



XAPP928 (v1.1) 2007 年 4 月 19 日

# LVDS/DVI を使用するデジタル ディスプレイ パネル IP のリファレンス デザイン

Spartan-3E ディスプレイ開発キット HW-SPAR3E-DISP-DK-UNI-G 向け

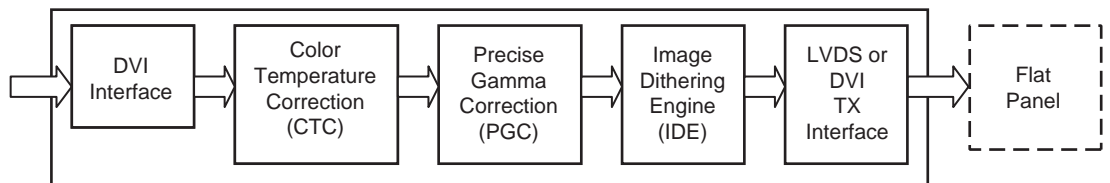
本資料は英語版 (v1.1) を翻訳したものです。英語の更新バージョンがリリースされている場合には、最新の英語版を必ずご参照ください。

## はじめに

デジタル ディスプレイは LCD、プラズマ、リアプロジェクション TV テクノロジーや携帯ハンドセットおよびオートモーティブ向けの小型ディスプレイを含む、多数のアプリケーションで構成され、急成長を遂げている市場です。デジタル画像処理によって表示される画像の全体的な画質を向上させ、製品の差別化を図ることができます。

ザイリンクスでは、この市場向けの製品開発をサポートするため、ディスプレイ パネル アプリケーション用のリファレンス デザイン IP コアを、Spartan™-3E ディスプレイ開発ボードに基づいて開発しました。FPGA のディスプレイ ソリューションは、[図 1](#) に示すように、DVI 入力インターフェイス、色温度調整 (CTC)、正確なガンマ補正 (PGC)、イメージ ディザリング エンジン (IDE)、および低電圧差動信号送信 (LVDS TX) または DVI TX 出力インターフェイスで構成されます。

このアプリケーション ノートでは、Spartan-3E ディスプレイ開発ボードについて説明します。また、DIP スイッチ設定について詳細に説明し、各 IP ブロックのリソース数も記載しています。



X928\_01\_040907

図 1: ザイリンクス ディスプレイ パネル IP リファレンス デザインのフロー

## IP ブロックの サマリ

表 1 で、Spartan-3E ディスプレイ開発ボードの主要な IP ブロックについて説明します。

表 1: IP ブロックの概要

ブロック名	説明
DVI レシーバ	DVI 形式の入力ビデオを受信
CTC (色温度調整)	フレーム全体の入力 RGB 値をユーザーが設定した色温度に調整
PGC (正確なガンマ補正)	ピクセルのストリームを受け取り、必要なガンマ値に処理した後、修正後のピクセル ストリームを次のモジュールに送信
IDE (イメージ ディザリング エンジン)	PGC モジュールからのガンマ補正されたピクセル ストリームを 3 x 10 ビットで受け取って、ビデオ画質を低下させることなく、3 x 8 ビットのピクセル ストリームにディザリングする
ディスプレイ インターフェイス	LVDS インターフェイスを介して DVI トランスミッタが LCD モジュールを直接駆動できるようにする

© 2007 Xilinx, Inc. All Rights Reserved. XILINX, Xilinx ロゴ、およびその他本文に含まれる商標名は Xilinx の商標です。本文書に記載されている「Xilinx」、ザイリンクスのロゴ、およびザイリンクスが所有する製品名等は、米国 Xilinx Inc. の米国における登録商標です。その他に記載されている会社名および製品名等は、各社の商標または登録商標です。保証否認の通知: Xilinx ではデザイン、コード、その他の情報を「現状有姿の状態」で提供しています。この特徴、アプリケーションまたは規格の一実施例としてデザイン、コード、その他の情報を提供しておりますが、Xilinx はこの実施例が権利侵害のクレームを全く受けないということを表明するものではありません。お客様がご自分で実装される場合には、必要な権利の許諾を受ける責任があります。Xilinx は、実装の妥当性に関するいかなる保証を行なうものではありません。この保証否認の対象となる保証には、権利侵害のクレームを受けないことの保証または表明、および市場性に対する適合性についての黙示的な保証も含まれます。

## ユーザー スイッチおよびプッシュ ボタンの設定

本書でのスイッチ設定は、Spartan-3E ディスプレイ開発ボード上でユーザー設定の DIP スイッチ ブロックで行います (図 2)。

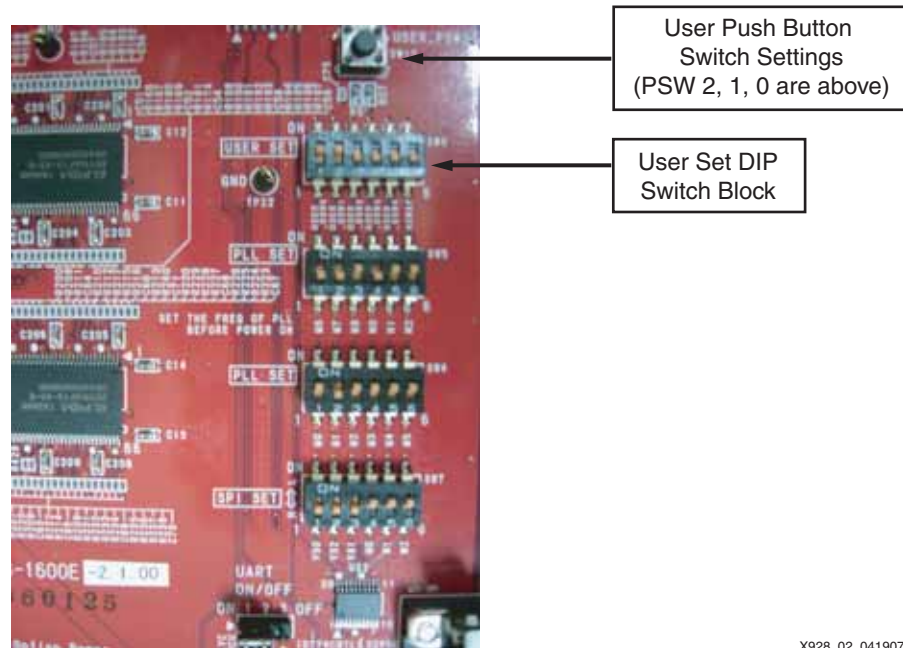


図 2: ディスプレイ開発ボードのスイッチ設定

表 2: DIP スイッチのユーザー設定

DIP スイッチ	機能
SW4	DVI RX。常に「ON」にする。
SW3、SW2	PGC (正確なガンマ補正) の制御 <ul style="list-style-type: none"> <li>On、On (00): ガンマ 2.2/2.2 (1.00)</li> <li>On、Off (01): ガンマ 2.4/2.2 (1.09)</li> <li>Off、On (10): ガンマ 2.6/2.2 (1.18)</li> <li>On、On (11): PGC 無効</li> </ul>
SW1、SW0	CTC (色温度調整) の制御 <ul style="list-style-type: none"> <li>On、On (00): バイパス</li> <li>On、Off (01): 6500K</li> <li>Off、On (10): 8500K</li> <li>On、On (11): 10000K</li> </ul>

イメージ デザリング エンジン は、プッシュ ボタンの設定で調整されます。

- 0: プッシュ ボタンを押す場合に使用
- 1: プッシュ ボタンを放す場合に使用

### DVI 受信 (RX) インターフェイス ブロック

このリファレンス デザインでは、DVI RX インターフェイスを使用する必要があります。DVI オプションの設定については、『ディスプレイ キット ユーザー ガイド』を参照してください。表 3 に、DVI RX インターフェイスを使用する際の DIP スイッチの設定を示します。

表 3 : LVDS/DVI RX を使用する場合の LVDS DIP スイッチ設定

DIP スイッチ	ポート名	設定
DIP_SWITCH_0 [ユーザー スイッチ SW4]	DIPS_RX_SELECT	常に「ON」にする。

## CTC IP ブロック

### 色温度について

白色光は色温度で表すことができます。ある光源の色温度を特定するには、その出力をある温度で理論的な黒体が放射する光と比較します(単位はケルビン、K)。具体的には、5000K ~ 5500K が通常の昼白色、2000K が赤/オレンジ色、15000K は青みがかった色となります。たとえば、空の太陽光が赤、オレンジ、白、青などのように位置によって異なって見えるように、色温度は、光源およびディスプレイテクノロジーによって異なります。

デジタルディスプレイにおける白色は、そのディスプレイの R、G、および B セルから出されるこれらの色の組み合わせで認識されます。RGB データは CIE x-y 色空間に合うように変換でき、色空間では、変換アルゴリズムで多数の演算が実行されます。

「ホワイトポイント」は色温度として大まかに定義されています。CIE 色度図(図 3)では、5500K でのホワイトポイントは、ほぼ  $x=y=0.33$  の点です。

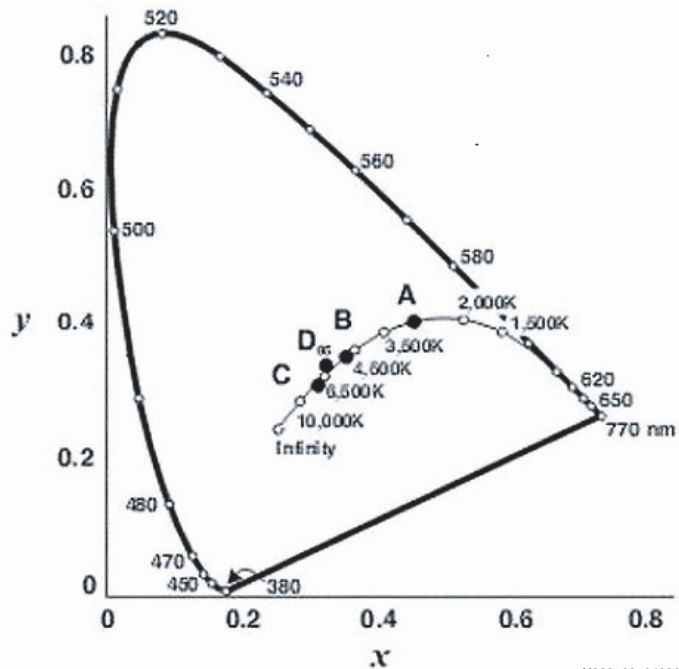


図 3 : CIE 色度図に示す色温度

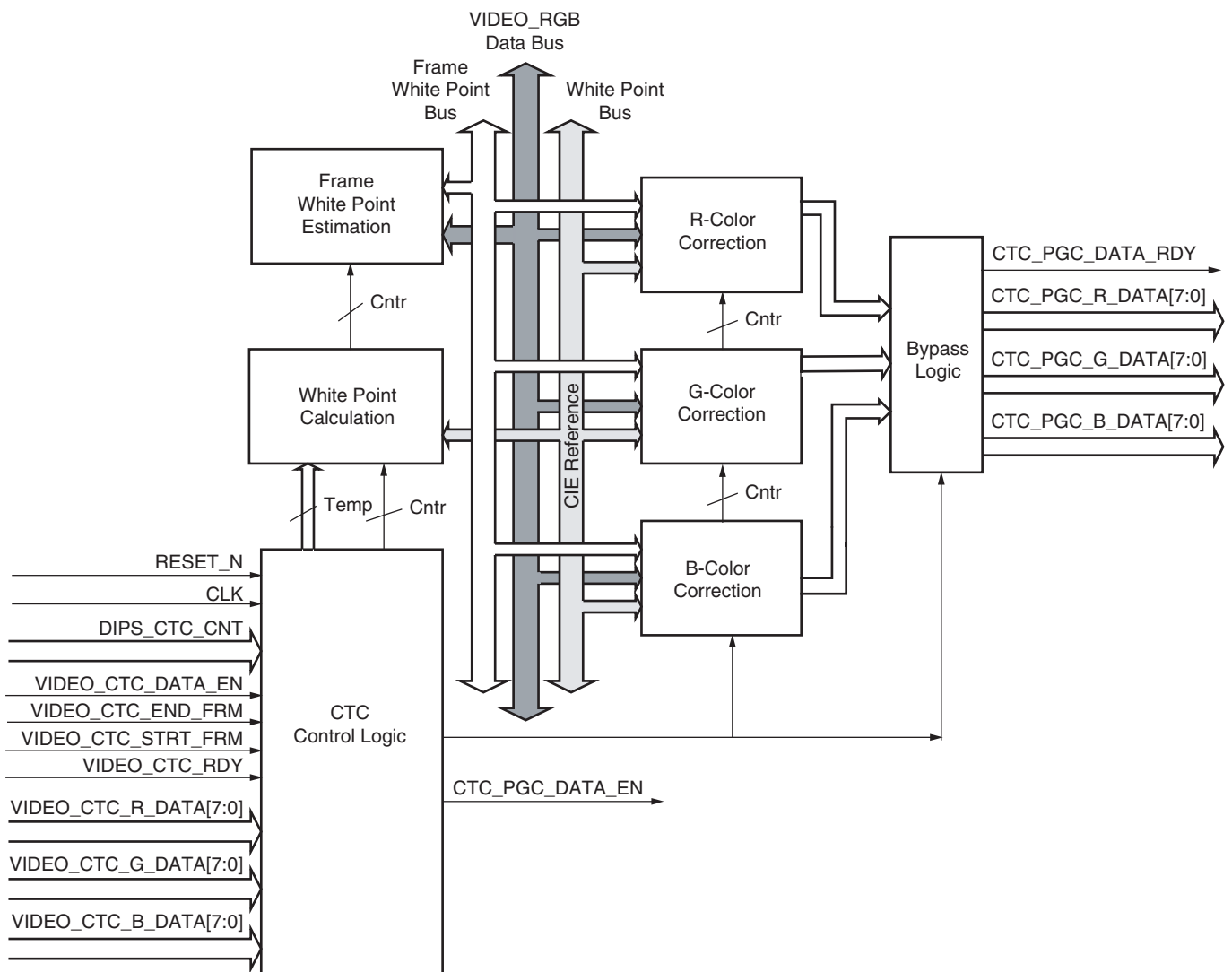
この図を見ると、PDP (プラズマディスプレイパネル) の色温度が低くなっており、特に CRT (ブラウン管) ディスプレイと比較するとこれは顕著です。PDP では基本的に青輝度が低いため、色温度も低くなります。さらに、PDP のセル構造および関連する動作回路がいったん固定されると、エンドカスタマが色温度を独自に変更することはできません。したがって、PDP などのデジタルディスプレイ出力は、より高品質の画像を実現するように調整される必要があります。

### CTC アルゴリズム

このアルゴリズムのインプリメンテーションでは、フレームのホワイトポイントまたは色温度が特定のフレームに求められる色温度に変更されます。最初に、CIEの基準ホワイトポイントが、ユーザーが設定した温度入力からxおよびy形式で選択されます。入力される全フレームに対するホワイトポイントが概算されます。8または10ビットのRGBデータの入力フレームはcolor-mixingプログラムを使用してx-y形式に変換されます。

次に、入力フレーム温度および基準温度が同じかどうかを確認されます。再帰演算がデータ温度値を制御し、正確な色温度調整を行うため、基準となるホワイトポイントが選択されます。色温度が同一(差分内)の場合は変更されず、色温度が異なると、フレームは適切な色温度に変更されます。

色温度の調整には、比率法が使用されます。CIE温度対入力フレームの温度比率は、入力フレームのRGBデータそれぞれに適応され、適切な色温度を設定します。この時点で、修正された色温度の変換が完了したことになります。



X928\_04\_041907

図 4 : CTC IP ブロック図

表 4 に、CTC モジュールで使用される信号の説明を示します。

表 4 : CTC モジュールの信号

信号	I/O	説明
RESET_N	入力	システム リセット。CTC モジュール内の全プロセスをリセットするには、この信号を Low にアサートする。
CLK	入力	システム クロック。ビデオ処理に必要な周波数で動作。
DIPS_CTC_CNT	入力	DIP スイッチからの制御信号 (ユーザー スイッチ SW1、SW0)。ターゲットとする色温度を決定。
VIDEO_CTC_STRT_FRM	入力	CTC はこの信号をアサートして、LVDS レシーバへ送信する新しいフレームの開始を示す。
VIDEO_CTC_END_FRM	入力	CTC はこの信号をアサートして、LVDS レシーバへ送信している現在のフレームの終了を示す。
VIDEO_CTC_DATA_EN	入力	CTC はこの信号をアサートして、有効なデータが LVDS レシーバ ブロックに与えられたことを示す。
VIDEO_CTC_RGB_DATA[23:0]	入力	CTC では RGB 形式の入力ピクセル データが、この 24 ビットのデータ バスから入力される。
CTC_PGC_DATA_RDY	出力	CTC はこの信号をアサートして、有効なデータが次の PGC ブロックでプロセスできる状態であることを示す。
CTC_PGC_DATA_EN	出力	この信号がアサートされると、PGC レシーバ ブロックの CTC_PGC_RGB_DATA[23:0] のデータが有効となる。
CTC_PGC_RGB_DATA[23:0]	出力	RGB 形式の出力ピクセル データが、この 24 ビットのデータ バスから PGC に出力される。

色温度の入力レジスタ値は、ターゲットとする色温度として標準温度値 (6500K、8000K、9300K) を示します。これらの値は外部の DIP スイッチで選択されます。そして、CIE の基準ホワイト ポイント値が決定されます。この後の処理では、ホワイト ポイントに対する RGB 値が使用されます。デフォルトでは、色温度は変更されず、入力データをバイパスします。次のフレームの開始時点で色温度を選択した後、CTC モジュールでは、基準点となるホワイト ポイントの値が更新されます。

#### フレームのホワイト ポイントの概算

ホワイト ポイントを概算するには、次のアルゴリズムを使用します。まず、RGB の和が最大となるピクセルをフレーム全体から見つけます。この和を示す RGB 値が、そのフレームのホワイト ポイントとみなされます。そして、このホワイト ポイントは次の入力フレームの開始で再び更新されます。

#### CTC (色温度調整) モジュール

CTC モジュールでは、フレーム全体の入力 RGB 値が新しい温度に調整されます。このモジュールでは、一定の比率を乗算する方法によって、値が調整されます。たとえば、ピクセルの R の調整後の値は等式 1 から求められます。

$$R_{in} \times \frac{R_{wt}}{R_{wf}} = R_{out} \quad \text{等式 1}$$

説明：

- $R_{in}$  = 入力ピクセルの R 値
- $R_{wt}$  = 標準温度の CIE ホワイト ポイント値
- $R_{wf}$  = 前のフレームのホワイト ポイントにおける R 値
- $R_{out}$  = CTC モジュールで調整された出力ピクセルの R 値

$R_{wt}$ 、 $R_{wf}$  比率はフレーム全体の間隔で一定です。G および B 値に対しても同じ方法で、調整後の値が求められます。ピクセルはクロック サイクルごとに同時に乗算されます。

CTC モジュールが次のモジュール、PGC (正確なガンマ補正) ブロックにデータを送信できる状態になると、CTC\_PGC\_DATA\_RDY フラグがアサートされます。

## DIP スイッチの設定

表 5 に、色温度を調整する際の DIP スイッチの設定を示します。

表 5: CTC モジュールの DIP スイッチ設定

DIP スイッチ	ポート名	設定
DIP_SWITCH_TYPE0 DIP_SWITCH_TYPE1	DIPS_CTC_CNT0_IP DIPS_CTC_CNT1_IP	CTC_TYPE1 & CTC_TYPE0 [スイッチ 1、スイッチ 0] <ul style="list-style-type: none"> <li>• 11 : バイパス</li> <li>• 10 : 6500K</li> <li>• 01 : 8500K</li> <li>• 00 : 10000K</li> </ul>

## CTC モジュールのデバイス使用率

表 6 に、CTC モジュールが使用するリソースのサマリを示します。これらの値は、ディスプレイ開発ボード上の Spartan-3E XC3S1600E FG484 FPGA に基づくものです。

表 6: CTC モジュールのリソース使用率

	使用数	利用可能数	使用率
<b>ロジック使用率</b>			
スライスのフリップフロップ数	5,275	29,504	18%
4 入力 LUT 数	3,579	29,504	12%
<b>4 入力 LUT 総数</b>	<b>3,664</b>	<b>29,504</b>	<b>12%</b>
ロジックとしての使用数	3,579		
ルートスルーとしての使用数	1		
シフトレジスタとしての使用数	84		
ボンディングされた IOB 数	58	376	15%
ブロック RAM 数	4	36	11%
MULT18X18 数	4	36	11%
GCLK 数	1	24	4%
<b>ロジック分散</b>			
使用スライス数	3,423	14,752	23%
関連ロジックのみを含むスライス数	3,423	3,423	100%
関連がないロジックのみを含むスライス数	0	3,423	0%

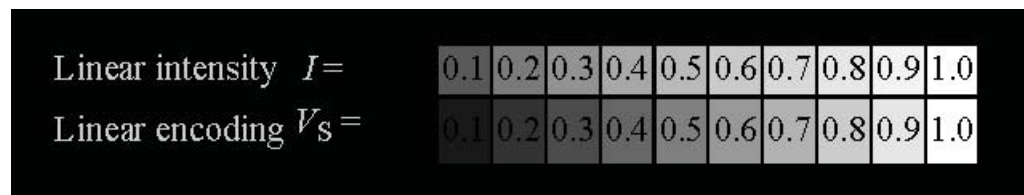
## PGC (正確なガンマ補正)

### ガンマについて

ディスプレイシステムでエンコードされた輝度と表示させたい画像の明度の間には、ガンマ特性という指数の関係があります。この関係は数学的な汎用関数では、出力 = 入力関数<sup>γ</sup> (ガンマ) として表され、逆関数によって、表示される前のデータがあらかじめ補正できます。入力と出力明度は、多くのディスプレイにおいて非線形関係であるため、ガンマ補正が必要となります。また、この補正によって知覚される画像の品質を向上させることができます。

未補正の画像や誤ったガンマ補正を施した画像は、コントラストが不鮮明になったり、色バランスが悪く、全体的な光のレベルが不適切になったりします。さらに、このような画質の不備をほかの方法で修正することは困難であるため、すべての画像で最初に適切なガンマをエンコードすることが重要です。

たとえば、図 5 に示すエンコードされた (未補正の) 線形入力信号  $V_s$  では、0.1 ~ 0.4 間の知覚明度の差が大きく、0.8 ~ 1.0 間の差は小さくなっています。基本的にガンマ関数は、このディスプレイの各入力ステップに対する出力明度が、出力  $I$  で示ように線形となるよう入力に適用されます。

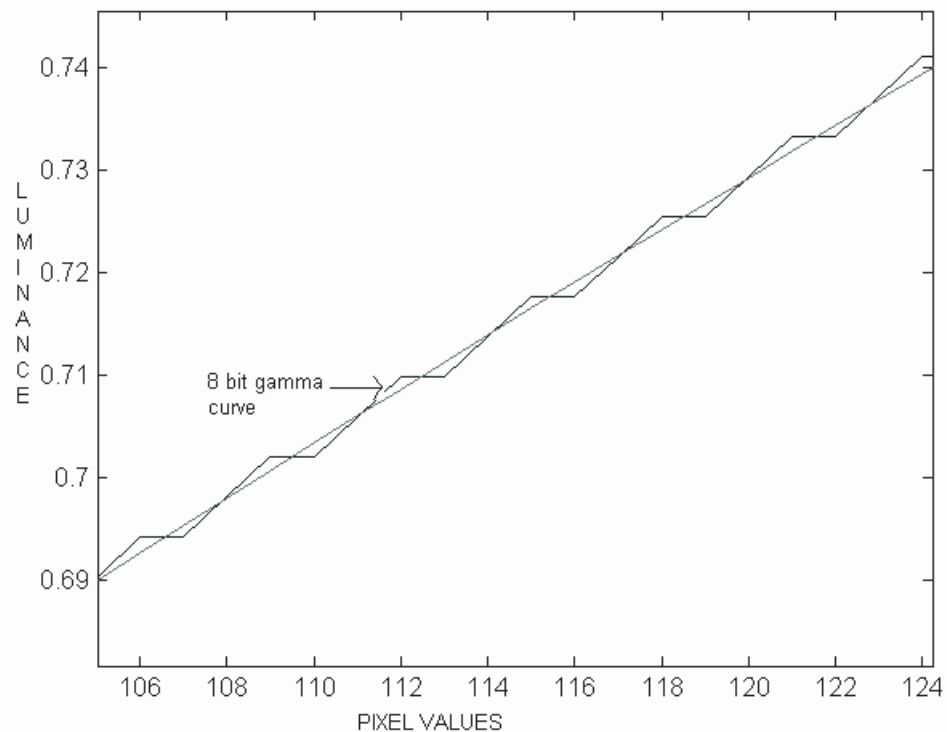


X928\_05\_040507

図 5 : 未補正 ( $V_s$ ) およびガンマ補正後 ( $I$ ) の明度

### ガンマ補正のインプリメンテーション

10 ビット出力は整数の近似値を求めやすいため、出力を正確に調整するには 10 ビット幅のガンマ曲線を選択しています。8 ビットと 10 ビットのガンマ曲線を図 6 と図 7 に示します。これらを比較してみてください。曲線には、入力ピクセル値に対するガンマ補正後の輝度値が示されています。灰色の曲線が 8 ビットまたは 10 ビットの精度の場合、黒色の曲線が実現したいレスポンスを表しています。



X928\_06\_040407

図 6 : 8 ビットのガンマ曲線 vs 実現したい線形レスポンス

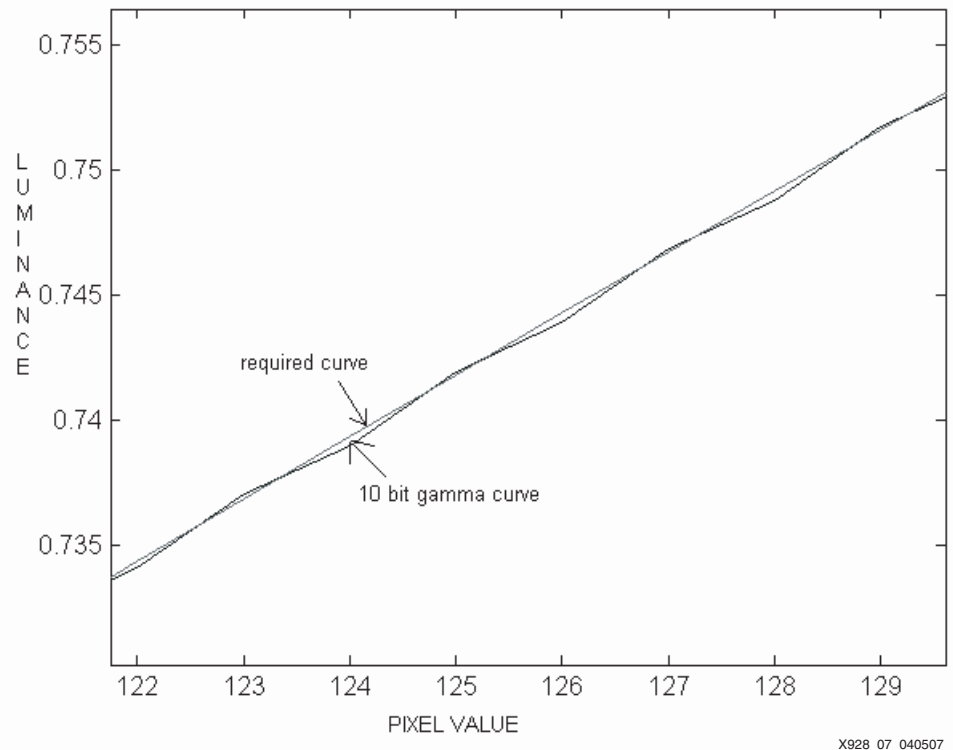


図 7: 10 ビットのガンマ曲線 vs 実現したい線形応答

これらの図からは、8 ビット出力の場合は階段のような関数となり、10 ビットの場合はスムーズで、出力に一致するよう関数で出力されていることがわかります。

10 ビットのガンマ補正は等式 2 を用いて行われました。

$$Y = \text{ROUND}\left[1023 \times \left(\frac{X}{256}\right)^\gamma \left(\frac{1}{\text{Gamma}}\right)\right] \quad \text{等式 2}$$

説明:

- X = RGB 入力 (8 ビットの R、G、B 値それぞれを整数で表す)
- Y = 10 ビットのガンマ補正 R'、G'、B' 出力
- ガンマ = ガンマ係数 (プログラム可能)

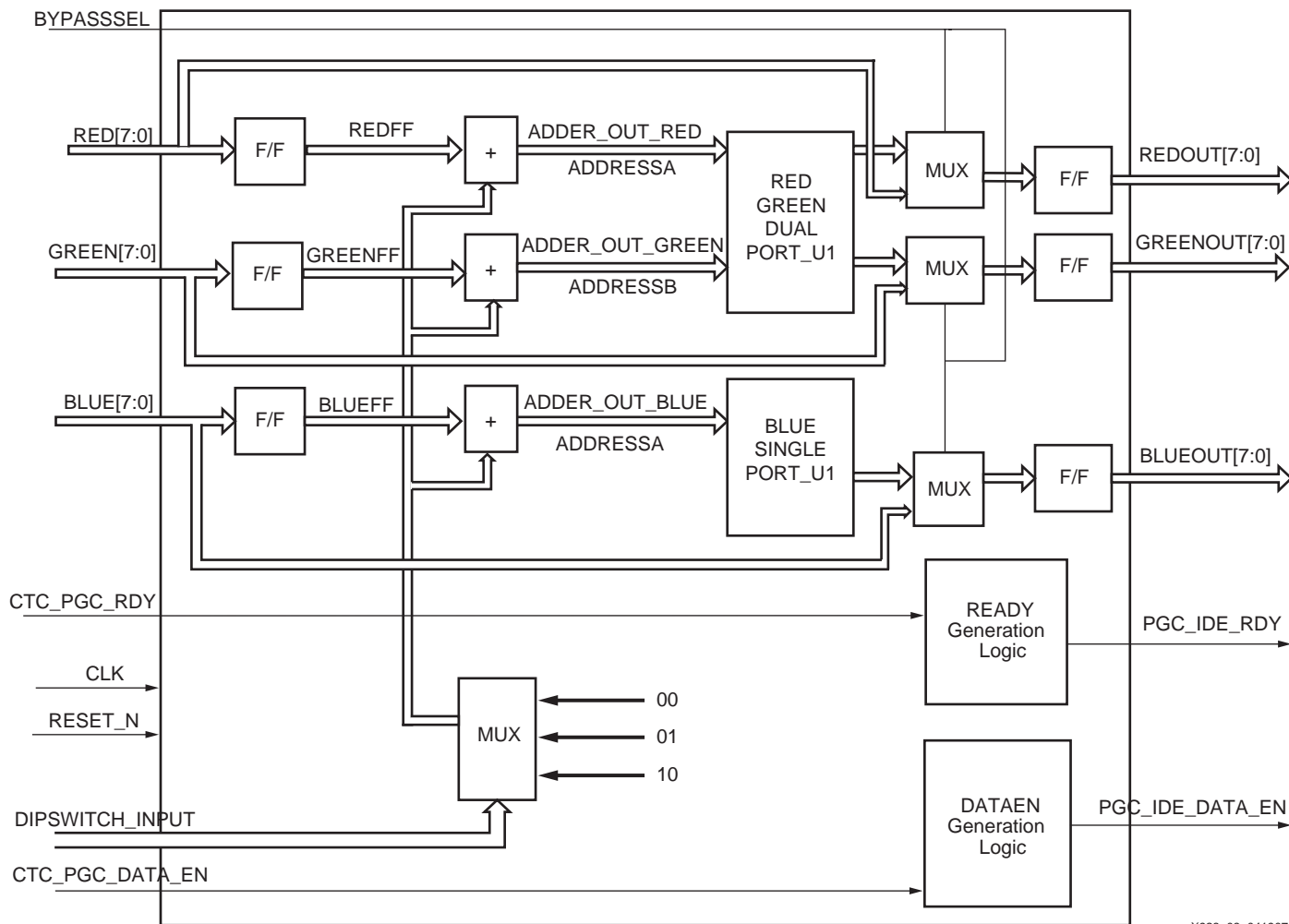
値の小数部が 0.5 以上の場合、Y は四捨五入されて近似整数値となり、それ以外の場合は小数部が切り捨てられます。出力は 10 ビット形式です。

### データパスの詳細説明

外部 DIP スイッチは、入力データに適用されるガンマ値に設定されます。ガンマ関数は一連の LUT (ルックアップテーブル) を用いて R、G、B それぞれに個別に適用されます。出力データは R'G'B' の 3 x 10 ビット データ形式です。

正確なガンマ補正が完了すると、次のモジュール IDE (イメージ デザリング エンジン) に DATA\_EN フラグがアサートされます。また、外部 DIP スイッチを使用し、このブロックをバイパスさせることも可能です。





X928\_08\_041907

図 8 : PGC のブロック図

表 7 に、PGC モジュールで使われる信号の説明を示します。

表 7 : PGC モジュールの信号

信号	I/O	説明
RESET_N	入力	アクティブ Low グローバルリセット。非同期信号。
CLK	入力	グローバルクロック。
CTC_PGC_RDY	入力	CTC モジュールはこの信号をアサートして、ピクセルデータが有効であることを PGC モジュールに伝える。
GREEN[7:0]	入力	Green 入力を含む 8 ビットバス。
BLUE[7:0]	入力	Blue 入力を含む 8 ビットバス。
RED[7:0]	入力	Red 入力を含む 8 ビットバス。
CTC_PGC_DATA_EN	入力	CTC モジュールは、フレームの有効なデータライン全体に対して、この信号を High にアサート。
BYPASSSEL	入力	PGC ブロックをバイパス可能にする。
DIPSWITCH_INPUT[1:0]	入力	これらの信号に求められるガンマ値を設定。

表 7 : PGC モジュールの信号 (続き)

信号	I/O	説明
REDOUT[9:0]	出力	PGC モジュールからのガンマ補正出力を含む 10 ビットバス。
GREENOUT[9:0]	出力	PGC モジュールからのガンマ補正出力を含む 10 ビットバス。
BLUEOUT[9:0]	出力	PGC モジュールからのガンマ補正出力を含む 10 ビットバス。
PGC_IDE_RDY	出力	PGC モジュールはこの信号をアサートして、ピクセルデータが有効であることを IDE モジュールに伝える。
PGC_IDE_DATA_EN	出力	PGC モジュールは、この信号を High にアサートし、フレームのデータライン全体が有効であることを IDE モジュールに伝える。

フリップフロップ、バウンダリ信号、レジスタのリセットステートは、すべてのビットでロジック '0' です。

### DIP スイッチの設定

表 8 に、正確なガンマ補正を行う際の DIP スイッチの設定を示します。

表 8 : PGC モジュールの DIP スイッチ設定

DIP スイッチ	ポート名	設定
DIP_SWITCH_TYPE0 DIP_SWITCH_TYPE1 ユーザー スイッチ SW3、SW2	DIPS_PGC_SEL0_IP DIPS_PGC_SEL1_IP	PGC_TYPE1 & PGC_TYPE0 [スイッチ SW3、スイッチ SW2] <ul style="list-style-type: none"> <li>00 : ガンマは 2.2 に設定</li> <li>01 : ガンマは 2.4 に設定</li> <li>10 : ガンマは 2.6 に設定</li> <li>11 : 予約済み</li> </ul>

### PGC モジュールのデバイス使用率

表 9 に、PGC モジュールが使用するリソースのサマリを示します。

表 9 : PGC モジュールのリソース

	使用数	利用可能数	使用率
<b>ロジック使用率</b>			
スライスのフリップフロップ数	50	29,504	1%
4 入力 LUT 数	31	29,504	1%
<b>4 入力 LUT 総数</b>	<b>31</b>	<b>29,504</b>	<b>1%</b>
ボンディングされた IOB 数	63	376	17%
ブロック RAM 数	2	36	6%
GCLK 数	1	24	4%
<b>ロジック分散</b>			
使用スライス数	42	14,752	1%
関連ロジックのみを含むスライス数	42	42	100%
関連がないロジックのみを含むスライス数	0	42	0%

## IDE ( イメージ デザリング エンジン )

### デザリングの概要

デザリングとは、限られた色深度から見かけ上の色深度を作成するテクニックです。デザリングされた画像では、使用不可の色に対して、表示可能な色からのピクセルを組み合わせることでそれに近い色が作成されます。肉眼では、組み合わせることで作成された色は別の色として認識されます。

たとえば、白色と黒色のみのディスプレイを使用し、デザリングによって灰色の画像が作成できます (図 9)。黒および白のピクセルをインターレースすることで、実際にはない灰色が作成されます。



図 9: デザリングによる新しい色の作成

ディスプレイ デバイスの一部には、入力データが 8 ビットでディスプレイ色深度が 6 ビットのように、入力データより小さい色深度しか持たないものがあります。入力データは切り捨て、あるいは四捨五入されますが、この方法では通常細かい情報が失われ、ある 1 色の大部分が縞模様になり、オリジナルの色とは大きく異なる色になる可能性があります。デザリングとは、このようなイメージに対して行われる処理です。

### イメージ デザリングのアルゴリズムについて

IDE モジュールは PGC モジュールから 30 ビット (10 ビット データを R'G'B' の 3 つ分) のピクセル ストリームを受信します。

IDE は有効なピクセルに対してのみデザリングを行います。このモジュールでは、10 ~ 8 ビットに対して 2x2 のデザリング マトリックスを用いる空間デザリング テクニックが使用されます。たとえば、3x10 ビットのデータがブロックに入力される場合、3 色 (R、G、B) の 2 つの LSB (最下位ビット) は最も詳しい色情報を含み、残りの 8 ビットの MSB とは区別されます。

切り捨てられた 2 つの LSB は事実上、エラーです。これら 2 ビットは 00 ~ 11 (バイナリ) の任意の値で、残りの 8 ビットの対する色のレベルを 4 段階 (0%、25%、50%、75%) で表します。エラーは隣接したピクセル全体にあります。

次の例では、2x2 マトリックスを使用したウェイトを示します。

0	1
2	3

次の例では、上記のウェイト表がディスプレイ空間全体にどのように適用されるかを示します。

0	1	0	1	.....
2	3	2	3	.....
0	1	0	1	.....
2	3	2	3	.....
.	.	.	.	.....

各ウェイトは実際にはしきい値です。入力ピクセルの最後 2 ビットの値が、表の該当箇所の値より大きい場合、2 つの LSB が示す値が残りの 8 ビットに足されます。この際、その結果の 8 ビットがオーバーフローしないよう注意する必要があります。

パターンをなくしてばらつかせるため、ウェイト表は2行ごと、またはフレームごとに回転させることができ、これによって空間ディザ全体に一時的なディザリングを追加します。

また、マトリックスウェイトはランダムに回転できるため、ディザリング処理全体がランダムになるようにできます。これは、ビデオスクリーンディスプレイに有用な方法です。入力データのLSBを使用してディザリングする場合、IDEでは2x2ピクセルブロックを用いた空間、一時空間、およびランダム空間のディザリングが実行されます。

ディザリングが完了すると、24ビットのデータ(R、G、B各8ビット)がdither\_data\_ready信号と共にLVDS/DVI TXインターフェイスに送信されます。

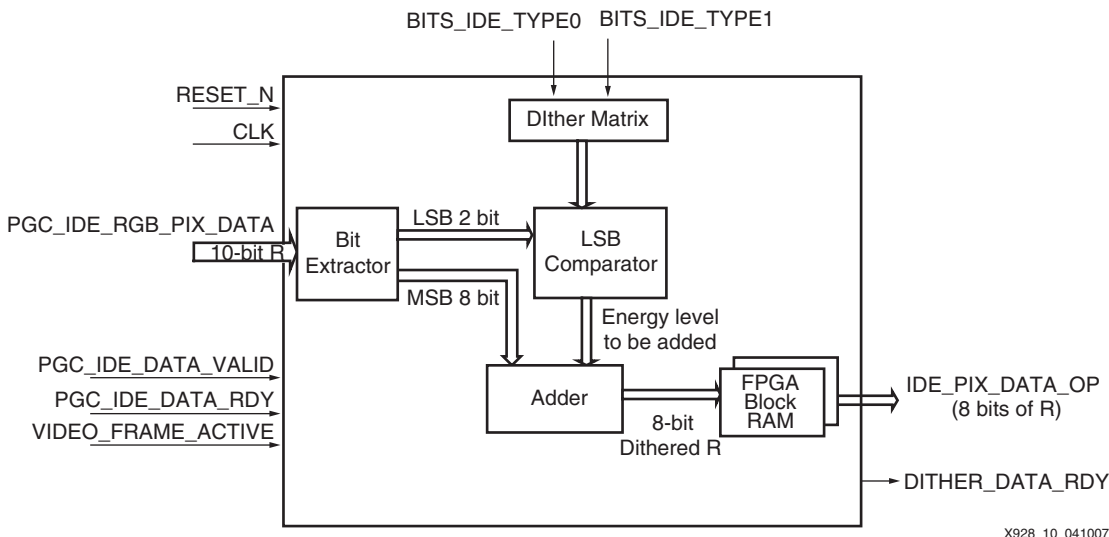


図 10 : イメージ デザリング エンジンの機能ブロック図 (ここでは 1 色のみを示す)

表 10 に、IDE モジュールで使用される信号の説明を示します。

表 10 : IDE モジュールの信号

ポート名	I/O	説明
RESET_N	入力	Low アサートにより、IDE モジュールをリセット。
CLK	入力	DCM (デジタルクロック マネージャ) によって生成されるシステムクロック。
BITS_IDE_TYPE0	入力	ディザタイプの選択。表 11 を参照。
BITS_IDE_TYPE1	入力	ディザタイプの選択。表 11 を参照。
PGC_IDE_DATA_VALID	入力	PGC モジュールはこの信号をアサートして、ピクセルデータが有効であることを IDE モジュールに伝える。ライン全体が処理されるまでアサートが維持される。
PGC_IDE_DATA_RDY	入力	PGC モジュールは、ガンマ補正後のデータが使用可能になると、IDE へのこの信号をパルスさせる。
PGC_IDE_RGB_PIX_DATA[29:0]	入力	PGC モジュールから IDE モジュールへ入力される、ガンマ補正された 10 ビットの R、G、B ピクセルデータはこのバスに出される。
VIDEO_FRAME_ACTIVE	入力	High にアサートされると、有効なフレームであることを示す。フレームの終わりで Low になる。

表 10 : IDE モジュールの信号 (続き)

ポート名	I/O	説明
IDE_PIX_DATA_OP[23:0]	出力	この出力バスにはディザリング後の 8 ビットの RGB 値が出力される。
DITHER_DATA_RDY	出力	アサートされると、ディザリグされたデータが使用可能であることを示す。

### DIP スイッチの設定

IDE モジュールは、表 11 に示すようにユーザー プッシュ ボタンを設定し、調整されます。

表 11 : IDE モジュールの DIP スイッチ設定

DIP スイッチ	ポート名	設定
DIP_SWITCH_TYPE0 DIP_SWITCH_TYPE1 [ユーザー プッシュ ボタン PSW0、PSW1]	DIPS_IDE_TYPE0_IP DIPS_IDE_TYPE1_IP	IDE_TYPE1 & IDE_TYPE0 [PSW1、PSW0] <ul style="list-style-type: none"> <li>• 00 : IDE 無効</li> <li>• 01 : 空間的ディザリング</li> <li>• 10 : 一時空間的ディザリング</li> <li>• 11 : ランダム ディザリング</li> </ul>

### IDE モジュールのデバイス使用率

表 12 に、IDE モジュールが使用するリソースのサマリを示します。

表 12 : IDE モジュールのリソース (XC3S1600EFG484 の場合)

	使用数	利用可能数	使用率
<b>ロジック使用率</b>			
スライスのフリップフロップ数	98	29,504	1%
4 入力 LUT 数	170	29,504	1%
<b>ロジック分散</b>			
使用スライス数	119	14,752	1%
関連ロジックのみを含むスライス数	119	119	100%
関連がないロジックのみを含むスライス数	0	119	0%
<b>4 入力 LUT 総数</b>	<b>188</b>	<b>29,504</b>	<b>1%</b>
ロジックとして使用される数	170		
ルートスルーとして使用される数	18		
ボンディングされた IOB 数	62	376	16%
GCLK 数	1	24	4%

## LVDS TX/DVI インターフェイス

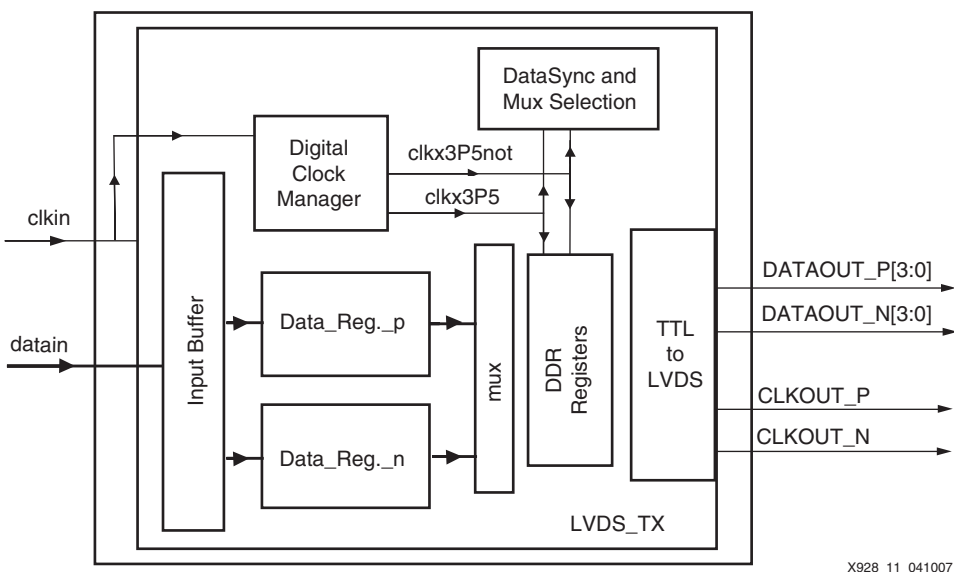
LVDS トランスミッタは、28 ビットの LVCMOS/LVTTL データを 4 つの LVDS データ ストリームに変換します。位相が同期した送信クロックが 5 つ目の LVDS リンクとして、データ ストリームと並行して入力され、28 ビットの入力データは、送信クロックの各サイクルでサンプルされて送信されます。Spartan-3E ディスプレイ開発ボードには、LVDS および DVI インターフェイスは共に備わっています。

RGB 出力インターフェイスは、24 ビットの RGB データと 4 ビットのタイミングおよび制御信号 (クロック、HSYNC、VSYNC、データ イネーブル) から成ります。

### LVDS トランスミッタのファンクション ブロック図

LVDS TX モジュール (図 11) は、DCM および serdes\_4b\_7to1 モジュールで構成されています。DCM は入力クロックを 3.5 逡倍した clkx3p5 および clkx3p5not クロックを生成します。これら 2 つのクロックは serdes\_4b\_7to1 モジュール内部で、28 ビットのデータを 4 つのラインにする 7-to-1 のシリアル化に使用されます。

ここでは clkx3p5 および clkx3p5not クロックを使用し、データをマルチプレクスすることによって 28 ビットのデータ ラインを 4 チャンネルにするシリアル化が実行されます。



X928\_11\_041007

図 11 : LVDS トランスミッタのブロック図

表 13 に、LVDS チャンネルでのデータ ビットの エンコーディングを示します (データは LSB から送信されます)。

表 13 : LVDS チャンネルでのデータ ビットのエンコード

LVDS チャンネル	入力
チャンネル 0	0, 4, 8, 12, 16, 20, 24
チャンネル 1	1, 5, 9, 13, 17, 21, 25
チャンネル 2	2, 6, 10, 14, 18, 22, 26
チャンネル 3	3, 7, 11, 15, 19, 23, 27

## トップ レベルからの デザイン階層

テストベンチを含むデザイン ファイルは、次の図に示すような階層で使用されます。これらのデザイン ファイルは次のザイリンクス ウェブ サイトから入手可能です。

<http://japan.xilinx.com/bvdocs/appnotes/xapp928.zip>

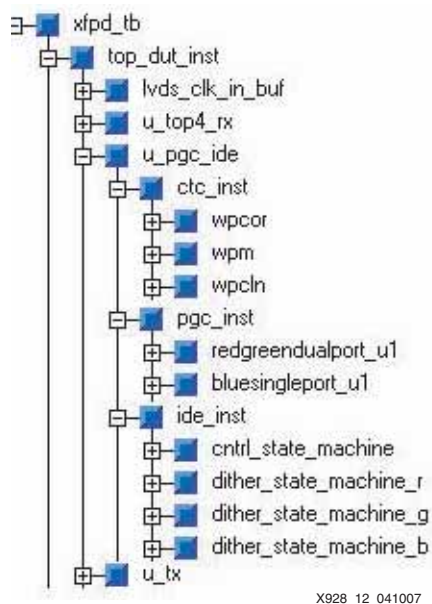


図 12：モジュール階層

## トップレベルの デバイス使用率

表 14 に、システム全体の合成結果を示します。

表 14：システム全体の合成結果

	使用数	利用可能数	使用率
<b>ロジック使用率</b>			
<b>スライス レジスタ総数</b>	<b>5,748</b>	<b>29,504</b>	<b>19%</b>
フリップフロップとしての使用数	5,743		
ラッチとしての使用数	5		
4 入力 LUT 数	3,878	29,504	13%
ロジックとしての使用数	3,878		
ルート スルーとしての使用数	33		
シフト レジスタとしての使用数	88		
<b>4 入力 LUT 総数</b>	<b>3,999</b>	<b>29,504</b>	<b>14%</b>
ボンディングされた IOB 数	93	376	25%
ブロック RAM 数	6	36	17%
MULT18X18 数	4	36	11%
GCLK 数	7	24	24%
DCM 数	3	8	38%
RPM マクロ数	2		
<b>ロジック分散</b>			
使用スライス数	3,709	14,752	25%
関連ロジックのみを含むスライス数	3,709	3,709	100%
関連がないロジックを含むスライス数	0	3,821	0%

システム レベルの I/O 信号

図 13 に、Spartan-3E FPGA を使用したディスプレイ パネル ソリューションの I/O 図を示します。

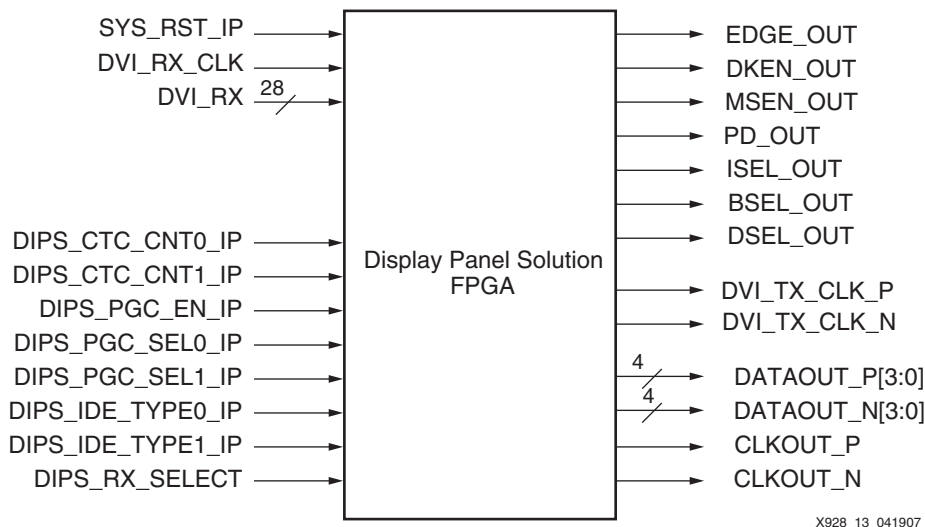


図 13: FPGA でのディスプレイ パネル ソリューションの I/O 図

CORE 仕様から次に信号タイプが使用されています。

- I: 入力標準の入力専用信号
- O: 出力標準出力

また、次の信号命名規則に従っています。

- すべてのアクティブ Low 信号に「\_N」接尾辞が付く
- 接尾辞「+」は差動信号の正側であることを表す
- 接尾辞「-」は差動信号の負側であることを表す

表 15 で Spartan-3E ディスプレイ開発ボードの信号について説明します。

表 15: ディスプレイ開発ボードで使用される信号

信号	I/O	説明
<b>システム信号</b>		
SYS_RST_IP	入力	このシステム リセット信号をアサートすると、モジュール内の全プロセスがリセットされる
<b>RGB 信号</b>		
DVI_RX_CLK	入力	DVI クロック入力
DVI_RX[27:0]	入力	DVI データ入力 (24 ビット RGB データ、VSYNC、HSYNC、DE)
<b>LVDS ディスプレイ インターフェイス信号</b>		
EDGE_OUT	出力	TFP410 の DVI_TX チップ専用信号 (DVI トランスミッタ)
DKEN_OUT	出力	TFP410 の DVI_TX チップ専用信号
MSEN_OUT	出力	TFP410 の DVI_TX チップ専用信号
PD_OUT	出力	TFP410 の DVI_TX チップ専用信号
ISEL_OUT	出力	TFP410 の DVI_TX チップ専用信号
BSEL_OUT	出力	TFP410 の DVI_TX チップ専用信号



表 15 : ディスプレイ開発ボードで使用される信号 (続き)

信号	I/O	説明
DSEL_OUT	出力	TFP410 の DVI_TX チップ専用信号
DVI_TX[27:0]	出力	DVI データ出力 (24 ビット RGB データ、VSYNC、HSYNC、DE)
DVI_TX_CLK	出力	DVI 出力クロック
DVI_TX_CLK_N	出力	TFP410 の DVI_TX チップ専用信号
<b>LVDS TX 信号</b>		
DATA_OUT_P[3:0] DATA_OUT_N[3:0]	出力	LVDS 差動出力ペア
CLKOUT_P CLKOUT_N	出力	LVDS クロック出力ペア

図 14 に、VSYNC、HSYNC、DE 信号間の同期タイミングを示します (メモ : 同期極性はターゲットとなるパネルの精度によって変更できます)。

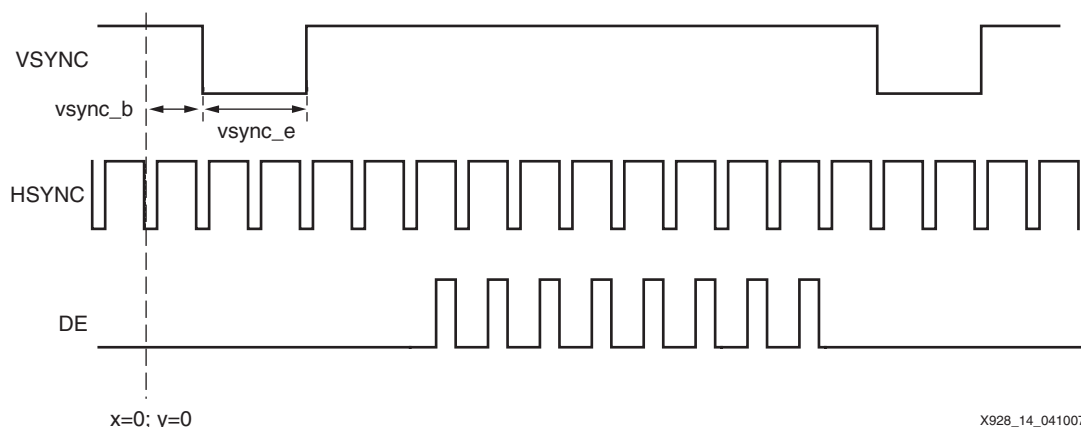


図 14 : リファレンス デザインにおける信号の関係

## まとめ

これらのデザイン ファイルは、ザイリンクスが提供する非常に優れた開発ボードでの一連のビデオ処理アルゴリズムを表し、ディスプレイ開発プログラムの設計開始をサポートします。どのザイリンクス デバイスをターゲットとしていても、すべての IP ブロックが必要に応じてカスタマイズ可能です。このシステムには、さらに新しいビデオ処理ブロックが追加でき、ご自身のビデオ拡張アルゴリズムが簡単に確認できます。

## 付録 : LVDS タイミング

詳細は、[XAPP486](#) : 『Spartan-3E FPGA における最高レート 666 Mbps での 7:1 のシリアル化』を参照してください。

## 改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	改訂内容
2007/03/23	1.0	初版リリース
2007/04/19	1.1	文書および図の微調整