

はじめに

このアプリケーション・ノートでは、現在の Altera® SDI MegaCore® ファンクションに基づく SDI (Serial Digital Interface) フライホイール・ビデオ・デコーダのリファレンス・デザインについて説明します。SDI フライホイール・ビデオ・デコーダは、ノイズも中断もなしで実際のビデオ信号ストリーミングを処理するためにビデオ・プロセッサで使用されます。SDI フライホイール・ビデオ・デコーダは、内部生成ビデオ・タイミングを受信ビデオ・ストリームに同期させます。そして、水平および垂直タイミングをビデオ・プロセッサ内の他のブロックに供給します。同期化後に、SDI フライホイール・ビデオ・デコーダは、入力ビデオ・ストリームと内部タイミング情報を継続的に比較します。違いが発生するたびに、SDI フライホイール・ビデオ・デコーダは、すぐに入力ビデオ・ストリームに再同期せずに、ビデオ・タイミングを生成させ続けています。入力ビデオ・ストリームに少数の破壊されたデータ・ワードのみが含まれる場合、内部タイミングは通常、SDI フライホイール・ビデオ・デコーダと同期して再開します。ビデオ・ストリームは異なったソースまたは規格に切り替えられる場合、それによって同期化が停止されると、SDI フライホイール・ビデオ・デコーダは受信ビデオ・ストリームに再同期させます。

以降の項では、機能の説明、インタフェース信号、およびリファレンス・デザインを使用することで使用法の手順について説明します。

機能の説明

1-2 ページの図 1 に、以下の 3 つのブロックを含む SDI トップレベル・ブロック (ALT_SDI_FLY_TOP) のブロック図を示します。

- SDI トリプル規格レシーバ・トランシーバ
- SDI トリプル規格レシーバ・プロトコル
- SDI フライホイール・ビデオ・デコーダ

図 1. SDI トップレベルのブロック図

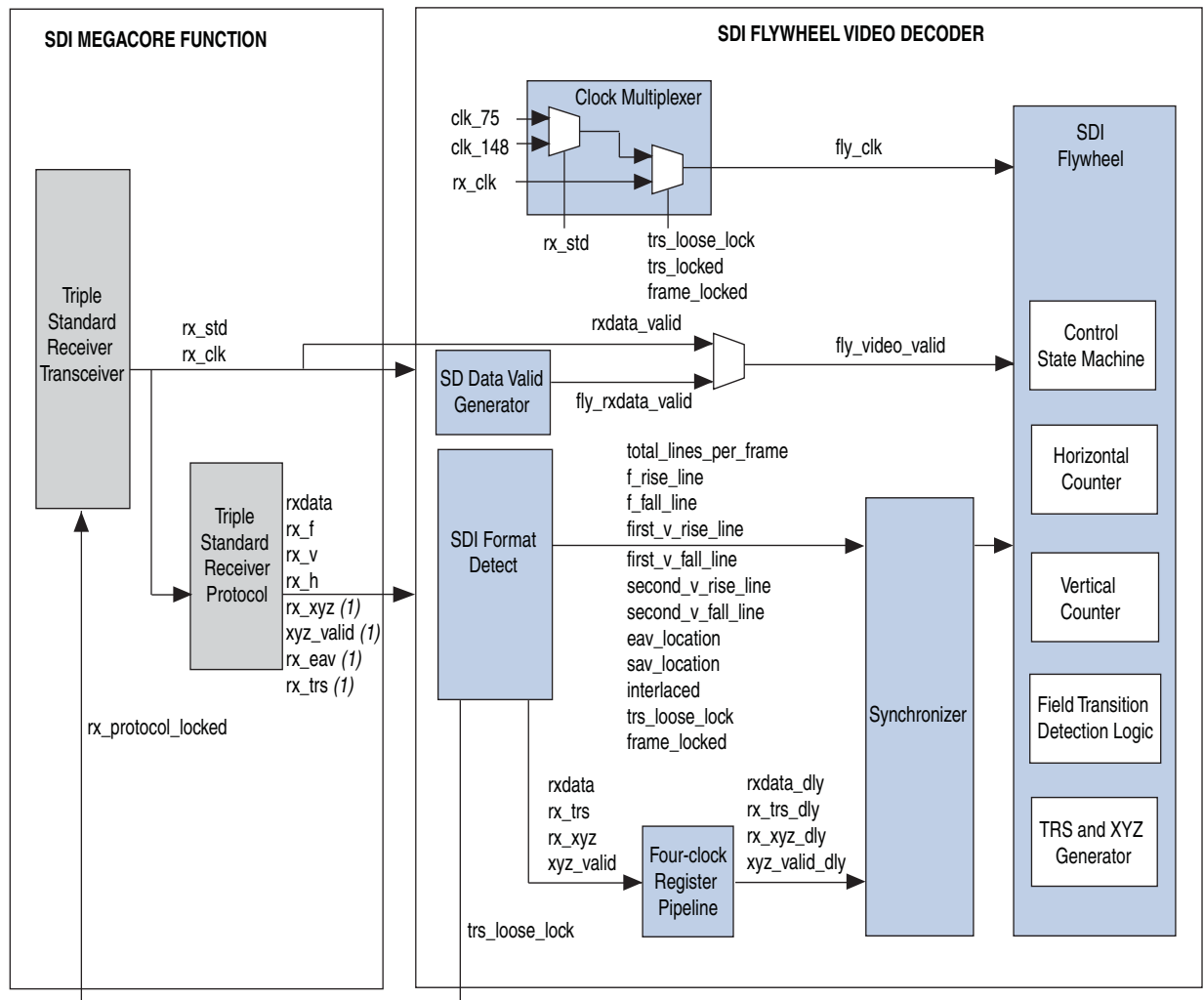


図 1 の注：

(1) これらの信号は SDI MegaCore ファンクションのバージョン 9.0 における出力ポートとして実装されます。

SDI トリプル規格レシーバ・トランシーバ

トランシーバが入力 `rx_protocol_locked` ポートがあるので、このリファレンス・デザインでは分割トランシーバが使用されます。`rx_protocol_locked` ポートはトランシーバ・コントロール・ロジックへの入力です。このポートのアクティブのとき、プロトコル・ブロックが受信ビデオ・ストリームにロックされるのをトランシーバ・コントロール・ロジックに示して、実行しているレートでトランシーバ検索アルゴリズムを停止します。非アクティブのとき、ポートはトランシーバ検索アルゴリズムを再起動します。

SDI フォーマット検出ブロックからの `trs_loose_lock` 信号はトランシーバのロックを制御します。シングルの有効な TRS (Time Reference Signal) はレシーバが有効な SDI サンプルを収集していることを示す `trs_loose_lock` 信号をアサートします。EAV (End of Active Videos) の連続不在の指定された数が検出されるとき、この信号がデアサートされます。

トリプルレート検出は、トランシーバが 3G-SDI または SD-SDI モードのいずれかに動作していることに依存します。SDI 信号がない場合、コアはレートを検出するにはトランシーバが 3G-SDI または SD-SDI モードに設定されます。HD-SDI モードが検出されると、コアはレートを検出するのにトランシーバを 3G-SDI または SD-SDI モードにリセットし、次に、トランシーバを正しく受け取ることができるように、HD-SDI モードに再プログラムします。

SDI トリプル規格レシーバ・プロトコル

SDI トリプル規格レシーバ・プロトコル・ブロック (SDI_TR_RX_PROTO) は TRS シンボルのビデオ・ストリームを検査します。また、ブロックは現在のワードが EAV またはアクティブなビデオの開始 SAV(Start of Active Video) のいずれかを示し、XYZ ワードをデコードして、ビット信号 (F、V、H) を出力します。

SDI フライホイール・ビデオ・デコーダ

SDI フライホイール・ビデオ・デコーダは以下のブロックで構成されます。

- SDI フォーマット検出
- SD データ有効なジェネレータ
- クロック・マルチプレクサ
- 4-クロック・レジスタのパイプライン
- シンクロナイザ
- SDI フライホイール

SDI フォーマット検出

SDI トリプル規格レシーバ・プロトコル・ブロックはすでに SDI フォーマット・コードを備えています。ラインとフレーム・タイミングの詳細を編集および変更することができるように、SDI フォーマット検出ブロック (SDI_FORMAT_DETECT) は既存の SDI フォーマット検出コードから強化されます。

SDI フォーマット検出ブロックは受信 SDI ストリームのラインおよびフレーム・タイミングをモニタします。このブロックはレシーバがストリームにロックされるか、またはロックが解除されているかを示す様々な信号（例えば、`trs_loose_lock`、`trs_locked`、および `frame_locked`）を生成します。また、SDI フォーマット検出ブロックは `f_rise_line`、`f_fall_line` などのフレーム情報をデコードします。



既存の SDI フォーマット・コードが使用されないため、コードで生成されたロックされた信号に関する `rx_status[4:3]` ビットを無視します。

SD データ有効なジェネレータ

SDI MegaCore ファンクションのトリプル規格の種類は SD-SDI データを受信するのに 11 回のオーバーサンプリングが必要です。このオーバーサンプリング手法に対して、データ有効なパターンは 1 High サイクル、4 Low サイクル、1 High サイクル、および 5 Low サイクルです。レシーバが受信 SD-SDI データにロックが解除されているとき、SD データ有効なジェネレータ・ブロック (SD_DATA_VALID_GEN) はデータ有効な信号

を生成します。このデータ信号で、SDI フライホイール・ブロックは、以前の SD レートでの SD-SDI ブラックレベル・ビデオ・データを生成し続けることができます。データ有効な信号は 148.5 MHz でローカルの PLL (Phase-Locked Loop) クロックに同期します。データ有効なスイッチオーバーのタイミング図について、1-4 ページの図 2 を参照してください。

クロック・マルチプレクサ

クロック・マルチプレクサ・ブロック (CLOCK_MUX) はクロック・スイッチオーバーを実行します。trs_loose_lock、trs_locked、および frame_locked 信号が High のとき、レシーバが受信データにロックされることを示し、SDI フライホイール・ブロックのクロック・ソースとしてトランシーバのリカバリ・クロック (rx_clk) を選択します。任意のロックされた信号が Low のとき、レシーバが受信データにロックが解除されていることを示し、現在受信したビデオ・レートに基づくローカルの PLL クロックを選択します。

図 2 のタイミング図に、レシーバが受信 SD-SDI 入力ストリームにロックが解除されているときのクロックおよびデータ有効なスイッチオーバーを示します。

図 2. クロックおよびデータ有効なスイッチオーバー

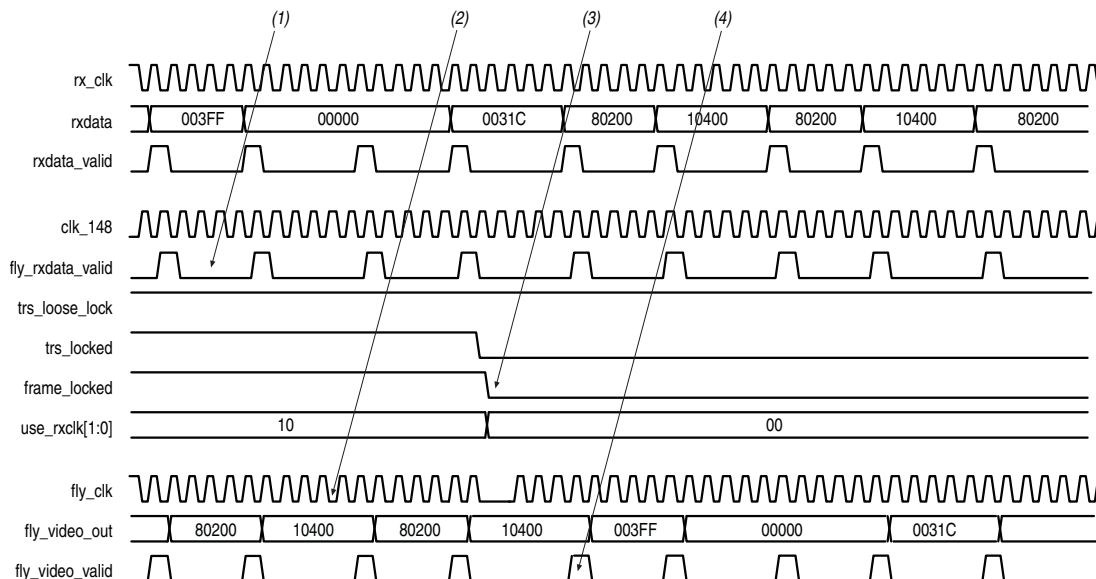


図 2 の注：

- (1) SD データ有効なジェネレータ・ブロックは 148.5 MHz でローカルの PLL クロック (clk_148) に同期する fly_rxdata_valid 信号を生成します。
- (2) レシーバが受信 SD-SDI にロックされるとき、SDI フライホイール・ブロックはクロック・ソースとして rx_clk を使用し、かつビデオ出力有効な信号として rxdata_valid を使用します。
- (3) レシーバが受信 SD-SDI にロックが解除される時、クロックおよびデータ有効なスイッチオーバーが発生します。
- (4) SDI フライホイール・ブロックはクロック・ソースとして clk_148 を使用して、ビデオ出力有効な信号として内部で生成された fly_rxdata_valid を使用します。

trs_locked 信号はレシーバが受信データにロックされ、そして rx_clk が安定することを示しますが、SDI フライホイール・ブロックは、rx_clk を選択する前に frame_locked 信号がアサートされることを待ちます。frame_locked 信号がアサートされる時、SDI フライホイール・ブロックは、入力ビデオ・ストリームに同期させるように SDI フライホイール・コントロール・ステート・マシンを開始します。ラインとフレーム・タイミングの詳細は、同期化のための SDI フライホイール・ブロックに重要で、役に立ちます。そして、詳細は frame_locked 信号がアサートされる時にのみ有効です。

4- クロック・レジスタのパイプライン

4-クロック・レジスタのパイプライン (4PIPE_REG) は 4 クロック・サイクルで入力ビデオ・ストリームおよび XYZ デコードされた信号 (rxdata、rx_trs、rx_xyz、および xyz_valid) を遅延します。入力ビデオ・ストリームおよび XYZ デコードされた信号を遅延することによって、クロック・レジスタ・パイプラインは TRS (EAV) の最初のワードで SDI トリプル規格レシーバ・プロトコル・ブロックからの rx_eav 信号をアラインメントします。

最初の 3 つのワード (3FF、000、000) がビデオ・ストリームに固有のパターンを形成します。遅延された入力ビデオ・ストリームの最初のワード (3FF) にアサートされたため、rx_eav 信号は、現在 rx_eav_first 信号であると考えられます。

図 3 に、SDI フライホイール・ブロックに移動する前に、入力ビデオ・ストリームおよび XYZ デコードされた信号が 4 クロック・サイクルにより遅延される方法を示します。

図 3. 入力ビデオ・ストリームおよび XYZ デコードされた信号の遅延

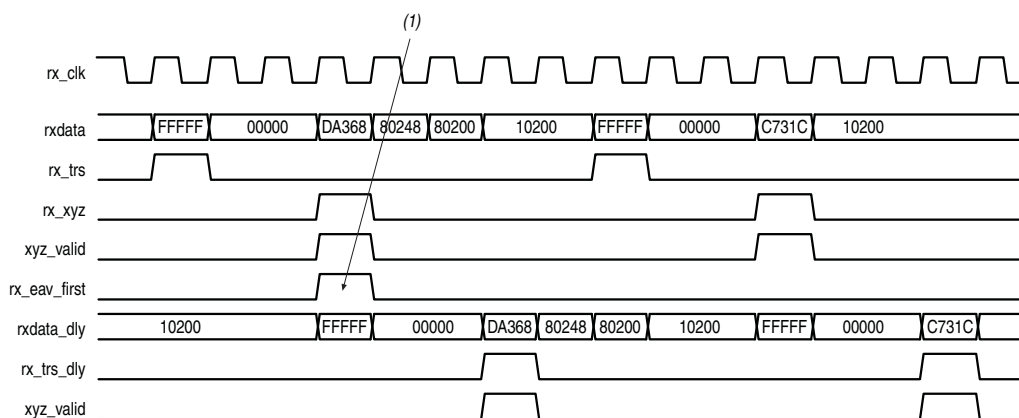


図 3 の注：

- (1) 現在、rx_eav_first 信号は TRS (EAV) の最初のワードに付属します。この信号は SDI トリプル規格レシーバ・プロトコル・ブロックから供給される rx_eav 信号と同じです。

シンクロナイザ

シンクロナイザ・ブロック (SYNC_TO_FLY_CLK) は準安定保護のためのデータを fly_clk ドメインに同期させます。frame_locked 信号の遅延信号は同期化の間に、SDI フライホイール・ブロックが安定なラインおよびフレーム・タイミングの詳細を得ることを確認するのにイネーブル・ラインとして使用されます。

SDI フライホイール

SDI フライホイール・ブロックは、ノイズまたは中断される入力ビデオ・データがある状態で、ビデオ・タイミングを提供します。SDI フォーマット検出ブロックはレシーバが入力ビデオ・ストリームにロックされることを示すために、`frame_locked` 信号をアサートするとき、SDI フライホイール・ブロックは入力ビデオ・ストリームに内部データを同期させます。同期化後に、生成されたビデオ・タイミングは入力ビデオ・ストリーム・タイミングに対応します。SDI フライホイール・ブロックは再び同期化する前に、発生する EAV の連続不在でかなりの数を必要とすることによってノイズ耐性を提供します。

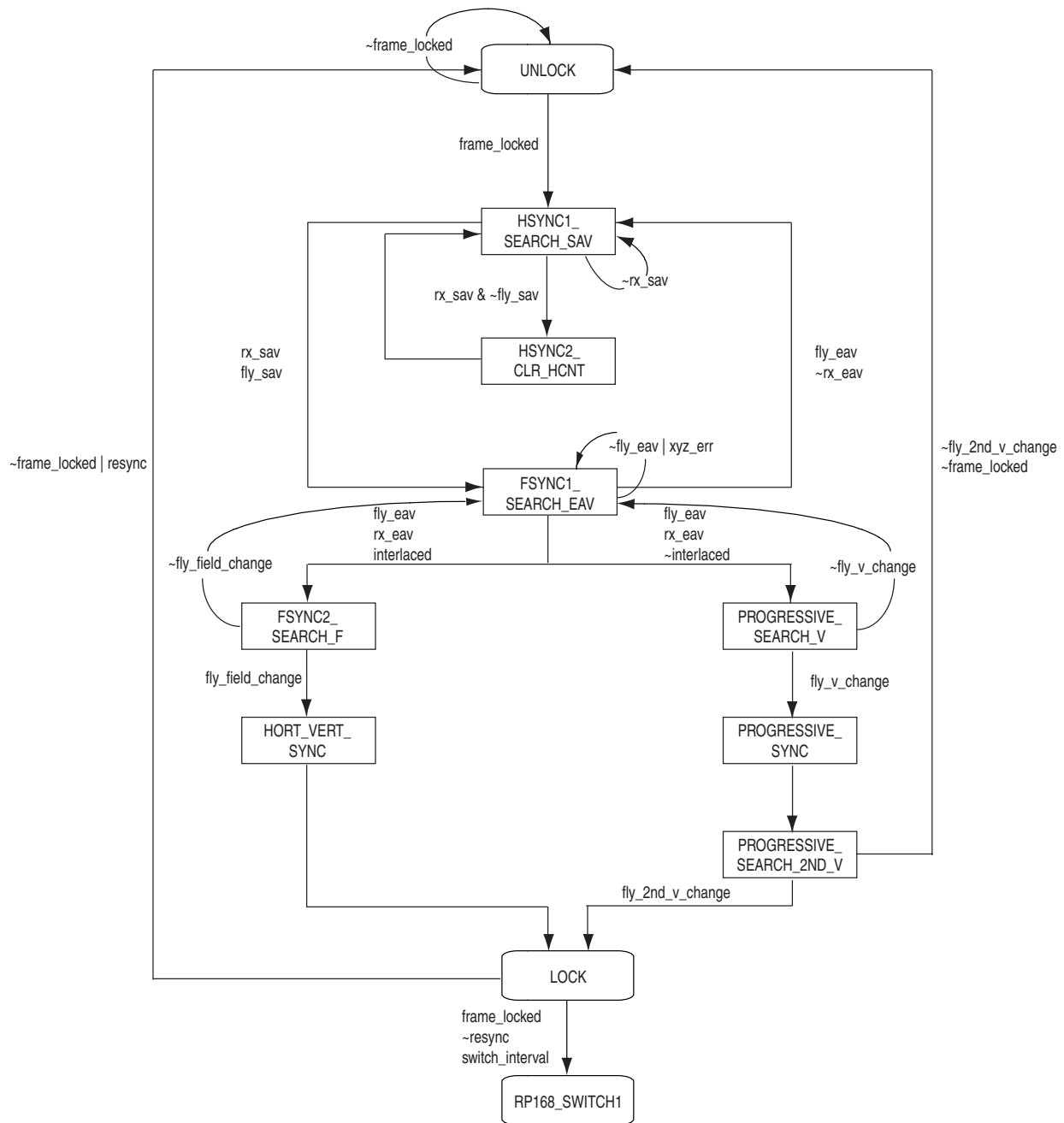
SDI フライホイール・ブロックは TRS シンボルを生成して、ビデオ・ストリームに挿入します。入力ビデオ・ストリームにおける損傷された TRS シンボルを修理するために、ブロックは入力ビデオ・ストリームのデータを上書きします（ここで、TRS シンボルが発生します）。完全な不在 TRS シンボルが指定したスレッシュホールドを超える場合、SDI フライホイール・ブロックは、ブラック・レベル・ビデオ・データを生成して、以前のタイミングで入力ビデオ・ストリームの TRS シンボルに代わって、それらを挿入します。

SDI フライホイール・ブロックには、以下のサブ・ブロックが含まれています。

- **水平・カウンタ (SDI_FLY_HCNT)** — 各有効なワードに対してインクリメントされます。また、それは TRS シンボル信号 (`fly_trs`)、XYZ シンボル信号 (`fly_xyz`)、および水平・ブランキング信号 (`fly_h_blank`) を生成します。
- **バーティカル・カウンタ (SDI_FLY_VCNT)** — EAV が検出されるときにインクリメントします。また、それはバーティカル・ブランキング信号 (`fly_v_blank`) を生成します。
- **フィールド遷移検出ロジック (SDI_FLY_FCNT)** — インターレース・フレームに対するフィールド遷移検出ロジックが含まれています。このロジックは入力ビデオ・ストリームにおけるすべての EAV からの F ビットをキャプチャして、ビットを以前の EAV からの F ビットと比較します。F ビットが変更したとき、このブロックは新しいフィールドの開始を検出して、`fly_field_change` 信号をアサートします。また、F ビット (`fly_f`) も生成します。
- **TRS および XYZ ジェネレータ (SDI_FLY_TRS_XYZ_GEN)** — フライホイール・ビデオ・デコーダの内部タイミングに基づく TRS シンボル (3FF、000、000、XYZ) および XYZ ワードを生成します。そして、それらをビデオ・ストリームに挿入します。それはレシーバが入力ビデオ・ストリームにロックが解除される時、ブラックレベル・ビデオ・データを生成します。
- **コントロール・ステート・マシン (SDI_FLY_CTRL_FSM)** — フライホイール・ビデオ・デコーダの動作を制御します。このステート・マシンはメイン・ループおよび同期ループに分けられます。

図 4 に、メイン・ループの機能を示します。

図 4. メイン・ループ



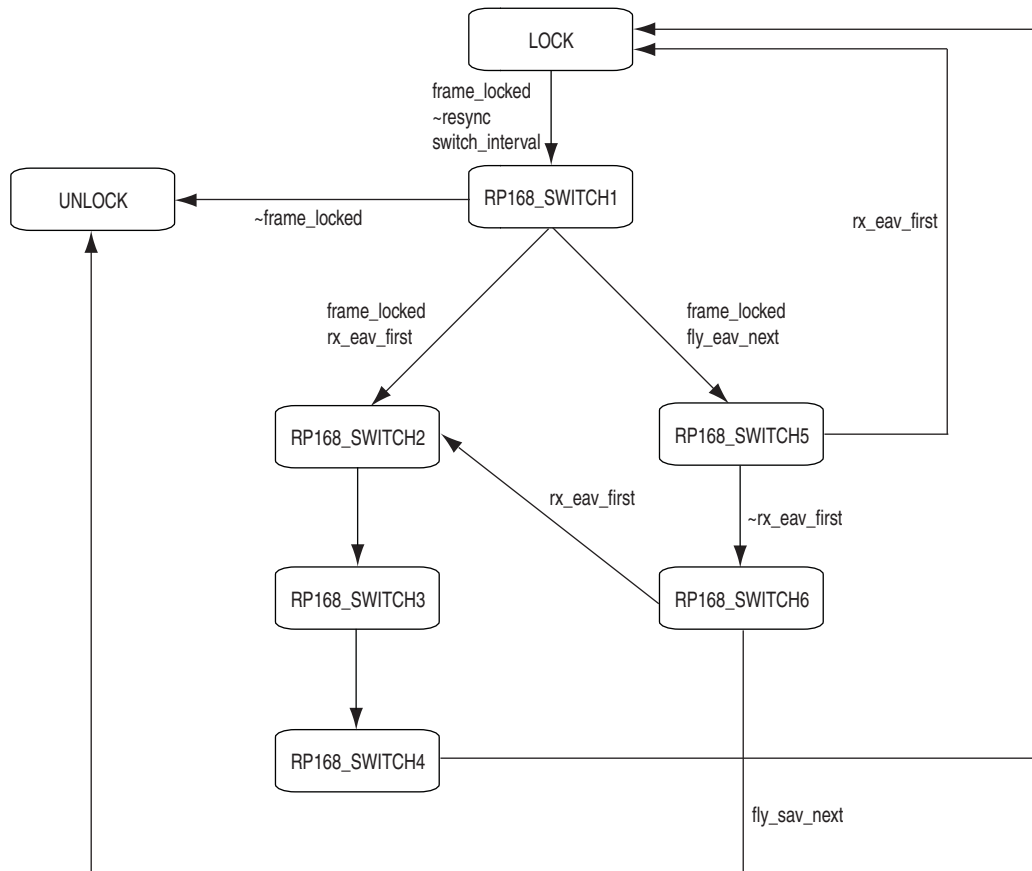
以下のステップでは、図 4 でのプロセスについて説明します。

1. FSM (Finite State Machine) は UNLOCK ステートで開始します。レシーバが入力ビデオ・ストリームにロックされることを示すために、SDI フォーマット検出ブロックが `frame_locked` 信号をアサートするまで、このステートに留まります。そして、FSM は入力ビデオ・ストリームに同期を試みます。FSM は入力ビデオ・ストリームに同期するのに 2 つのフィールド以内（または 1 つのフレーム以内）に完了します。

2. HSYNC1_SEARCH_SAV ステートでは、FSM は最初に、入力ビデオ・ストリームで SAV シンボルを探すことによって、水平に同期します。SAV が受信するたびに、FSM は水平・カウンタ・ブロックにおけるフライホイール・ビデオ・デコーダの水平・カウンタをクリアします。クリアは、入力ビデオ・ストリームの SAV の同じタイミングにフライホイール・ビデオ・デコーダが SAV を生成するまで繰り返されます。フライホイール・ビデオ・デコーダの SAV と入力の SAV の位置が合っているとき、FSM は入力ビデオ・ストリームに水平に同期します。
3. FSM は FSYNC1_SEARCH_EAV ステートに遷移して、垂直に同期しようとした。FSM は、以下の方法の 1 つで垂直的に同期するように新しいフィールドの開始を探します。
 - インターレース・フレームに対して、FSM はフィールド遷移検出ロジック・ブロックからの fly_field_change 信号を待ちます。新しいフィールドが検出されるとき、ステート・マシンはラインおよびフレーム・タイミングに基づく決定した値でバーティカル・カウンタをロードします。
 - プログレッシブ・フレームに対して、FSM は垂直的に同期させるように立ち下り V ビットを待っています。バーティカル・カウンタ・ブロックによって立ち下り V ビットが検出されるとき、ステート・マシンはラインおよびフレーム・タイミングに基づく決定した値でバーティカル・カウンタをロードします。
4. バーティカル・カウンタがロードされると、フライホイール・ビデオ・デコーダは入力ビデオ・ストリームに同期されます。FSM は LOCK ステートに移動して、fly_locked 信号をアサートします。SDI フォーマット検出ブロックが入力ビデオ・ストリームの不一致の F と V ビット、または指定したスレッシュホールドを超える EAV の連続不在が検出されるまで、FSM はロック状態を保持します。不一致のビットまたは不在 EAV が検出されるとき、レシーバは入力ビデオ・ストリームにロックが解除されて、FSM は UNLOCK ステートに移動します。
5. FSM は UNLOCK ステートに移動すると、フライホイールが再同期するまで、フライホイール・ビデオ・デコーダは内部タイミングに基づく有効な TRS シンボルを発生させ続けています。SDI フォーマット検出ブロックは次の新しい入力ビデオ・ストリームにおけるフレーム・フォーマット変更を検出するまで、FSM は fly_locked 信号をネゲートしません。すべてのダウンストリームのビデオ機器を同期させ続けるので、fly_locked 信号をネゲートしないことは、役に立ちません。


図 5 に、同期スイッチング・ループの機能を示します。

図 5. 同期スイッチング・ループ



以下のステップでは、図 5 でプロセスについて説明します。

1. 同期ビデオ・ソースが切り替えられるとき、水平・タイミングとストリームのアラインメントを妨害しますが、バーティカル・タイミングが同期されたままになります。

 SMPTE RP168 により、同期スイッチングが 1 フィールドあたり 1 つのラインで発生する可能であることを定義します。SDI フライホイール・ビデオ・デコーダはこれらの同期スイッチング・ラインの上に発生する水平・オフセットに対応します。そして、オフセットが検出される場合、それをすぐに受信ビデオ・ストリームに再同期します。

2. 同期スイッチング中に、en_sync_switch パルスをアサートすると、FSM が RP168_SWITCH1 ステートに移動します。このステートでは、FSM はその内部の EAV と入力ビデオ・ストリームにある EAV が同時に発生するかどうかを決定します。そうしないと、FSM は、入力ビデオ・ストリームで EAV シンボルの位置を合わせるために水平・カウンタを再ロードします。

同期スイッチング・ループの例について、図 17 および図 19 を参照してください。

インタフェース信号

表 1 に、SDI トップレベル・ブロックの入力ポートと出力ポートを示します。

表 1. インタフェース信号 (その 1)

ポート	幅	入力/出力	説明
rst	1	入力	レシーバおよびフライホイール・ビデオ・デコーダをリセット状態に維持するリセット信号。 rx_serial_refclk クロック・ドメインに同期しなければなりません。
rx_serial_refclk	1	入力	SDI トリプル規格のトランシーバ・トレーニング・クロック。
clk_148	1	入力	ローカル PLL クロック (148.5 MHz)。
clk_75	1	入力	ローカル PLL クロック (74.25 MHz)。
gxb2_cal_clk	1	入力	Stratix II GX トランシーバのキャリブレーション・クロック。
sdi_rx	1	入力	シリアル入力。
enable_sd_search	1	入力	トリプル規格モードにおける SD 信号の検索を可能にします。
enable_hd_search	1	入力	トリプル規格モードにおける HD 信号の検索を可能にします。
enable_3g_search	1	入力	トリプル規格モードにおける 3G 信号の検索を可能にします。
en_sync_switch	1	入力	ダウンストリームが完全に非破壊的であるように、アライナ、フォーマット検出、およびフライホイール・ブロックをすぐにリアラインメントすることを可能にします。
sdi_reconfig_clk	1	入力	エンベデット・トランシーバ・インスタンスに対するクロック入力。
sdi_reconfig_togxb	4	入力	エンベデット・トランシーバ・インスタンスに対するデータ入力。Stratix II GX DPRIO ビット、sdi_reconfig_togxb[3] はこのリファレンス・デザインで使用されません。
sdi_reconfig_fromgxb	17	出力	エンベデット・トランシーバ・インスタンスからのデータ出力。Stratix II GX DPRIO ビット、sdi_reconfig_fromgxb[16:1] はこのリファレンス・デザインで使用されません。
sdi_start_reconfig	1	出力	リコンフィギュレーションを開始するのに SDI MegaCore ファンクションからの要求。
sdi_reconfig_done	1	入力	リコンフィギュレーションの完了を示します。
rx_clk	1	出力	トランシーバ・クロック・データ・リカバリ (CDR) クロック。
rx_std (1)	2	出力	受信ビデオ規格。 00 = SD、01 = HD、10 = 3G

表 1. インタフェース信号 (その 2)

ポート	幅	入力/出力	説明
rx_status (1)	11	出力	レシーバ・ステータス : rx_status[0]— トランシーバ PLL ロック rx_status[1]— リセットのレシーバ rx_status[2]— アラインメント・ロック (TRS が検出済み、ワード・アラインメントが実行済み) rx_status[10:3]— このリファレンス・デザインで未使用
rx_ln (1)	22	出力	レシーバ・ライン数の出力。 ビット 11 ~ 21 がこのリファレンス・デザインで使用されません。
trs_loose_lock (1)	1	出力	トランシーバ・コントロール・ステート・マシンは、コアがリセットされるべきであるか、またはコアが別のビデオ規格を検索されるべきであるかどうかを決定するのにこの信号を使用します。シングルと有効な TRS がこの信号をアサートします。EAV の連続不在の数 (ユーザー・プログラマブル) がこの信号をディアサートします。 アサートされると、トランシーバ・コントロール・ステート・マシンはロックして、トランシーバ検索アルゴリズムを現在のレートで停止します。ディアサートされると、トランシーバ・コントロール・ステート・マシンはトランシーバ (3G モードに再プログラムされた) をリセットして、検索アルゴリズムを再開します。
trs_locked (1)	1	出力	同じ EAV と SAV タイミングの 6 つの連続ラインはこの信号をアサートします。
frame_locked (1)	1	出力	複数のフレームのなかで、F または V ビット上で、一定になった立ち上がりエッジまたは立ち下がりエッジはこの信号をアサートします。
rx_ap (1)	2	出力	レシーバ・アクティブ・ピクチャーの同期化出力。 ビット 1 がこのリファレンス・デザインで使用されません。
fly_clk	1	出力	フライホイール出力クロック。trs_loose_lock、trs_locked、および frame_locked が High のとき、この出力クロックは rx_clk に相当します。それ以外の場合、それはレシーバ・ビデオ規格に基づく clk_148 または clk_75 に相当します。
fly_trs (2)	1	出力	フライホイール TRS フラグ。TRS シンボルの間にアサートされます (3FF、000、000、XYZ)。
fly_eav_next (2)	1	出力	フライホイール EAV インジケータ・フラグ。次のワードが EAV の最初のワードである際に、アサートされます。
fly_sav_next (2)	1	出力	フライホイール SAV インジケータ・フラグ。次のワードが SAV の最初のワードである際に、アサートされます。
fly_xyz_word (2)	1	出力	フライホイール XYZ ワード・フラグ。現在のワードが TRS の XYZ ワードであるときアサートされます。

表 1. インタフェース信号 (その 3)

ポート	幅	入力/出力	説明
fly_video_out (2)	20	出力	フライホイール出力ビデオ・ストリーム。
fly_video_valid (2)	1	出力	フライホイール出力ビデオ・ストリーム有効な信号。
fly_f (2)	1	出力	フライホイール・フレームの同期化出力。
fly_v_blank (2)	1	出力	フライホイール・バーティカル同期化出力。
fly_h_blank (2)	1	出力	フライホイール・水平同期化出力。
fly_vcount (2)	14	出力	フライホイール現在のバーティカル・カウント。
fly_hcount (2)	11	出力	フライホイール現在の水平・カウント。
fly_locked (2)	1	出力	フライホイール同期化およびロックされた信号。フライホイール・ビデオ・デコーダが入力ビデオ・ストリームに同期されるときにアサートされます。レシーバは新しい入力ビデオ・ストリームでフォーマットの変更を検出するときにディアサートされます。
fly_std (2)	2	出力	フライホイール・ビデオ規格。 00 = SD、01 = HD、10 = 3G
format_change (2)	1	出力	フライホイール・フォーマット変更インジケータ信号。新しい入力ビデオ・ストリームのフレーム・フォーマットがフライホイール・ビデオ・デコーダの内部で生成されたフレーム・フォーマットに一致していないときにアサートされます。

表 1 の注:

- (1) rx_clk ドメインに同期します。
(2) fly_clk ドメインに同期します。

使用法

この項では、Stratix II GX Audio Video 開発ボードを使用してフライホイール・ビデオ・デコーダ・リファレンス・デザインを実証するための要件および関連する手順について説明します。この項では、以下の章で構成されています。

- ハードウェアおよびソフトウェア要件
- ディレクトリ構造
- ハードウェア設定
- リファレンス・デザインのデモンストレーション

ハードウェアおよびソフトウェア要件

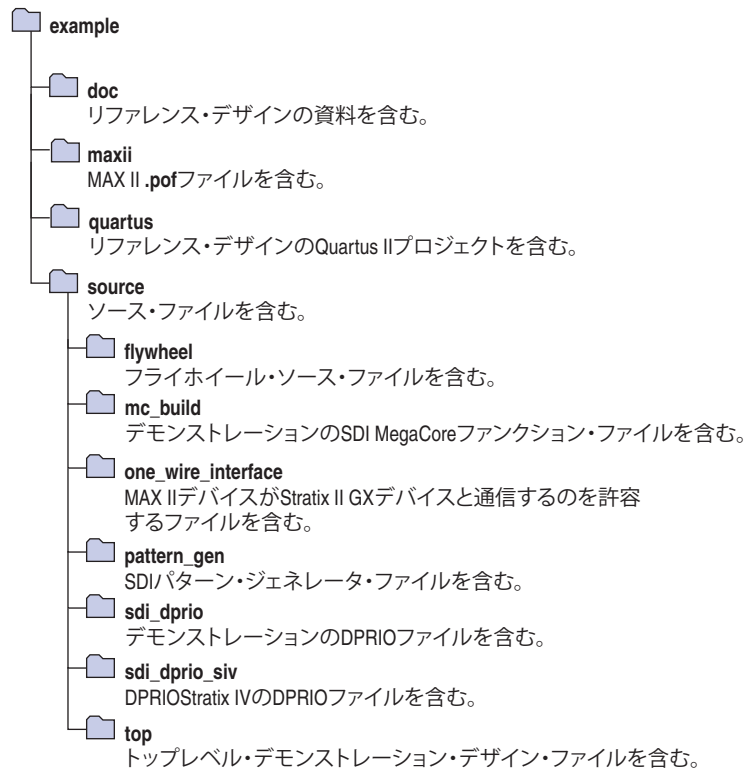
デモンストレーションでは、以下のハードウェアとソフトウェアが要求されます。

- Stratix® II GX オーディオ・ビデオ開発ボード
- SDI MegaCore ファンクション v9.0 以降
- Quartus II ソフトウェア v9.0 以降

ディレクトリ構造

図 6 に、リファレンス・デザインのディレクトリ構造を示します。

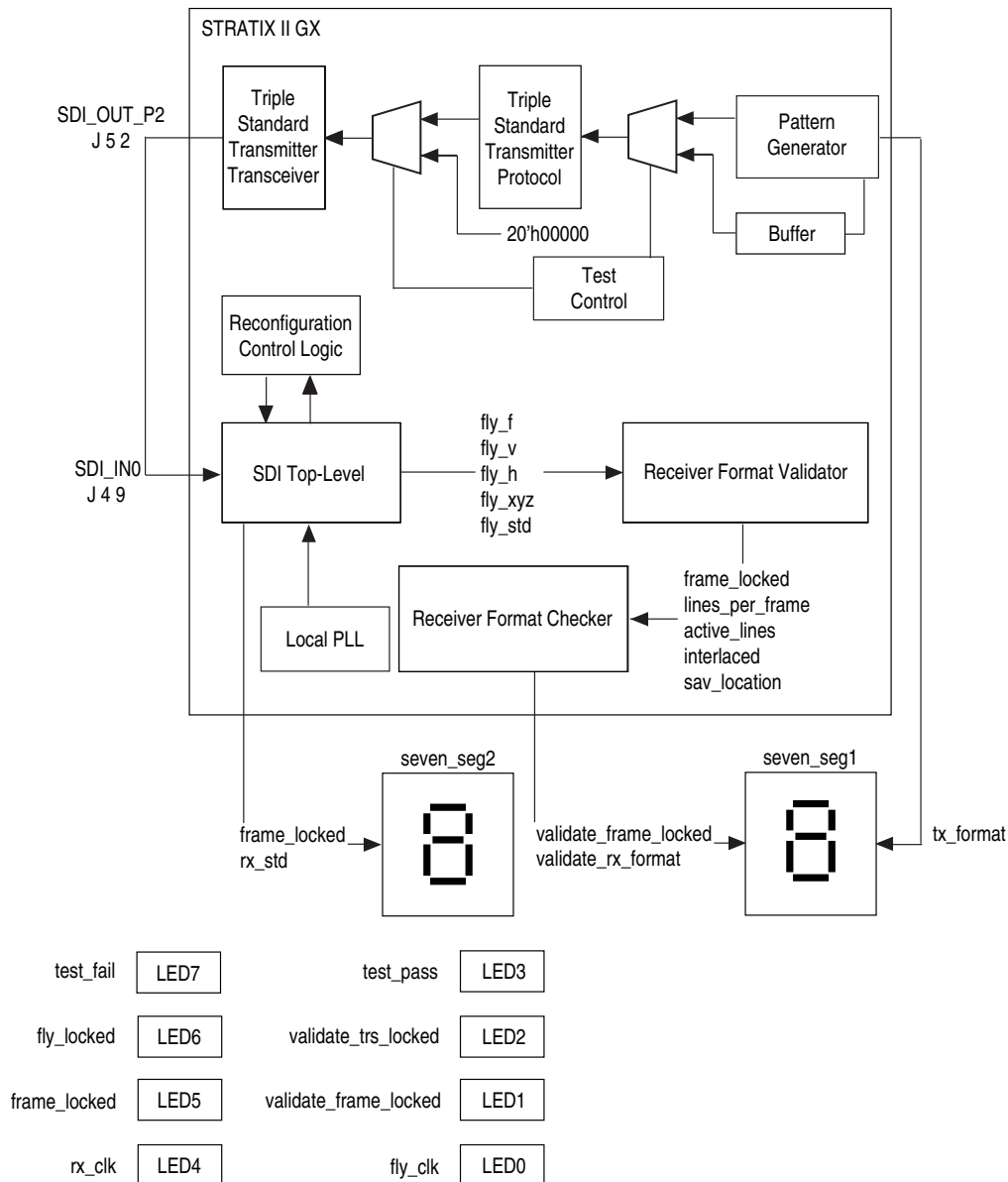
図 6. ディレクトリ構造



ハードウェア設定

図 7 に、デモンストレーションのハードウェア・システムの設定を示します。

図 7. ハードウェア・システムの設定



SDI トップレベル・ブロック

SDI トリプル標準トランスミッタ、SDI トリプル標準トランスミッタ・プロトコルおよび SDI フライホイール・ビデオ・デコーダ周辺に包み込む SDI トップレベル・ブロック (ALT_SDI_FLY_TOP) です。

パターン・ジェネレータ

パターン・ジェネレータは以下のいずれかのテスト・パターンを生成します。

- 2.97-Gbps 3G-SDI

- 1.485-Gbps HD-SDI
- 270-Mbps SD-SDI

表 2 に示すとおりに、テスト・パターン・ジェネレータはさまざまなフォーマットを生成するようにカスタマイズされます。

表 2. テスト・パターン・ジェネレータで生成されるフォーマット

SMPTE	フォーマット	ワード/アクティブなライン	アクティブなライン/フレーム	ワード/総ライン	総ライン/フレーム	フレーム・レート
260 M	1035i	1,920	1,035	2,200	1,125	30
274 M	1080i	1,920	1,080	2,200	1,125	30
				2,640	1,125	25
	1080p	1,920	1,080	2,200	1,125	60
				2,640	1,125	50
				2,200	1,125	30
				2,640	1,125	25
				2,750	1,125	24
295 M	1080i	1,920	1,080	2,376	1,250	25
296 M	720p	1,280	720	1,650	750	60
				1,980	750	50
				3,300	750	30
				3,960	750	25
				4,125	750	24
125 M	NSTC	1,440	487	1,716	525	30
	PAL	1,440	576	1,728	625	25

SDI トリプル規格トランスミッタ・トランシーバ

SDI トリプル規格トランスミッタはパターン・ジェネレータからの入力を取り込んで、同じテスト・パターンを生成します (2.97-Gbps 3G-SDI、1.485-Gbps HD-SDI、または 270-Mbps SD-SDI)。RP168 機能をテストしているときのデッドタイム期間中に、分割トランシーバおよびプロトコル・コンフィギュレーションを使用して、すべてゼロを送信します。

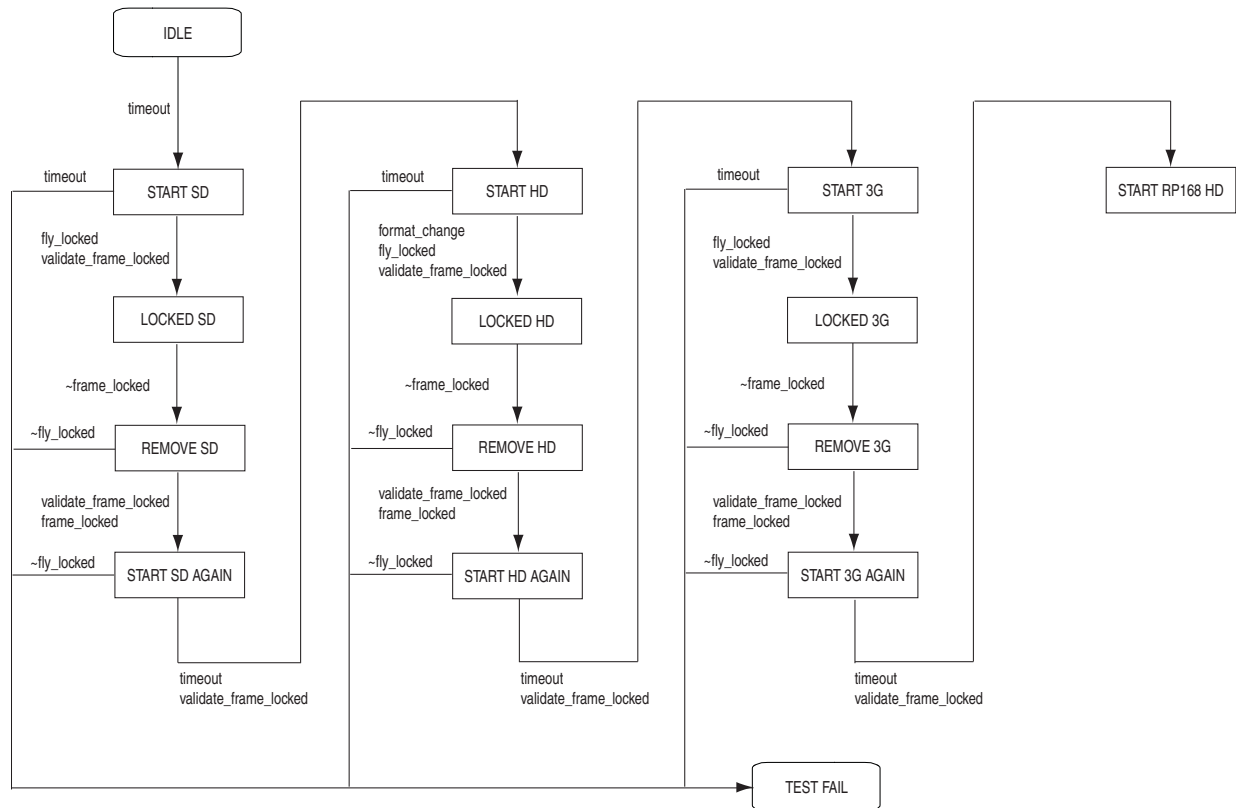
テスト・コントロール

テスト・コントロール・ブロックは FSM に基づいています。この FSM が 2 つのメイン・ループがあります。

- 非同期スイッチング・ループ
- 同期スイッチング・ループ

図 8 に、FSM における非同期スイッチング・ループを示します。

図 8. 非同期スイッチング・ループの図



SDI トップレベル・ブロック (ALT_SDI_FLY_TOP) は各ビデオ規格に認識およびロックするかどうかを判断するために、送信する SD-SDI、HD-SDI、および 3G-SDI を通して非同期スイッチング・ループをサイクルします。システム・トップレベル・ファイル (**tr_sdi.v**) では、各ビデオ規格の送信フォーマットに対して以下のデフォルト設定が使用されます。

```

// user specified Transmit format
parameter [3:0] TX_FORMAT_SD = FORMAT_PAL;
parameter [3:0] TX_FORMAT_HD = FORMAT_GH;
parameter [3:0] TX_FORMAT_3G = FORMAT_I_3G;
  
```

各ビデオ規格に対する以下の TX_FORMAT_XX パラメータを変更することによってサポートされたどんなフォーマットも送信するのを選択することができます。

```
// This group of parameters defines the transmit format that supported
parameter [3:0] FORMAT_NTSC = 4'b0000; // 25M SD
parameter [3:0] FORMAT_PAL  = 4'b0001; // 125M SD
parameter [3:0] FORMAT_AB   = 4'b0010; // 260M HD 1035i30
parameter [3:0] FORMAT_C    = 4'b0011; // 295M HD 1080i25
parameter [3:0] FORMAT_DE   = 4'b0100; // 274M HD 1080i30
parameter [3:0] FORMAT_F    = 4'b0101; // 274M HD 1080i25
parameter [3:0] FORMAT_GH   = 4'b0110; // 274M HD 1080p30
parameter [3:0] FORMAT_I    = 4'b0111; // 274M HD 1080p25
parameter [3:0] FORMAT_JK   = 4'b1000; // 274M HD 1080p24
parameter [3:0] FORMAT_LM   = 4'b1001; // 296M HD 720p60
parameter [3:0] FORMAT_N    = 4'b1010; // 296M HD 720p50
parameter [3:0] FORMAT_O    = 4'b1011; // 296M HD 720p30
parameter [3:0] FORMAT_G_3G = 4'b1100; // 3G 1080p60
parameter [3:0] FORMAT_I_3G = 4'b1101; // 3G 1080p50
```

また、入力ビデオは中断されるか、または取り外されるときに、フライホイール・ビデオ・デコーダは TRS シンボルを発生する方法をこのテストで示しています。トランシーバ・コントロール・ステート・マシンは、SDI が正しく受け取られるかどうか決定するのにストリームにおける TRS 信号の存在または不在を使用します。シングルの TRS はレシーバが有効な SDI サンプルを収集していることをトランシーバ・コントロール・ステート・マシンに示します。ステート・マシンが指定したスレッショルドを超える任意の TRS シーケンスを検出しない場合にのみ、この信号がデアサートされます。トランシーバをリセットせずに、このフライホイール・ビデオ・デコーダは不在 EAV を許容します。システム・トップレベル・ファイル (tr_sdi.v) における TOTAL_CONSECUTIVE_MISSING_EAV パラメータを変更することによって、スレッショルドを設定できます。

```
// This parameter allows the missing EAVs to be tolerated
parameter [7:0] TOTAL_CONSECUTIVE_MISSING_EAV = 8'd4;
```

テスト・コントロール・ブロックの FSM は IDLE ステートで開始されます。タイムアウトの後に、FSM は SD-SDI 出力を送信するためにパターン・ジェネレータおよび SDI トランスミッタをコンフィギュレーションすることによって、テストを始めます。SDI トップレベル・ブロックは受信ストリームに認識およびロックした後、そしてダウンストリーム・ロジック・レシーバ・フォーマット・バリデータ (RX_FORMAT_VALIDATE) およびレシーバ・フォーマット・チェッカー (RX_FORMAT_CHECKER) のブロックは SDI トップレベル・ブロックによって生成されたフレームのフォーマットを検出して確認します；FSM は LOCKED_SD ステートに移動します。

1-18 ページの図 9 に、フライホイール・ビデオ・デコーダがロックされ、受信入力ストリームに同期されるときタイミング図を示します。use_fly 信号はフライホイール・ビデオ・デコーダが入力ストリームにある TRS シンボルを内部的に生成される TRS シンボルに置き換えることを示します；プロセスにおけるノイズで壊されるどんな TRS シンボルも修理します。

図 9. 受信入力カストリームに同期およびロックしたフライホイール

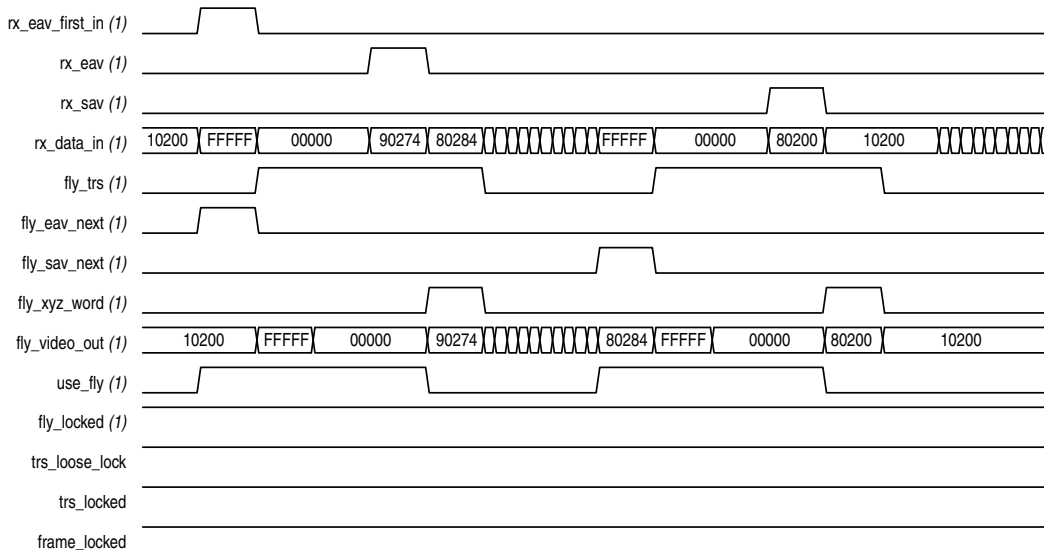


図 9 の注 :

(1) エンベデッド SDI フライホイール・ブロックの入力および出力ポート。

表 3 に、追加テスト・ステータスを記録するのに FSM で使用される 13 ビット・レジスタ (result_reg) を示しています。

表 3. 13 ビット・レジスタ

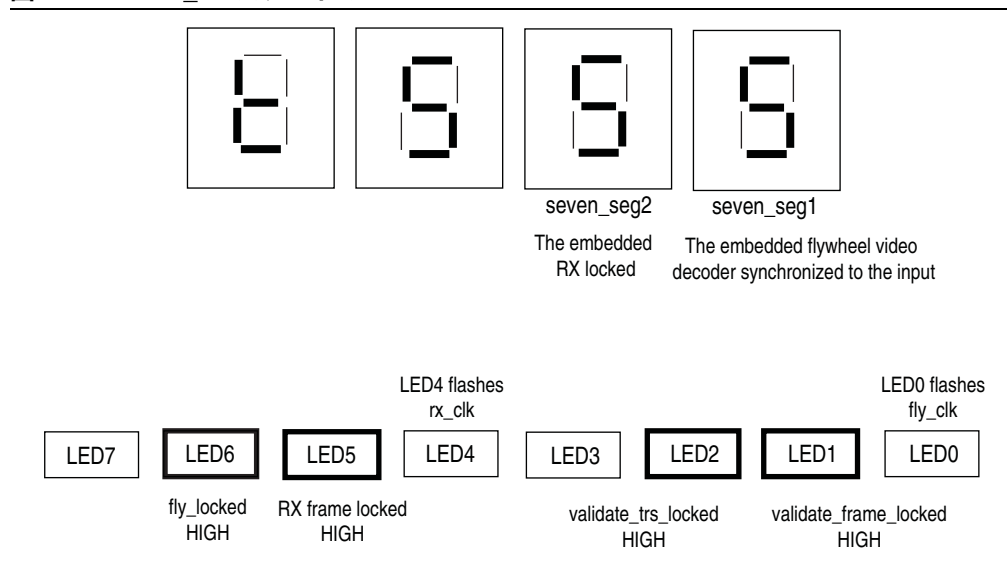
ビット	説明
0	SD-SDI 向け。SDI トップレベル・ブロックによって生成された SD RX フォーマットは TX フォーマットと一致することを示します。
1	SD-SDI 向け。入力を取り除かれている場合でも、SDI トップレベル・ブロックは以前の SD タイミングで生成し続けることを示します。
2	SD-SDI 向け。同じ規格および同じフォーマット間で切り替えを行う場合、ゼロ割り込みがロックされた信号 (例えば、fly_locked 信号はアサートされたままで) に対して SDI トップレベル・ブロックで検出されることを示します。
3	HD-SDI 向け。SDI トップレベル・ブロックによって生成された HD RX フォーマットは TX フォーマットと一致することを示します。
4	HD-SDI 向け。入力を取り除かれている場合でも、SDI トップレベル・ブロックは以前の HD タイミングで生成し続けることを示します。
5	HD-SDI 向け。同じ規格および同じフォーマット間で切り替えを行う場合、ゼロ割り込みがロックされた信号 (例えば、fly_locked 信号はアサートされたままで) の SDI トップレベル・ブロックで検出されることを示します。
6	3G-SDI 向け。SDI トップレベル・ブロックによって生成された 3G RX フォーマットは TX フォーマットと一致することを示します。
7	3G-SDI 向け。入力を取り除かれている場合でも、SDI トップレベル・ブロックは以前の 3G タイミングで生成し続けることを示します。

表 3. 13 ビット・レジスタ

ビット	説明
8	3G-SDI 向け。同じ規格および同じフォーマット間で切り替えを行う場合、ゼロ割り込みがロックされた信号（例えば、fly_locked 信号はアサートされたままで）において SDI トップレベル・ブロックで検出されることを示します。
9	HD-SDI 向け。デッドタイム期間中に、ゼロ割り込み（fly_locked および frame_locked 信号がアサートされたままで）が検出されることを示します。
10	HD-SDI 向け。同期スイッチングの後に、ゼロ割り込み（fly_locked および frame_locked 信号がアサートされたままで）が検出されることを示します。
11	3G-SDI 向け。デッドタイム期間中に、ゼロ割り込み（fly_locked および frame_locked 信号がアサートされたままで）が検出されることを示します。
12	3G-SDI 向け。同期スイッチングの後に、ゼロ割り込み（fly_locked および frame_locked 信号がアサートされたままで）が検出されることを示します。

図 10 に、FSM が LOCKED_SD ステートにあるとき、ボードが表すものを示していません。

図 10. LOCKED_SD ステートの FSM



ケーブルを取り外すまで、FSM は LOCKED_SD ステート状態を維持します。そして、FSM は REMOVE_SD ステートに移動して、ケーブルを再接続するのを待っています。1-20 ページの図 11 に、入力を取り除くとき、ボードが表すものを示しています。

図 11. REMOVE_SD ステートの FSM

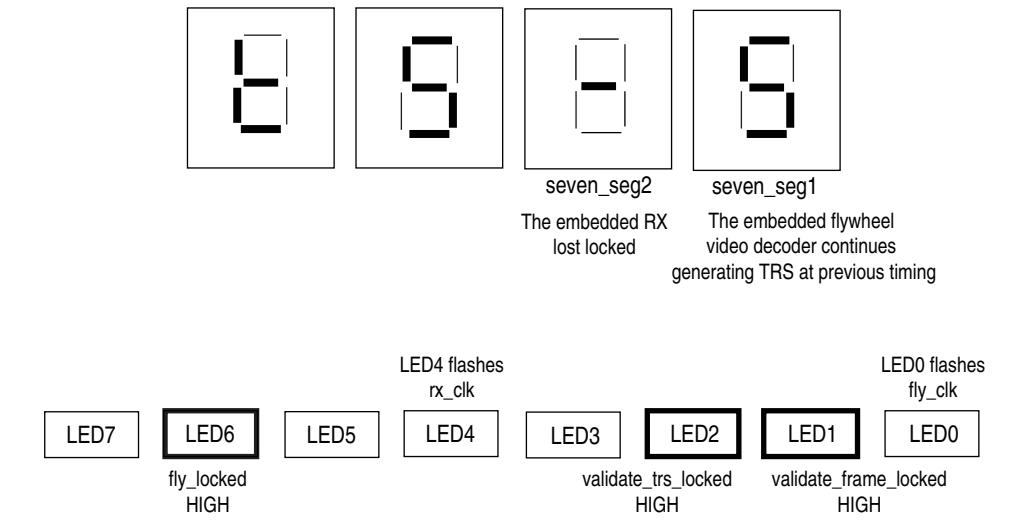


図 12 のタイミング図で、入力ストリームが中断または削除される時、フライホイール・ビデオ・デコーダが入力ストリームに TRS シンボルを継続的に生成と挿入する方法を示しています。

図 12. TRS シンボルを生成し続けるフライホイール・ビデオ・デコーダ

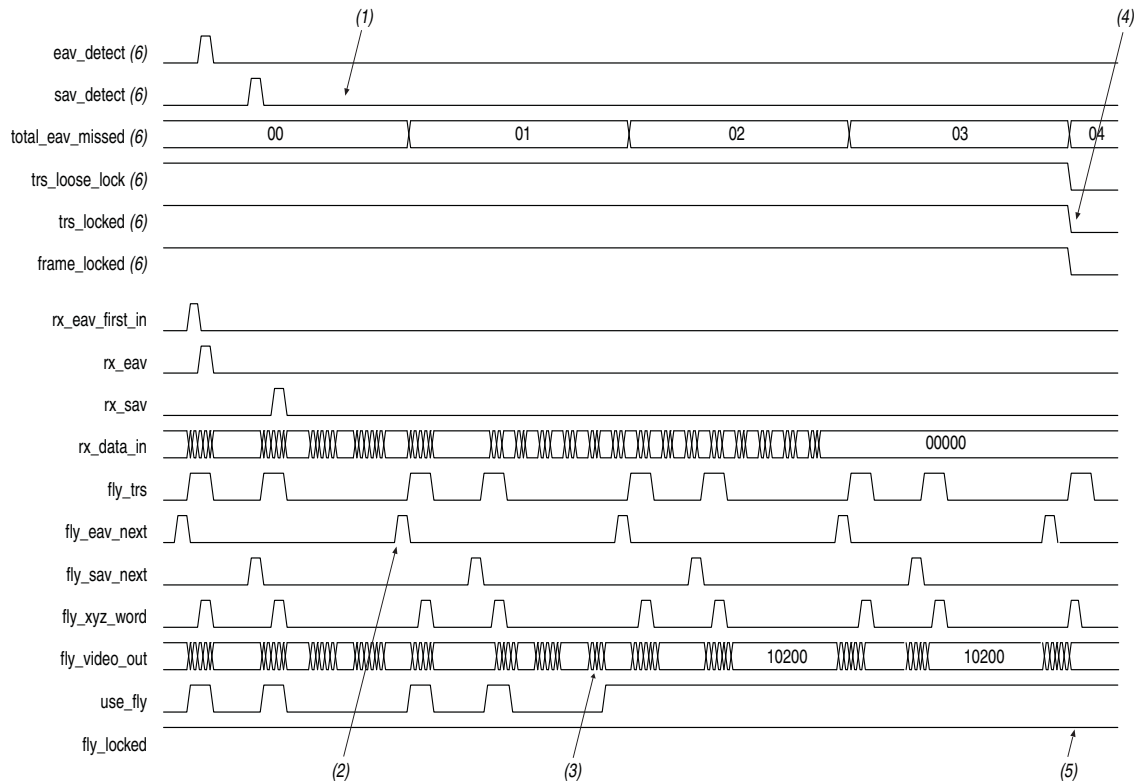


図 12 の注：

- (1) 入力ビデオが取り除かれます。
- (2) フライホイール・ビデオ・デコーダは以前のタイミングで TRS シンボルを生成し続けます。現在、得られたビデオ・ストリームにおける視覚的な情報は無効になりますが、TRS シンボルにおけるタイミング情報は有効になります。
- (3) 次の EAV で、受信ストリーム (use_fly 信号はアサートされたままで) を使用する代わりに、フライホイール・ビデオ・デコーダはブラックレベル・ビデオ・データを生成し開始します。
- (4) 4 つ EAV の連続不在が検出される際に、エンベデッド・フォーマット検出ブロックのロックされた信号がディアサートされます。このリファレンス・デザインでは、TOTAL_CONSECUTIVE_MISSING_EAV を 8'd4 に設定します。
- (5) fly_locked 信号はアサートされたままで、フライホイール・ビデオ・デコーダが以前のタイミングでブラックレベル・ビデオ・データを生成し続けます。
- (6) エンベデットされた SDI フォーマット検出ブロックからの信号です。

ケーブルを再接続した後に、エンベデッド・フライホイール・ビデオ・デコーダは新しい受信入力に再同期されます。FSM は START_SD_AGAIN ステートに移動します。現在、図 10 に示すようにボードが出力を表示します。同じ規格および同じフォーマットで入力を取り除くおよび再接続するとき、フライホイール・ビデオ・デコーダは fly_locked 信号をディアサートしません。この期間中に、FSM は fly_locked 信号を継続的にモニタします。

図 13 のタイミング図で、フライホイール・ビデオ・デコーダは同じビデオ規格およびフォーマット間で切り替えるとき、新しい受信入力に同期される方法を示します。

図 13. フライホイール・ビデオ・デコーダの同期化 — 同じビデオ規格とフォーマット間のスイッチング

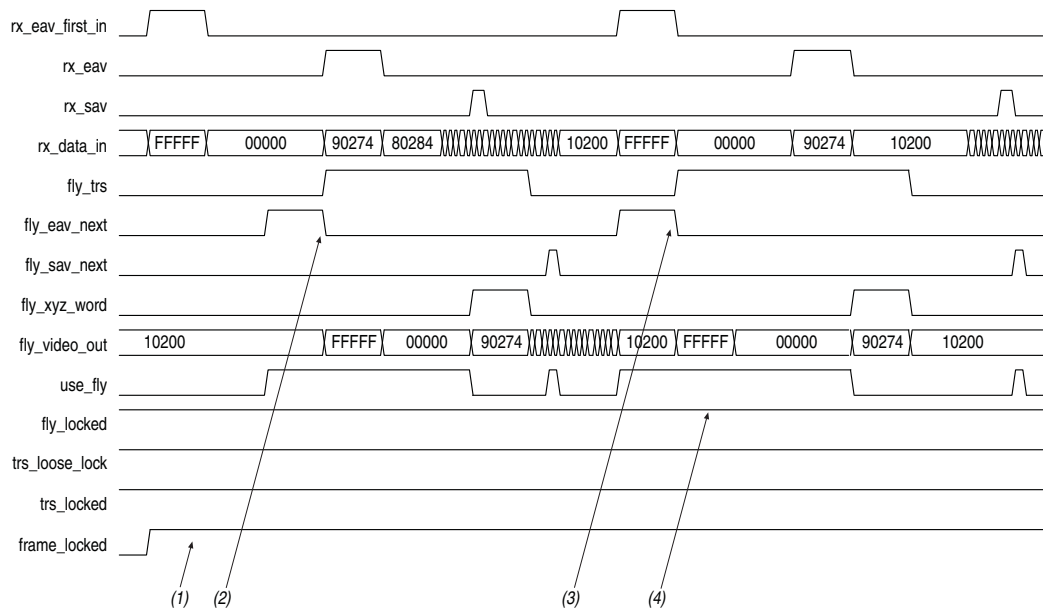


図 13 の注：

- (1) エンベデッド・フォーマット検出ブロックによりラインまたはフレームの数に対する一定の同期パルス (F、V、H) が検出されると、`frame_locked` がアサートされて、フライホイール・コントロール・ステート・マシンは新しい受信入力ストリームに再同期し開始します。
- (2) フライホイール・ビデオ・デコーダが新しい水平・タイミングに同期する前に、以前の水平・タイミングで発生し続けています。
- (3) フライホイール・ビデオ・デコーダはすぐに、次の EAV で新しい水平・タイミングに同期します。
- (4) 受信ビデオ・フォーマットで変更がない場合、`fly_locked` 信号は High になったままです。

この同期化ステートまで、SDI トップレベル・ブロックは受信入力に同期し、入力がか断または削除されるときに TRS を生成して、同じビデオ規格およびフォーマット間のスイッチングにゼロ割り込み (`fly_locked` 信号はアサートされたまま) を生成します。

次に、指定された時間の後に `START_HD` ステートへの FSM 転送です。START_HD ステートは HD-SDI テスト・パターンを送信するために、パターン・ジェネレータおよび SDI TX をコンフィギュレーションします。ビデオ・フォーマットにおける変更がある場合、FSM は SDI トップレベル・ブロックによってアサートされた `format_change` 信号を検出します。

図 14 のタイミング図で、フライホイール・ビデオ・デコーダは異なるビデオ規格およびフォーマット間で切り替えるとき、新しい受信入力に同期される方法を示します。

図 14. フライホイールの同期化 — 異なるビデオ規格とフォーマットでのスイッチング

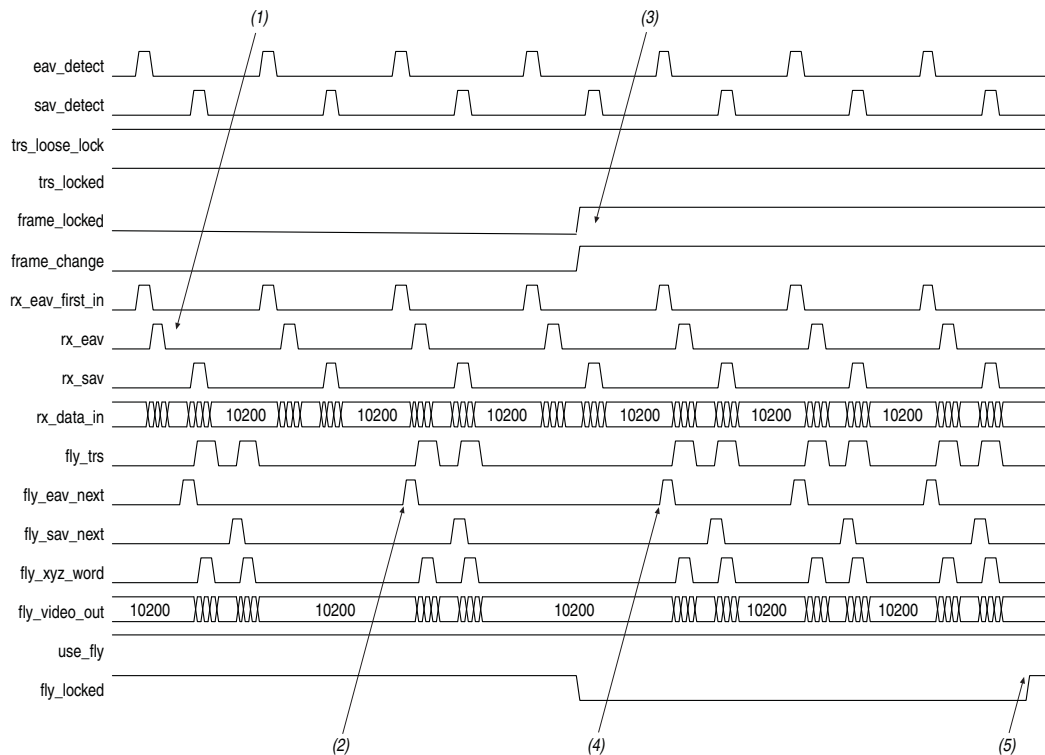


図 14 の注：

- (1) 3G レートでの新しい受信入力です。
- (2) フライホイール・ビデオ・デコーダは以前の HD レートでブラックレベル・ビデオ・データをまだ生成しています。
- (3) フォーマット検出ブロックは連続有効なフレームを検出し、そしてフライホイール・ビデオ・デコーダに対して再同期をするために frame_locked 信号をアサートします。受信ビデオ・フォーマットが変更するとき、fly_locked 信号がデアサートされます。
- (4) フライホイール・ビデオ・デコーダは次の受信 EAV で受信 3G 入力に同期します。
- (5) フライホイール・ビデオ・デコーダは受信 3G 入力に同期してロックします。

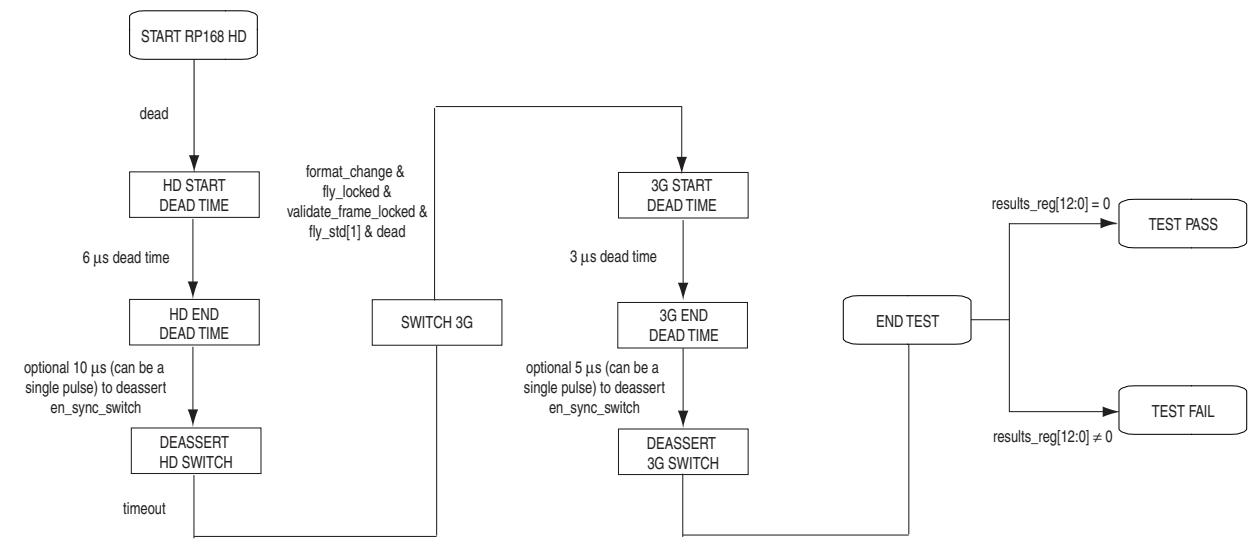
SDI トップレベル・ブロックは HD 入力にロックし、送信フォーマットで生成されたフォーマットを検証すると、FSM は異なるビデオ規格とフォーマット (例えば、SD PAL から HD 1080p30 まで) がある非同期スイッチングが起こるのを示す LOCKED_HD ステートに移動します。ケーブルを取り外すまで、FSM はこのステートの状態を維持します。そして、FSM は REMOVE_HD ステートに移動して、ケーブルを再接続するのを待っています。

FSM は 3G-SDI (1-16 ページの図 8 を参照) に対して同じテスト・シーケンスを繰り返します。3G-SDI テスト・ケースのケーブルを取り除きおよび再接続すると、FSM が同期スイッチング・ループ (1-24 ページの図 15 を参照) に入ります。スイッチング・ラインの間、このループは同じビデオ・フォーマットを切り替えることによる RP168 機能をテストします。完全に非破壊的な状態で切り替えをして、ダウンストリーム・ユーザー・ロジックはこの変化を意識していません。fly_locked および frame_locked 信号は同期スイッチングの実行中と実行後にアサート状態を維持します。

同期スイッチング・ループ

1-24 ページの図 15 に、FSM における同期スイッチング・ループを示します。

図 15. 同期スイッチング・ループの図

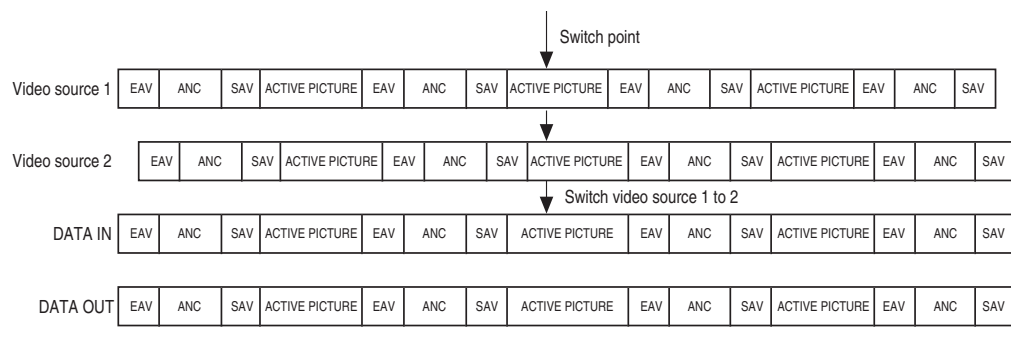


送信 HD-SDI と 3G-SDI を通してサイクルする同期スイッチング・ループは非同期スイッチング・ループに似ています。FSM はデッドタイム期間を開始して、HD_START_DEAD_TIME ステートに移動します。デッドタイム期間中に、FSM は SDI トリプル規格トランスミッタ・ブロックを制御して、レシーバ側にある信号損失の状況を模擬するのにすべてゼロを送信します。

HD 1080p30 に対して、デッドタイム期間は 6μs であり、信号損失がライン 7 で行います。そして、スイッチング領域は 625 と 1,070 ピクセルの間です。

6 μs の後、FSM は HD_END_DEAD_TIME ステートに移動して、突然スイッチングを実行します。パターン・ジェネレータからの TX データの遅延バージョンを使用するために、SDI トリプル規格トランスミッタ・プロトコル・ブロックがマシンで制御されます。SDI トップレベル・ブロックはデッドタイム期間後で入力ストリームの遅延バージョンを受信します。ビデオ・ソース 1 からビデオ・ソース 2 への切り替えのシナリオが FSM で模擬されます。これで、図 16 に示すようにビデオ・ソース 2 のストリームはいくつかのクロック・サイクルで遅れます。また、FSM はトップレベル・ブロックのための en_sync_switch パルスも生成します。

図 16. ビデオ・ソース 1 からビデオ・ソース 2 へのスイッチング



スイッチングはビデオ・ソース 1 からビデオ・ソース 2 に行います。ここで、ビデオ・ソース 2 の rx_eav 信号は fly_eav 信号の後にアサートされます。図 17 のタイミング図に、スイッチングの実行方法を示します。

図 17. ビデオ・ソース 1 からビデオ・ソース 2 への HD 同期スイッチング

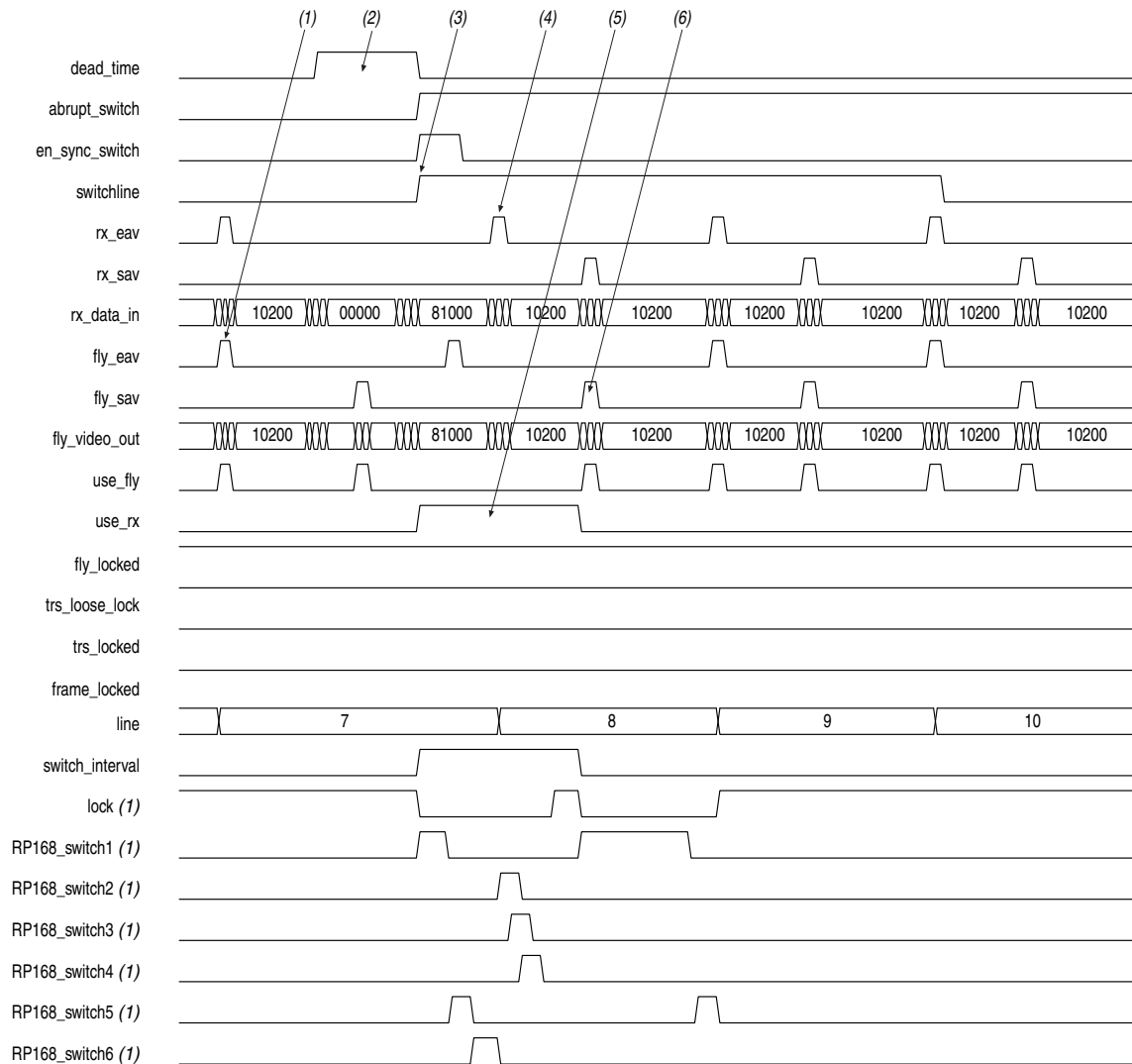


図 17 の注：

- (1) フライホイール・ビデオ・デコーダは受信 HD 入力にロックします。
- (2) HD の場合には 6 us デッドタイム期間です。すべてのゼロはレシーバによって受け取られますが、フライホイール・ビデオ・デコーダは、TRS タイミングを生成し続けています。
- (3) デッドタイム期間の終わりおよびビデオ・スイッチ・ポイントです。ロックされた信号なしでフライホイール・ビデオ・デコーダをすぐに新しい TRS アラインメントに再アラインメントするように、フライホイール・ビデオ・デコーダはアサートされた en_sync_switch パルスを 3 つのライン（スイッチライン）にラッチします。
- (4) 新しい EAV は fly_eav の後にアサートされます。水平・タイミングおよびストリームのアラインメントの外乱です。
- (5) 同期スイッチング・ライン（スイッチ・インターバルが High のとき）間に、フライホイール・ビデオ・デコーダは EAV を入力ビデオ・ストリームから出力ビデオ・ストリームに直接渡すように設計されています。
- (6) フライホイール・ビデオ・デコーダはすぐに、次の SAV で新しい水平・タイミングに再同期されます。

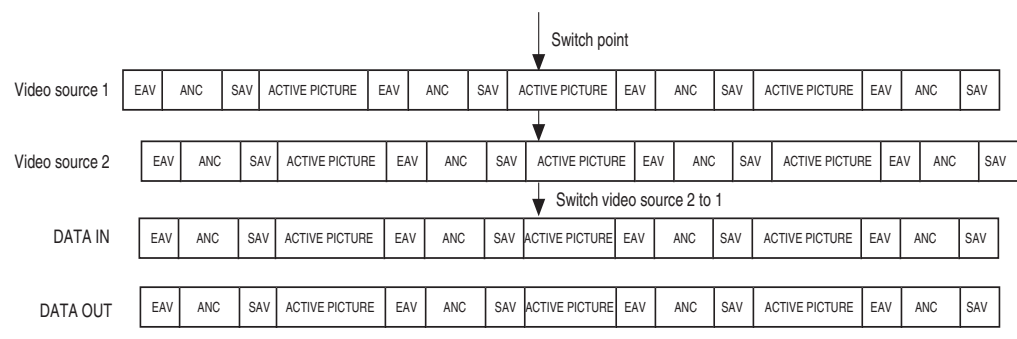
SWITCH_3G ステートに移動する前に、FSM は同期スイッチングの実行中または実行後に fly_locked および frame_locked 信号を継続的にモニタします。

次に、FSM は SWITCH_3G ステートに移動します。SDI トップレベル・ブロックは 3G 入力にロックし、送信フォーマットで生成されたフォーマットを検証すると、マシンはデッドタイム期間を開始します。FSM は SDI トリプル規格トランスミッタ・ブロックを制御して、レシーバ側にある信号損失の状況を模擬するのにすべてゼロを送信します。

3G 1080p50 に対して、デッドタイム期間は 3 μ s であり、信号損失がライン 7 で行います。そして、スイッチング領域は 625 と 1,070 ピクセルの間です。

3 μ s の後、FSM は END_3G_DEAD_TIME ステートに移動して、突然スイッチングを実行します。FSM は、ビデオ・ソース 2 からビデオ・ソース 1 へのスイッチングのシナリオを模倣することによって、次の EAV を以前のタイミングより早く発生させるようにパターン・ジェネレータを制御します。ここで、図 18 に示すとおり、ビデオ・ソース 2 はいくつかのクロック・サイクルで遅れます。

図 18. ビデオ・ソース 2 からビデオ・ソース 1 へのスイッチング



スイッチングはビデオ・ソース 2 からビデオ・ソース 1 に行います。ここで、ビデオ・ソース 1 の rx_eav 信号は fly_eav 信号より早くアサートされます。

1-27 ページの図 19 のタイミング図に、スイッチングの実行方法を示します。

図 19. ビデオ・ソース 2 からビデオ・ソース 1 への 3G 同期スイッチング

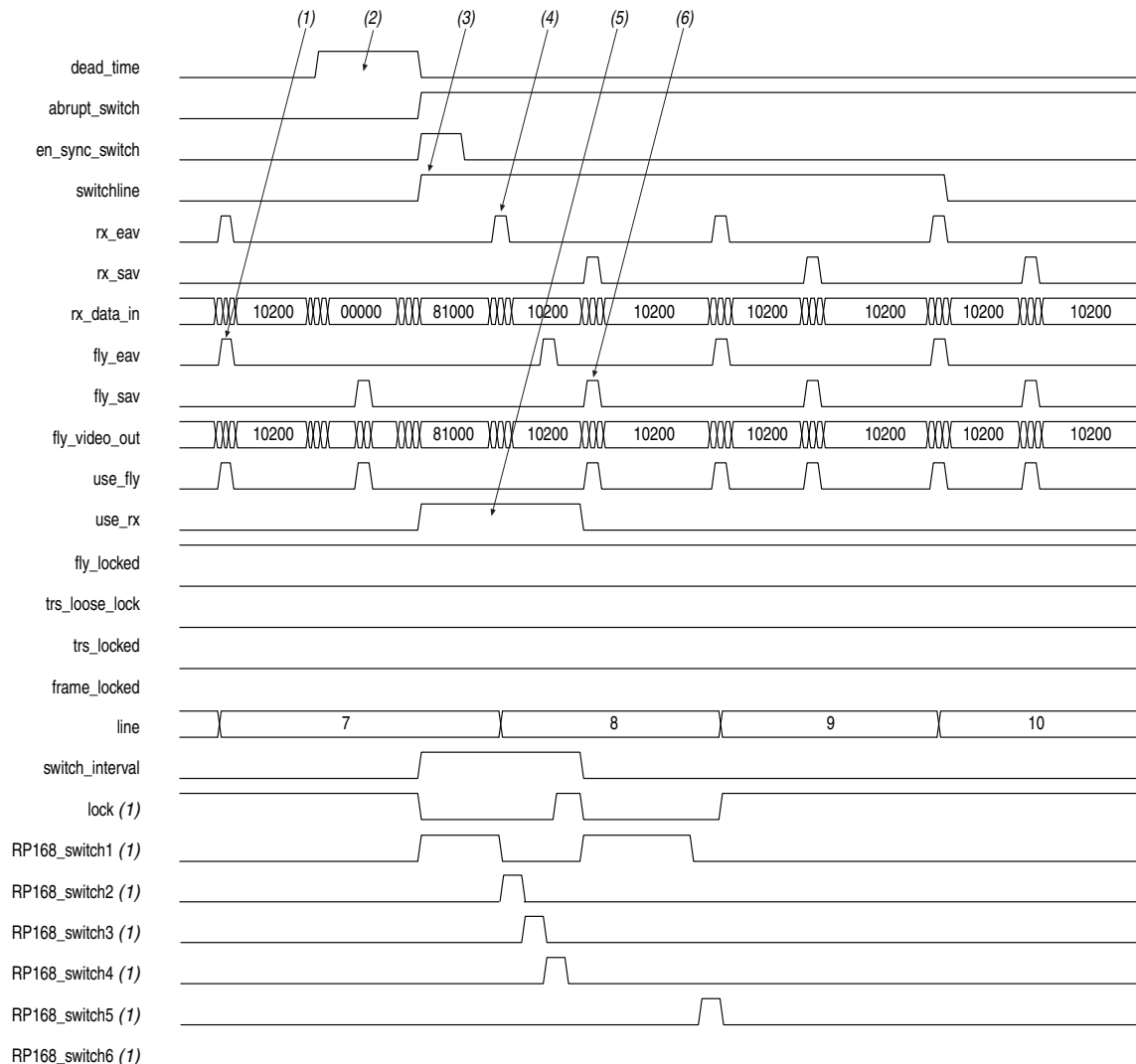


図 19 の注：

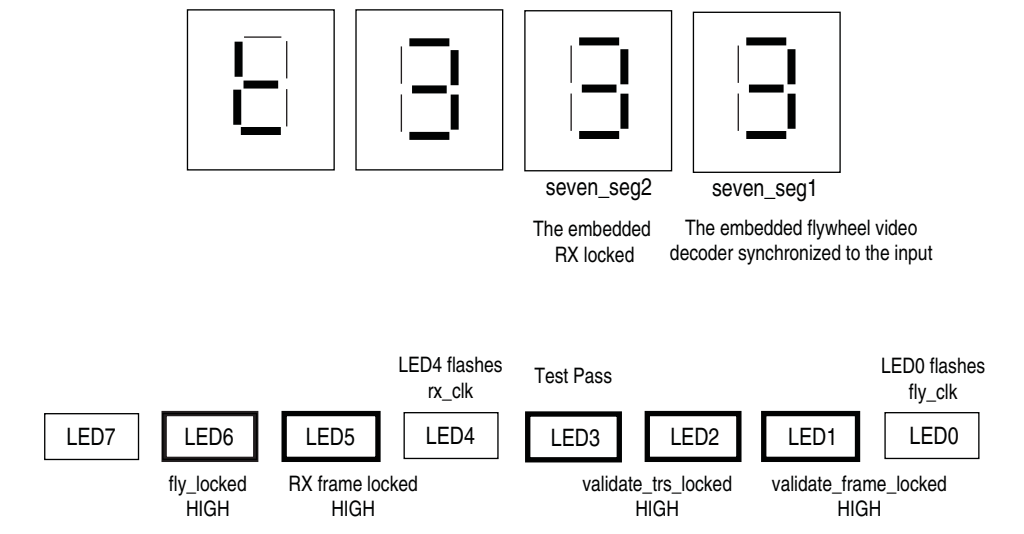
- (1) フライホイール・ビデオ・デコーダは受信 3G 入力にロックします。
- (2) 3G の場合には 3 us デッドタイム期間です。すべてのゼロがレシーバによって受け取られますが、フライホイール・ビデオ・デコーダは、TRS タイミングを生成し続けています。
- (3) ビデオ・スイッチ・ポイントおよびデッド期間の終了。ロックされた信号なしでフライホイール・ビデオ・デコーダをすぐに新しい TRS アラインメントに再アラインメントするように、フライホイール・ビデオ・デコーダは en_sync_switch パルスを 3 つのライン（スイッチライン）にラッチします。
- (4) 新しい EAV は fly_eav 信号より早くアサートされます。水平同期・タイミングおよびストリームのアラインメントの外乱です。
- (5) 同期スイッチング・ライン（スイッチ・インターバルが High のとき）間に、フライホイール・ビデオ・デコーダは EAV を入力ビデオ・ストリームから出力ビデオ・ストリームに直接渡すように設計されています。
- (6) フライホイール・ビデオ・デコーダはすぐに、次の SAV で新しい水平同期・タイミングに再同期されます。

また、マシンは SDI トップレベル・ブロックに対する必要な en_sync_switch パルスを生成します。再び、FSM は同期スイッチング中および同期スイッチング後に、fly_locked および frame_locked 信号を継続的にモニタします。

同期スイッチング・ループは、スイッチが同期スイッチングによる水平障害を検出した直後に、SDI トップレベル・ブロックが水平・タイミングを再同期化するかどうかをチェックします。

最後に、すべてのテストのステータスをチェックすることによってプロセスを終了します。図 20 に示すようにすべてのテストが合格する場合、LED3 を点灯させ、そして 1 つ以上のテストが失敗する場合、LED7 を点灯させます。

図 20. すべてのテストが合格されているテストの終わり



ローカル PLL

ローカル PLL ブロック (FLY_CLK_GEN) は SDI トップレベル・ブロックに対して、74.25 MHz (HD-SDI) および 148.5 MHz (SD-SDI と 3G-SDI) を供給します。SDI トップレベル・ブロックはレシーバが入力ビデオ・ストリームにロックされる時のみこのローカル・クロックを使用します。

リコンフィギュレーション・コントロール・ロジック

リコンフィギュレーション・コントロール・ロジック・ブロック (ALT2GXB_RECONFIG) は SDI トップレベル・ブロックのレシーバ部分のリコンフィギュレーションを処理します。

レシーバ・フォーマット・バリデータ

レシーバ・フォーマット・バリデータ (RX_FORMAT_VALIDATE) は SDI トップレベル・ブロックにエンベデッドされたフォーマット検出ブロックと同じです。レシーバ・フォーマット・バリデータは SDI トップレベル・ブロックによって生成される SDI ストリームのラインおよびフレーム・タイミングをモニタします。受信ストリームがロックされるか、またはロックが解除されるかを示すために様々なロックされた信号を生成します。また、レシーバ・フォーマット・チェッカ・ブロックに対するフレーム・フォーマット情報もデコードします。

レシーバ・フォーマット・チェッカ

レシーバ・フォーマット・チェッカ・ブロック (RX_FORMAT_CHECKER) により、レシーバ・フォーマット・バリデータ・ブロックでデコードされたラインおよびフレーム・タイミングに基づく SDI トップレベル・ブロックによって生成される SDI ストリームのビデオ・フォーマットが決まります。それは受信ビデオ・フォーマットを TX フォーマットと比較します。

リファレンス・デザインのデモンストレーション

リファレンス・デザインを実行するには、以下の手順を実行します。


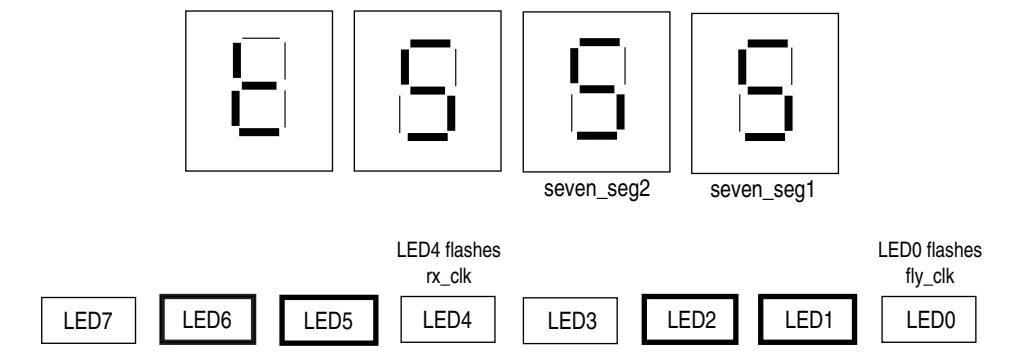

1. ボード接続を設定します。テスト・パターン・トランスミッタ出力、SDI_OUT_P2 (BNC J52) を SDI トップレベル・ブロックのレシーバ入力、SDI_IN0 (BNC J49) に接続します。
2. Quartus II ソフトウェアを起動します。
 - a. File メニューの **Open Project** をクリックし、/quartus/tr_sdi.qpf に移動して、**Open** をクリックします。
 - b. Processing メニューで、**Start Compilation** をクリックします。
3. Quartus II の生成された SRAM オブジェクト・ファイル (.sof)、/quartus/tr_sdi.sof をダウンロードします。
4. LED が  に示すとおりに表示されると、BNC J49 コネクタからケーブルを取り外します。

図 21. SD-SDI にロックした場合



 始動してから 3 秒以降にのみ、ケーブルを取り外します。


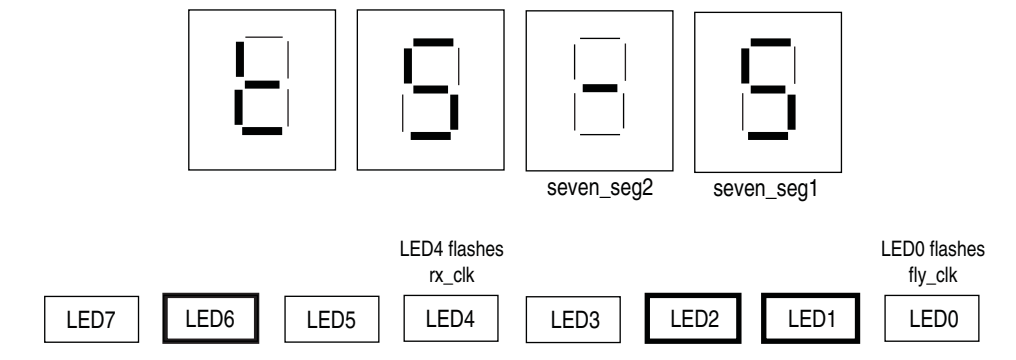
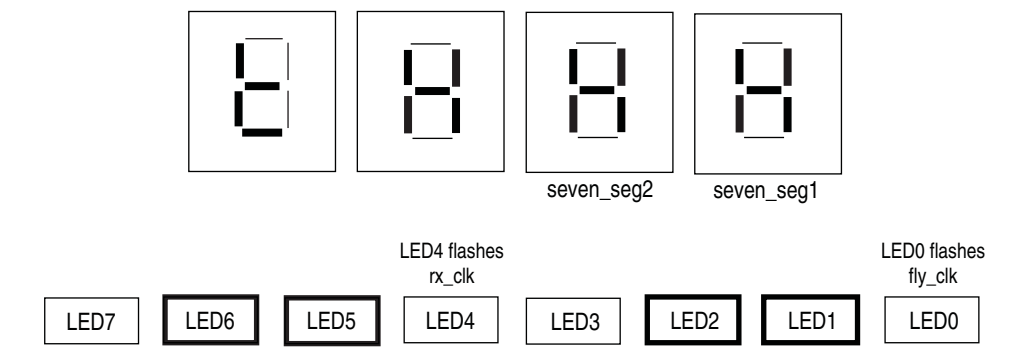
5. BNC J49 コネクタからケーブルを取り外します。LED が  に示すとおりに表示されるはずですが。


図 22. 以前の SD レートでデータを生成し続ける SDI トップレベル・ブロック



6. ケーブルを BNC J49 コネクタに再接続して、LED が図 21 に示すように表示されるまでを待ってください。
7. 数秒後に、テストがトランスミット HD-SDI を通じてサイクルします。LED が図 23 に示すとおりに表示されるはずですが。

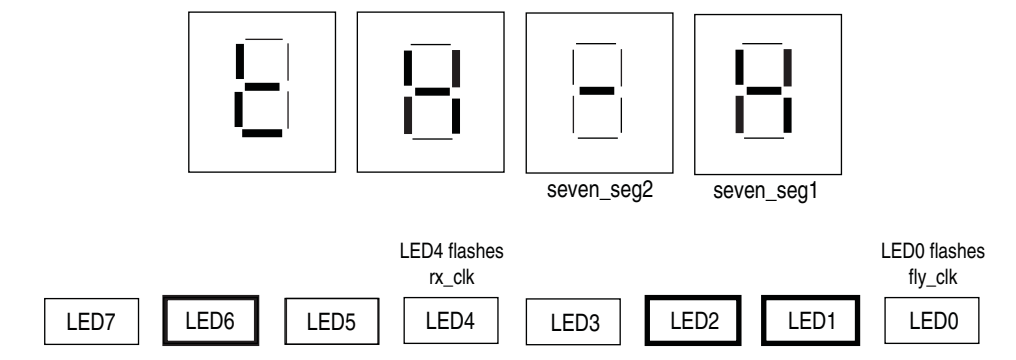
図 23. HD-SDI にロックした場合



 LED が図 23 に示すように表示されない場合、ケーブルを取り外して、再接続します。

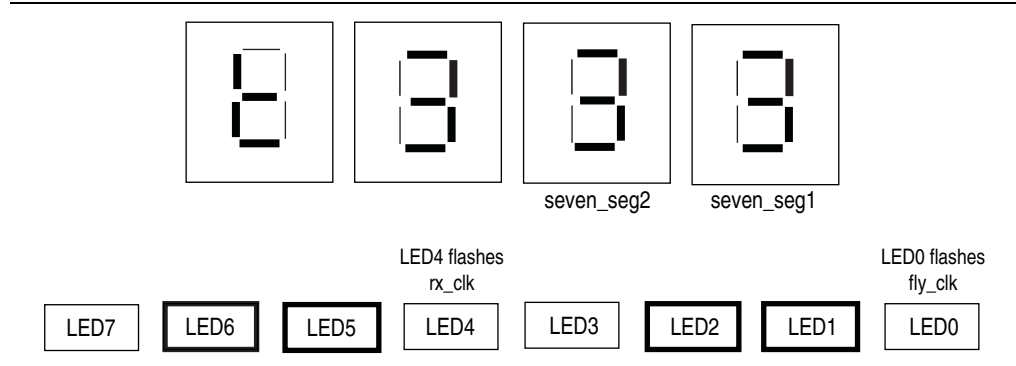
8. BNC J49 コネクタからケーブルを取り外して、LED が図 24 に示すように表示されるまでを待ってください。

図 24. 以前の HD レートでデータを生成し続ける SDI トップレベル・ブロック



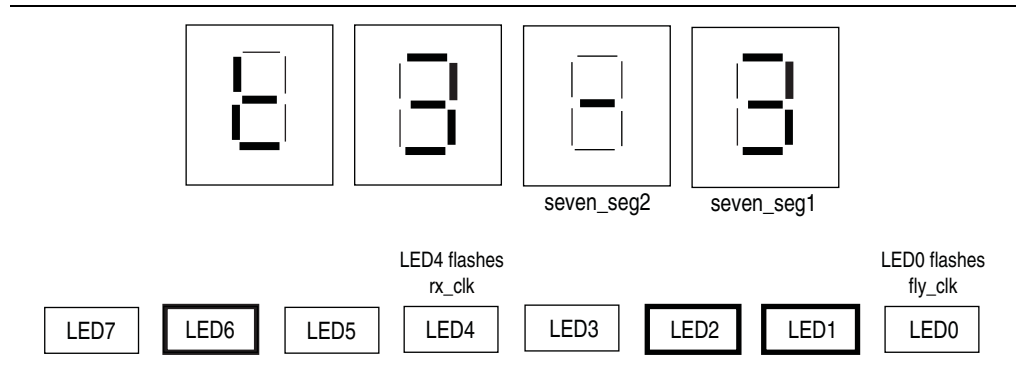
9. 次に、ケーブルを BNC J49 コネクタに再接続して、LED が再び図 23 に示すように表示されるまでを待ってください。
10. 数秒後に、テストがトランスミット 3G-SDI を通じてサイクルします。現在、LED が図 25 に示すとおりに表示されるはずですが。

図 25. 3G-SDI にロックした場合



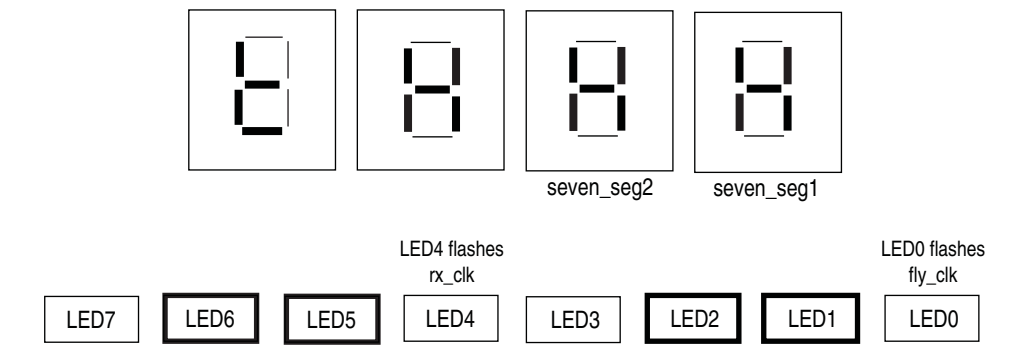
11. BNC J49 コネクタからケーブルを取り外して、LED が図 26 に示すように表示されるはずですが。

図 26. 以前の 3G レートでデータを生成し続ける SDI トップレベル・ブロック



12. 次に、ケーブルを BNC J49 コネクタに再接続して、LED が再び図 25 に示すように表示されるまでを待ってください。
13. 数秒後で、テストは最初の送信 HD-SDI によって同期スイッチング・ループ (RP168) に入ります。現在、LED が図 27 に示すとおりに表示されるはずですが。

図 27. (RP 168) HD-SDI にロックした場合




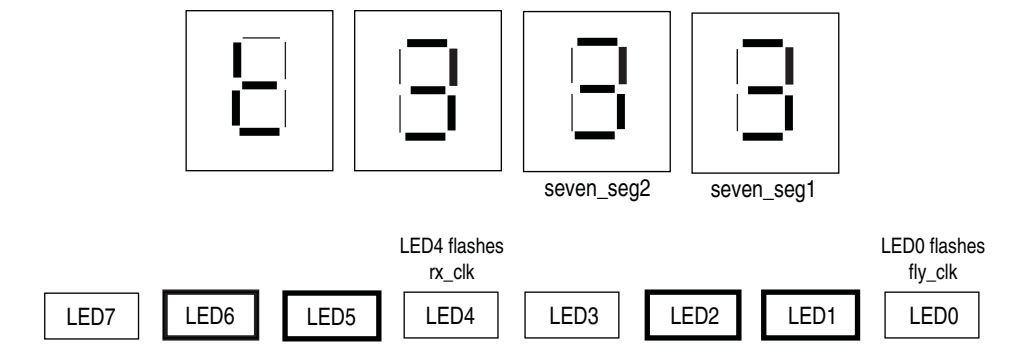
14. テストが同期スイッチングを自動的に実行します。SDI トップレベル・ブロックからの `fly_locked` および `frame_locked` 信号は同期スイッチング中および同期スイッチングの後に継続的にモニタされます。テストの終わりで、テスト・ステータスがレコードされて、チェックされます。
15. 3G-SDI を送信し続けて、LED が  28 に示すように表示されます。

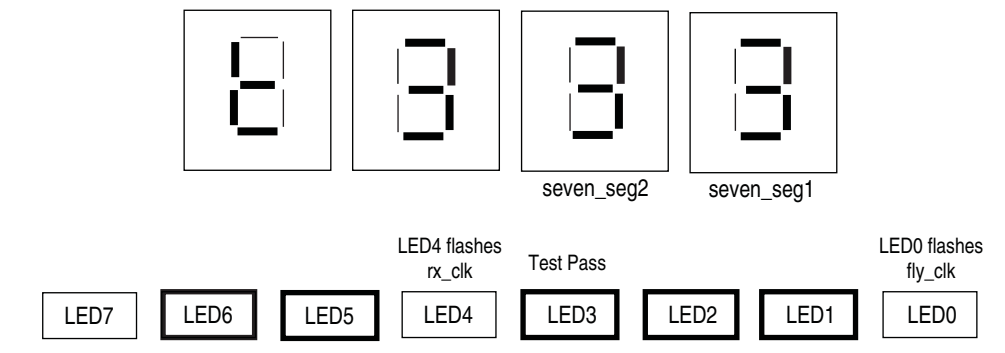
図 28. (RP 168) 3G-SDI にロックされる



16. 同様に、テストが同期スイッチングを自動的に実行します。SDI トップレベル・ブロックからの `fly_locked` および `frame_locked` 信号は同期スイッチング中および同期スイッチングの後に継続的にモニタされます。テストが完了したら、テスト・ステータスがレコードされて、チェックされます。

17. テストが成功すると、LED が図 29 に示すように表示されます。

図 29. テスト・パス



改訂履歴

表 4 に、このアプリケーション・ノートの改訂履歴を示します。

表 4. 改訂履歴

日付およびリビジョン	変更内容	概要
2009 年 4 月、 バージョン 1.0	初版。	—