行うために VDC3 を初期化します。
90  * Argument  : void
91  * Return Value : void
92  *****/
93 void io_vdc3_init(void)
94 {
95     int i, j;
96
97     /* ==== データの初期化 ==== */
98     /* ---- グラフィックス画像領域 1 の画像作成 ---- */
99     for( i = 0; i < GRPHCS1_Y_SIZE; i++){
100         for( j = 0 ; j < GRPHCS1_X_SIZE; j++){
101             grph_buffer1[0][i][j] = RGB565_BLACK;
102             grph_buffer1[1][i][j] = RGB565_BLACK;
103         }
104     }
105     /* ---- グラフィックス画像領域 2 の画像作成 ---- */
106     for( i = 0; i < GRPHCS2_Y_SIZE; i++){
107         for( j = 0 ; j < GRPHCS2_X_SIZE; j++){
108             grph_buffer2[0][i][j] = RGB565_WHITE;
109             grph_buffer2[1][i][j] = RGB565_WHITE;
110         }
111     }
112
113     /* ==== PORT ==== */
114     /* ---- Video (in) ---- */
115     PORT.PFCR1.BIT.PF7MD = 3;      /* DV_DATA7 */
116     PORT.PFCR1.BIT.PF6MD = 3;      /* DV_DATA6 */
117     PORT.PFCR1.BIT.PF5MD = 3;      /* DV_DATA5 */
118     PORT.PFCR1.BIT.PF4MD = 3;      /* DV_DATA4 */
119     PORT.PFCR0.BIT.PF3MD = 3;      /* DV_DATA3 */
120     PORT.PFCR0.BIT.PF2MD = 3;      /* DV_DATA2 */
121     PORT.PFCR0.BIT.PF1MD = 3;      /* DV_DATA1 */
122     PORT.PFCR0.BIT.PF0MD = 3;      /* DV_DATA0 */
123     PORT.PECR1.BIT.PE5MD = 3;      /* DV_HSYNC */
124     PORT.PECR1.BIT.PE4MD = 3;      /* DV_VSYNC */
125     PORT.PFCR2.BIT.PF8MD = 3;      /* DV_CLK */
126
```

## 3.7 サンプルプログラムリスト "io\_vdc3\_alpha\_blend.c" (4)

```
127     /* ---- Display (out) ---- */
128     PORT.PGCR7.WORD = 0x5A01u;      /* LCD_DATA0 書き込み時はビット操作不可*/
129     PORT.PGCR5.BIT.PG20MD= 1;      /* LCD_EXTCLK */
130     PORT.PGCR4.WORD = 0x1111u;     /* LCD_CLK, LCD_DE, LCD_HSYNC, LCD_VSYNC */
131     PORT.PGCR3.WORD = 0x1111u;     /* LCD_DATA15-12 */
132     PORT.PGCR2.WORD = 0x1111u;     /* LCD_DATA11-08 */
133     PORT.PGCR1.WORD = 0x1111u;     /* LCD_DATA07-04 */
134     PORT.PGCR0.BIT.PG3MD = 1;      /* LCD_DATA03 */
135     PORT.PGCR0.BIT.PG2MD = 1;      /* LCD_DATA02 */
136     PORT.PGCR0.BIT.PG1MD = 1;      /* LCD_DATA01 */
137
138     /* ==== CPG ==== */
139     CPG.STBCR7.BIT.MSTP74 = 0;      /* VDC3 のクロック供給許可 */
140
141     /* ==== VDC3 ==== */
142     /* ---- 入力タイミング制御部、映像受信の初期化 ---- */
143     io_vdc3_init_video_in();
144
145     /* ---- 映像供給部の初期化 ---- */
146     io_vdc3_init_video_out();
147
148     /* ---- グラフィックス部 1 の初期化 ---- */
149     io_vdc3_init_grphcs1();
150
151     /* ---- グラフィックス部 2 の初期化 ---- */
152     io_vdc3_init_grphcs2();
153
154     /* ---- パネル制御部、出力タイミング制御部の初期化 ---- */
155     io_vdc3_init_disp();
156
157     /* ---- 動作開始 ---- */
158     io_vdc3_start();
159 }
160
```

## 3.8 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (5)

```

161  /*****
162  * ID      :
163  * Outline  : バッファ領域の変更
164  * Include  :
165  * Declaration : void io_vdc3_change_buffer( int grphcs_no, unsigned short *buffer);
166  * Description : 本関数はグラフィックス1または2の表示バッファ領域を変更します。
167  * Argument  : int grphcs_no ; I : グラフィックス画像の選択
168  *           :                   : (1:グラフィックス1, 2:グラフィックス2)
169  *           : unsigned short * buffer ; I : 画像データの読み出しアドレス
170  * Return Value : void
171  *****/
172  void io_vdc3_change_buffer( int grphcs_no, unsigned short *buffer )
173  {
174      if( grphcs_no == 1 ){
175          VDC3.GROPSADR1.LONG = (0x1FFFFFFFul & (unsigned long)buffer);
176          VDC3.GRCMEN1.BIT.WE = 1;      /* グラフィックス設定を転送(次のVsync から有効)*/
177      }
178      else{
179          VDC3.GROPSADR2.LONG = (0x1FFFFFFFul & (unsigned long)buffer);
180          VDC3.GRCMEN2.BIT.WE = 1;      /* グラフィックス設定を転送(次のVsync から有効)*/
181      }
182  }
183
184  /*****
185  * ID      :
186  * Outline  : 入力タイミング制御部、映像受信の初期化
187  * Include  : iodef.h
188  * Declaration : static void io_vdc3_init_video_in(void);
189  * Description : 本関数は映像入力を行うためにVDC3を初期化します。
190  *           : 入力映像の規格は、BT.656として設定します。
191  * Argument  : void
192  * Return Value : void
193  *****/
194  static void io_vdc3_init_video_in(void)
195  {
196      /* ---- 入力映像フォーマットの設定 ---- */
197      VDC3.VIDEO_MODE.BIT.SEL_EXSYNC = 0;      /* 外部入力の同期信号を無効化 */
198      VDC3.VIDEO_MODE.BIT.SEL_656601 = 0;     /* BT.656 入力を選択 */
199      VDC3.VIDEO_MODE.BIT.SEL_525625 = 0;     /* 入力映像のライン数：525ライン(NTSC) */
200      VDC3.VIDEO_TIM_CNT.LONG = 0x00000000ul; /* DV_DATA 入力信号を立ち上がりでラッチ */
201                                          /* その他は BT656 のため設定不要 */
202  }
203

```

## 3.9 サンプルプログラムリスト "io\_vdc3\_alpha\_blend.c" (6)

```

204  /*****
205  * ID      :
206  * Outline : 映像供給部の初期化
207  * Include : iodef.h
208  * Declaration : static void io_vdc3_init_video_out(void);
209  * Description : 本関数は映像表示を行うために VDC3 を初期化します。
210  *           : 入力映像の規格は、BT.656 として設定します。
211  * Argument : void
212  * Return Value : void
213  *****/
214  static void io_vdc3_init_video_out(void)
215  {
216      /* ---- 映像表示モードの設定( NTSC, BT.656 ) ---- */
217      VDC3.VIDEO_MODE.BIT.BURST_MODE_DISP = 0; /* 映像供給部のパス:16byte バースト転送 */
218      VDC3.VIDEO_MODE.BIT.BURST_MODE_MAIN = 0; /* 映像受信部のパス:16byte バースト転送 */
219      VDC3.VIDEO_MODE.BIT.ENDIAN_DISP = 0; /* 映像供給部のパス:ビッグエンディアン */
220      VDC3.VIDEO_MODE.BIT.ENDIAN_MAIN = 0; /* 映像受信部のパス:ビッグエンディアン */
221      VDC3.VIDEO_MODE.BIT.VIDEO_MODE = 1; /* 映像表示機能を選択 */
222      VDC3.VIDEO_INT_CNT.LONG = 0x00000000ul; /* 映像割り込み不許可 */
223      VDC3.VIDEO_SIZE.BIT.VIDEO_HEIGHT = VIN_INPUT_HEIGHT;
224      VDC3.VIDEO_SIZE.BIT.VIDEO_WIDTH = VIN_INPUT_WIDTH;
225                                     /* 入力有効映像のライン数とピクセル数を設定 */
226      VDC3.VIDEO_VSTART.BIT.VIDEO_VSTART_TOP = VIN_VSTART_VALIDDATA_TOP;
227      VDC3.VIDEO_VSTART.BIT.VIDEO_VSTART_BTM = VIN_VSTART_VALIDDATA_BTM;
228      VDC3.VIDEO_HSTART.BIT.VIDEO_HSTART = VIN_HSTART_VALIDDATA;
229                                     /* 垂直方向の入力有効映像開始位置(TOP)
230                                     垂直方向の入力有効映像開始位置(BOTTOM)
231                                     水平方向の入力有効映像開始位置 */
232      VDC3.VIDEO_VSYNC_TIM1.BIT.VIDEO_VSYNC_START1_TOP
233      = VIN_VSTART_VALIDDATA_TOP+VOUT_IO_DFLINE-TFT_DE_START_V+(VOUT_BUF_MARGIN/2)
-VOUT_DISP_POS_IO_DF;
234      VDC3.VIDEO_VSYNC_TIM1.BIT.VIDEO_VSYNC_START1_BTM
235      = VIN_VSTART_VALIDDATA_BTM+VOUT_IO_DFLINE-TFT_DE_START_V+(VOUT_BUF_MARGIN/2)
-VOUT_DISP_POS_IO_DF;
236                                     /* TOP フィールドの基準 Vsync 位置と
237                                     BOTTOM フィールドの基準 Vsync 位置を設定 */
238      VDC3.VIDEO_SAVE_NUM.BIT.FIELD_SAVE_NUM = 0; /* 保存フィールド数(映像表示機能では 0 を設定) */
239      VDC3.VIDEO_IMAGE_CNT.LONG = 0x80800311ul; /* 輝度コントラスト調整なし */
240                                     /* 輝度ブライツ調整なし */
241                                     /* 輝度クリップ有効 */
242                                     /* 色差クリップ有効 */
243                                     /* 垂直方向スケーリング:1/3 */
244                                     /* 水平方向スケーリング:1/3 */

```

## 3.10 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (7)

```
245     VDC3.VIDEO_BASEADR.LONG          = (unsigned long)video_line_buffer;
246     VDC3.VIDEO_LINE_OFFSET.LONG      = VOUT_LINE_OFFSET;
247     VDC3.VIDEO_FIELD_OFFSET.LONG     = 0;          /* フィールドオフセット(映像表示機能では無効) */
248     VDC3.VIDEO_LINEBUFF_NUM.BIT.VIDEO_LINEBUFF_NUM = VOUT_LINEBUF_NUM;
249                                     /* ラインバッファのアドレス、
250                                     ラインオフセット(ライン当りのバイト数)、
251                                     使用ライン数 を設定 */
252     VDC3.VIDEO_DISP_SIZE.BIT.VIDEO_DISP_HEIGHT = VOUT_DISP_SZ_Y;
253     VDC3.VIDEO_DISP_SIZE.BIT.VIDEO_DISP_WIDTH  = VOUT_DISP_SZ_X;
254     VDC3.VIDEO_DISP_HSTART.BIT.VIDEO_DISP_HSTART = (TFT_DE_START_H-16);
255                                     /* 表示する映像データの垂直/水平方向サイズ、
256                                     水平方向開始位置 を設定 */
257     VDC3.SGMODE.BIT.EX_SYNC_MODE= 1;          /* 基準 Vsync を、映像入力 Vsync に同期 */
258     VDC3.VIDEO_VSYNC_TIM2.LONG        = (TFT_DE_START_V + VOUT_DISP_POS_Y - 1);
259                                     /* 映像表示用 Vsync のタイミングを設定 */
260 }
261
```

## 3.11 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (8)

```

262  /*****
263  * ID      :
264  * Outline  : グラフィックス部 1 の初期化
265  * Include  : iodefine.h
266  * Declaration : static void io_vdc3_init_grphcs1(void);
267  * Description : 本関数はグラフィックス部 1 の初期化処理を行います。
268  * Argument  : void
269  * Return Value : void
270  *****/
271  static void io_vdc3_init_grphcs1(void)
272  {
273      /* ---- グラフィックス表示開始の設定 ---- */
274      VDC3.GRA_VSYNC_TIM.LONG = 0;          /* 基準 VSYNC の 0 ライン目にグラフィックス表示 */
275      VDC3.GRCBUSCNT1.LONG = 0;           /* バス制御設定 */
276                                          /* 16Byte バースト転送 */
277                                          /* RGB565 形式 */
278                                          /* Iバス: ビッグエンディアン */
279      VDC3.GROPSADR1.BIT.GROPSADR = (0xFFFFFFFFul & (unsigned long)grph_buffer1[0]);
280                                          /* 画像データの読み出しアドレス */
281      VDC3.GROPSWH1.BIT.GROPSH = GRPHCS1_Y_SIZE; /* グラフィックス画像領域の高さ */
282      VDC3.GROPSWH1.BIT.GROPSW = GRPHCS1_X_SIZE; /* グラフィックス画像領域の幅 */
283      VDC3.GROPSOFST1.BIT.GROPSOFST = GRPHCS1_LINE_OFFSET;
284                                          /* グラフィックス画像ラインオフセット(バイト数) */
285      VDC3.GROPDPHV1.BIT.GROPDPV = GRPHCS1_POS_Y + TFT_DE_START_V - 1;
286                                          /* 垂直表示開始位置 */
287      VDC3.GROPDPHV1.BIT.GROPDPH = GRPHCS1_POS_X + TFT_DE_START_H -16;
288                                          /* 水平表示開始位置 */
289      VDC3.GROPBASERGB1.LONG = 0x00000000ul; /* 領域外の色指定(黒) */
290
291      /* ----  $\alpha$  制御領域表示の設定 ---- */
292      VDC3.GROPEWH1.BIT.GROPEH = GRPHCS1_ALPHA_Y_SIZE; /*  $\alpha$  制御領域の高さ */
293      VDC3.GROPEWH1.BIT.GROPEW = GRPHCS1_ALPHA_X_SIZE; /*  $\alpha$  制御領域の幅 */
294      VDC3.GROPEDPHV1.BIT.GROPEDPV = GRPHCS1_ALPHA_POS_Y + TFT_DE_START_V - 1; /* 垂直開始位置 */
295      VDC3.GROPEDPHV1.BIT.GROPEDPH = GRPHCS1_ALPHA_POS_X + TFT_DE_START_H -16; /* 水平開始位置 */
296      VDC3.GROPEDPAL.BIT.DEFA = 255; /*  $\alpha$  値の初期値 */
297      VDC3.GROPEDPAL.BIT.ACOEF = 1; /*  $\alpha$  値演算係数 */
298      VDC3.GROPEDPAL.BIT.ARATE = 0; /* 1 フレーム毎に  $\alpha$  値演算 */
299      VDC3.GROPEDPAL.BIT.WE = 1; /*  $\alpha$  値の転送許可 */
300      VDC3.GROPEDPAL.BIT.AMOD = 2; /*  $\alpha$  値加算処理 */
301      VDC3.GROPEDPAL.BIT.AEN = 1; /*  $\alpha$  値制御許可 */
302  }
303

```

## 3.12 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (9)

```

304  /*****
305  * ID      :
306  * Outline : グラフィックス部 2 の初期化
307  * Include : iodef.h
308  * Declaration : static void io_vdc3_init_grphcs2(void);
309  * Description : 本関数はグラフィックス部 2 の初期化処理を行います。
310  * Argument  : void
311  * Return Value : void
312  *****/
313  static void io_vdc3_init_grphcs2(void)
314  {
315      /* ---- グラフィックス表示開始の設定 ---- */
316      VDC3.GRA_VSYNC_TIM.LONG = 0;          /* 基準 VSYNC の 0 ライン目にグラフィックス表示 */
317      VDC3.GRCBUSCNT2.LONG = 0;            /* バス制御設定 */
318                                          /* 16Byte バースト転送 */
319                                          /* RGB565 形式 */
320                                          /* Iバス : ビッグエンディアン */
321      VDC3.GROPSADR2.BIT.GROPSADR = (0x1FFFFFFul & (unsigned long)grph_buffer2[0]);
322                                          /* 画像データの読み出しアドレス */
323      VDC3.GROPSWH2.BIT.GROPSH = GRPHCS2_Y_SIZE; /* グラフィックス画像領域の高さ */
324      VDC3.GROPSWH2.BIT.GROPSW = GRPHCS2_X_SIZE; /* グラフィックス画像領域の幅 */
325      VDC3.GROPSOFST2.BIT.GROPSOFST = GRPHCS2_LINE_OFFSET;
326                                          /* グラフィックス画像ラインオフセット(バイト数) */
327      VDC3.GRODPHV2.BIT.GRODPV = GRPHCS2_POS_Y + TFT_DE_START_V - 1;
328                                          /* 垂直表示開始位置 */
329      VDC3.GRODPHV2.BIT.GRODPH = GRPHCS2_POS_X + TFT_DE_START_H -16;
330                                          /* 水平表示開始位置 */
331      VDC3.GROPBASERGB2.LONG = 0x00000000ul; /* 領域外の色指定(黒) */
332
333      /* ----  $\alpha$  制御領域表示の設定 ---- */
334      VDC3.GROPEWH2.BIT.GROPEH = GRPHCS2_ALPHA_Y_SIZE; /*  $\alpha$  制御領域の高さ */
335      VDC3.GROPEWH2.BIT.GROPEW = GRPHCS2_ALPHA_X_SIZE; /*  $\alpha$  制御領域の幅 */
336      VDC3.GROPEDPHV2.BIT.GROPEDPV = GRPHCS2_ALPHA_POS_Y + TFT_DE_START_V - 1; /* 垂直開始位置 */
337      VDC3.GROPEDPHV2.BIT.GROPEDPH = GRPHCS2_ALPHA_POS_X + TFT_DE_START_H -16; /* 水平開始位置 */
338      VDC3.GROPEDPA2.BIT.DEFA = 0; /*  $\alpha$  値の初期値 */
339      VDC3.GROPEDPA2.BIT.ACOEF = 1; /*  $\alpha$  値演算係数 */
340      VDC3.GROPEDPA2.BIT.ARATE = 0; /* 1 フレーム毎に  $\alpha$  値演算 */
341      VDC3.GROPEDPA2.BIT.WE = 1; /*  $\alpha$  値の転送許可 */
342      VDC3.GROPEDPA2.BIT.AMOD = 1; /*  $\alpha$  値加算処理 */
343      VDC3.GROPEDPA2.BIT.AEN = 1; /*  $\alpha$  値制御許可 */
344  }

```

## 3.13 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.c" (10)

```
345  /*****
346  * ID      :
347  * Outline  : パネル制御部、出力タイミング制御部の初期化
348  * Include  : iodef.h
349  * Declaration : static void io_vdc3_init_disp(void);
350  * Description : 本関数はパネル制御部と出力タイミング制御部の初期化処理をします。
351  * Argument  : void
352  * Return Value : void
353  *****/
354  static void io_vdc3_init_disp(void)
355  {
356  /* ---- パネル向け制御信号出力の設定 ---- */
357  VDC3.SYNCNT.LONG = 0x0000000Cu1; /* 全信号を立ち上がり同期で出力 */
358  /* LCD_VSYNC/LCD_HSYNC 信号 : 反転出力 */
359  VDC3.PANEL_CLKSEL.BIT.ICKEN = 0; /* パネルクロックブロックの動作禁止 */
360  VDC3.PANEL_CLKSEL.LONG = 0x0000200Cu1; /* クロックソース : B $\phi$  (72MHz) */
361  /* クロック周波数 : 6MHz */
362  VDC3.PANEL_CLKSEL.BIT.ICKEN = 1; /* パネルクロックブロックの動作許可 */
363  VDC3.SYN_SIZE.BIT.SYN_HEIGHT= TFT_TOTAL_SZ_V; /* 帰線期間を含めた垂直方向ライン数 */
364  VDC3.SYN_SIZE.BIT.SYN_WIDTH = TFT_TOTAL_SZ_H; /* 帰線期間を含めた水平方向ピクセル数 */
365  VDC3.PANEL_VSYNC_TIM.LONG = TFT_VSYNC_WDTH; /* パネル出力用 VSYNC のタイミング設定 */
366  VDC3.PANEL_HSYNC_TIM.LONG = TFT_HSYNC_WDTH; /* パネル出力用 HSYNC のタイミング設定 */
367  VDC3.DE_SIZE.BIT.DE_HEIGHT = TFT_DISP_SZ_V; /* データイネーブル領域の高さ */
368  VDC3.DE_SIZE.BIT.DE_WIDTH = TFT_DISP_SZ_H; /* データイネーブル領域の幅 */
369  VDC3.DE_START.BIT.DE_START_V= TFT_DE_START_V; /* 垂直方向のデータイネーブル領域の開始位置 */
370  VDC3.DE_START.BIT.DE_START_H= TFT_DE_START_H; /* 水平方向のデータイネーブル領域の開始位置 */
371  }
372
```

## 3.14 サンプルプログラムリスト "io\_vdc3\_alpha\_blend.c" (11)

```
373  /*****
374  * ID      :
375  * Outline : VDC3 の動作開始
376  * Include : iodef.h
377  * Declaration : static void io_vdc3_start(void);
378  * Description : 本関数は VDC3 の動作開始設定を行います。
379  *             : VDC3 の各種レジスタを設定した後に本関数を実行します。
380  *             : レジスタ設定は次の Vsync 同期にて更新されます。
381  * Argument : void
382  * Return Value : void
383  *****/
384  static void io_vdc3_start(void)
385  {
386      /* ---- 映像受信部の動作許可 ---- */
387      VDC3.VIDEO_MODE.BIT.VIDEO_MAIN_EXE = 1;
388
389      /* ---- 映像供給部の動作許可 ---- */
390      VDC3.VIDEO_MODE.BIT.VIDEO_DISP_EXE = 1;
391
392      /* ---- グラフィックス部 2 の表示許可 ---- */
393      VDC3.GRCMEN2.LONG = 0x80000003ul; /* カレントレイヤ:許可、下位レイヤ:許可 */
394
395      /* ---- グラフィックス部 1 の表示許可 ---- */
396      VDC3.GRCMEN1.LONG = 0x80000003ul; /* カレントレイヤ:許可、下位レイヤ:許可 */
397
398  }
399  /* End of File */
400
```

## 3.15 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.h" (1)

```
1  /*****
2  *  DISCLAIMER
3  *
4  *  This software is supplied by Renesas Electronics Corporation and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics Corporation and is protected under
8  *  all applicable laws, including copyright laws.
9  *
10 *  THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *  REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *  INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *  PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *  DISCLAIMED.
15 *
16 *  TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *  ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *  FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *  FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *  AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *  Renesas reserves the right, without notice, to make changes to this
23 *  software and to discontinue the availability of this software.
24 *  By using this software, you agree to the additional terms and
25 *  conditions found by accessing the following link:
26 *  http://www.renesas.com/disclaimer
27 *****/
28 /* Copyright (C) 2011 Renesas Electronics Corporation. All Rights Reserved.*/
29 /*****
30 *  System Name : SH7264 Sample Program
31 *  File Name   : io_vdc3_alpha_blend.h
32 *  Abstract    : VDC3  $\alpha$ ブレンド窓機能の使用例
33 *  Version     : 1.00.00
34 *  Device      : SH7264
35 *  Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *              : C/C++ compiler package for the SuperH RISC engine family
37 *              :                      (Ver.9.03 Release00).
38 *  OS          : None
39 *  H/W Platform: M3A-HS64G50(CPU board), M3A-HS64G02(Option board)
40 *  Description :
41 *****/
42 *  History     : Feb.28,2011 Ver.1.00.00 VDC3の既存サンプルコードを統合して作成
43 *****/
44
45
```

## 3.16 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.h" (2)

```

46  /*****
47  Macro definitions
48  *****/
49  #define BYTES_PER_PIXEL      2      /* 1 ピクセル当たりのバイト数 */
50  #define RGB565_BLACK        0x0000u /* 黒 */
51  #define RGB565_WHITE        0xFFFFu /* 白 */
52  #define RGB565_GREEN        0x07E0u /* 緑 */
53  #define RGB565_BLUE         0x001Fu /* 青 */
54
55  /* ---- 映像入力パラメータ ---- */
56  #define VIN_VSTART_VALIDDATA_TOP 16 /* TOP 領域の垂直方向キャプチャタイミング */
57  #define VIN_VSTART_VALIDDATA_BTM 279 /* BOTTOM 領域の垂直方向キャプチャタイミング */
58  #define VIN_HSTART_VALIDDATA     276 /* 水平方向のキャプチャタイミング */
59  #define VIN_INPUT_HEIGHT         240 /* 入力有効映像のライン数 */
60  #define VIN_INPUT_WIDTH          720 /* 入力有効映像のピクセル数 */
61
62  /* ---- 映像表示パラメータ ---- */
63  #define VOUT_DISP_SZ_Y          160 /* 映像表示領域の高さ */
64  #define VOUT_DISP_SZ_X          240 /* 映像表示領域の幅 */
65  #define VOUT_BUF_MARGIN         6 /* ラインバッファのマージン(6以上) */
66  #define VOUT_IO_DFLINE          127 /* 入力周期と表示周期の差分ライン数 */
67                                     /* = VIN_INPUT_HEIGHT - (VOUT_DISP_SZ_Y * (0.045/0.064)) */
68  #define VOUT_LINEBUF_NUM        (VOUT_IO_DFLINE + VOUT_BUF_MARGIN) /* ラインバッファのライン数 */
69  #define VOUT_LINE_OFFSET        (((VOUT_DISP_SZ_X * BYTES_PER_PIXEL) + 15) & 0xFFFFFFF0ul)
70                                     /* 1 ライン当りのバイト数 */
71  #define VOUT_DISP_POS_Y         80 /* 垂直方向の映像表示開始位置(パネル最上部から) */
72  #define VOUT_DISP_POS_X         0 /* 水平方向の映像表示開始位置(パネル最左部から) */
73  #define VOUT_DISP_POS_IO_DF     56 /* VOUT_DISP_POS_Y を入力映像のライン数に換算した値 */
74                                     /* = VOUT_DISP_POS_Y * (0.045/0.064) */
75
76  /* ---- グラフィックス画像パラメータ ---- */
77  #define GRPHCS1_Y_SIZE          80 /* グラフィックス画像 1 の高さ */
78  #define GRPHCS1_X_SIZE          80 /* グラフィックス画像 1 の幅 */
79  #define GRPHCS1_LINE_OFFSET     (((GRPHCS1_X_SIZE * BYTES_PER_PIXEL) + 15) & 0xFFFFFFF0ul)
80                                     /* グラフィックス画像 1 のラインオフセット(バイト数) */
81                                     /* 128 バイトまたは 16 バイト境界に配置 */
82  #define GRPHCS1_POS_Y           100 /* 垂直方向の表示開始位置 (0:最上部) */
83  #define GRPHCS1_POS_X           60 /* 水平方向の表示開始位置 (0:最左部) */
84  #define GRPHCS1_ALPHA_Y_SIZE    320 /*  $\alpha$  制御領域の高さ */
85  #define GRPHCS1_ALPHA_X_SIZE    240 /*  $\alpha$  制御領域の幅 */
86  #define GRPHCS1_ALPHA_POS_Y     0 /*  $\alpha$  制御領域の垂直開始位置 */
87  #define GRPHCS1_ALPHA_POS_X     0 /*  $\alpha$  制御領域の水平開始位置 */
88

```

## 3.17 サンプルプログラムリスト"io\_vdc3\_alpha\_blend.h" (3)

```

89  #define GRPHCS2_Y_SIZE      80      /* グラフィックス画像 2 の高さ */
90  #define GRPHCS2_X_SIZE      80      /* グラフィックス画像 2 の幅 */
91  #define GRPHCS2_LINE_OFFSET ((GRPHCS2_X_SIZE * BYTES_PER_PIXEL) + 15 ) & 0xFFFFFFFF0ul)
92                                /* グラフィックス画像 2 のラインオフセット(バイト数) */
93                                /* 128 バイトまたは 16 バイト境界に配置 */
94  #define GRPHCS2_POS_Y      140     /* 垂直方向の表示開始位置 (0:最上部) */
95  #define GRPHCS2_POS_X      100     /* 水平方向の表示開始位置 (0:最左部) */
96  #define GRPHCS2_ALPHA_Y_SIZE 320   /*  $\alpha$  制御領域の高さ */
97  #define GRPHCS2_ALPHA_X_SIZE 240   /*  $\alpha$  制御領域の幅 */
98  #define GRPHCS2_ALPHA_POS_Y 0      /*  $\alpha$  制御領域の垂直開始位置 */
99  #define GRPHCS2_ALPHA_POS_X 0      /*  $\alpha$  制御領域の水平開始位置 */
100
101 /* ---- TFT 液晶表示モジュールパラメータ ---- */
102 #define TFT_TOTAL_SZ_V      327     /* 帰線期間を含めた垂直方向ライン数 */
103 #define TFT_TOTAL_SZ_H      273     /* 帰線期間を含めた水平方向ピクセル数 */
104 #define TFT_DISP_SZ_V      320     /* 垂直ディスプレイ有効区間 */
105 #define TFT_DISP_SZ_H      240     /* 水平ディスプレイ有効区間 */
106 #define TFT_VSYNC_WIDTH    1       /* LCD_VSYNC のパルス幅(ライン) */
107 #define TFT_HSYNC_WIDTH    5       /* LCD_HSYNC のパルス幅(ピクセル) */
108 #define TFT_DE_START_V     1       /* 基準 Vsync から有効区間までのライン数 */
109 #define TFT_DE_START_H     (TFT_TOTAL_SZ_H - 251)
110                                /* 基準 Hsync から有効区間までのピクセル数 */
111
112 /******
113 Imported global variables and functions (from other files)
114 *****/
115 /* ==== Global functions ==== */
116 extern void io_vdc3_init(void);
117 extern void io_vdc3_change_buffer( int grphcs_no, unsigned short *buffer );
118
119 /* ==== Global variables ==== */
120 extern unsigned short grph_buffer1[2][GRPHCS1_Y_SIZE][GRPHCS1_LINE_OFFSET /
BYTES_PER_PIXEL];
121 extern unsigned short grph_buffer2[2][GRPHCS2_Y_SIZE][GRPHCS2_LINE_OFFSET /
BYTES_PER_PIXEL];
122 extern unsigned short video_line_buffer[VOUT_LINEBUF_NUM][VOUT_LINE_OFFSET /
BYTES_PER_PIXEL];
123
124 /* End of File */
125

```

#### 4. 参考ドキュメント

- ソフトウェアマニュアル  
SH-2A、SH2A-FPU ソフトウェアマニュアル Rev.3.00  
(最新版をルネサスエレクトロニクスのホームページから入手してください。)
- ハードウェアマニュアル  
SH7262 グループ、SH7264 グループ ハードウェアマニュアル Rev.2.00  
(最新版をルネサスエレクトロニクスのホームページから入手してください。)

## ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com/>

お問い合わせ先

<http://japan.renesas.com/inquiry>

すべての商標および登録商標は、それぞれの所有者に帰属します。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2009.04.08	—	初版発行
1.01	2009.05.19	8	<ul style="list-style-type: none"> <li>・ 図 2 修正</li> <li>・ 誤記修正</li> </ul>
1.02	2010.01.19	16	図 9 注意事項追加
		25	参考プログラムについての補足を追加 (640KB 版に対応)
		31	PGCR7 への書き込み方法を修正
		37、39	$\alpha$ ブレンド開始位置の計算式を修正
		26~39	ヘッダコメントのフォーマット変更
1.03	2011.03.23	15~41	ソースコードの構成を変更

## 製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

### 1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

### 2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

### 3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

### 4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

### 5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違っていると、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連して発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続きを行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。  
注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。  
総合お問合せ窓口：<http://japan.renesas.com/inquiry>