

Application Note

システム レベルの **ESD 耐性を目的とする FPD-Link ADAS 設計の最適化**



Casey McCrea, Taylor Gage

概要

FPD-Link カメラの設計は多くの場合、OEM 認定において厳格な車載 EMC テストを受けています。自動車内のカメラシステムが、車両の安全性アーキテクチャにおいて受け持つ責任が増え続けている現状で、その認定に対する堅牢性の要件もさらに厳格になってきています。多くの OEM のお客様は、伝導干渉、放射干渉、さらにはモジュール、コネクタ、ケーブルへの ESD 衝撃時など、さまざまな高ストレス電気的条件の下で、エラー フリーの性能を要求しています。このアプリケーション ノートでは、FPD-Link のシステム設計者がハードウェア設計とソフトウェア設計の両方に実装して、システム レベルの ESD 性能を最大限に高めることができる、設計に関する主なガイドラインについて概説しています。

目次

1 はじめに	2
2 代表的なテスト標準の概要	2
2.1 ISO 10605 標準.....	2
2.2 性能ステータスの分類.....	3
3 FPD-Link ハードウェアの最適化	4
3.1 コネクタの接地.....	4
3.2 PCB から筐体へのグラウンドィング.....	4
3.3 MODE の選択.....	5
4 FPD-Link ソフトウェアの最適化	7
4.1 ロック検出のチューニング.....	7
4.2 パリティ エラーの処理.....	8
4.3 前方誤り訂正.....	9
5 最適化のテストデータ	11
5.1 ベースライン ハードウェア - ソフトウェア最適化なし.....	11
5.2 最適化されたハードウェア - ソフトウェア最適化なし.....	12
5.3 最適化されたハードウェアとソフトウェア.....	12
6 ソフトウェア最適化のスクリプト例	13
7 システム レベルの追加ソフトウェア オプション	14
8 まとめ	15
9 参考資料	15
10 改訂履歴	16

商標

すべての商標は、それぞれの所有者に帰属します。

1 はじめに

システム レベルの ESD テストは、広範な全世界の車載 OEM に共通の要件です。システム レベルの ESD テストの目標は通常、モジュール内の電気部品が損傷を受けないことの確認と、ESD イベント発生時のアプリケーション レベルの性能を評価することの 2 つです。システム レベルの ESD テストに関する一般的な誤解は、アプリケーション レベルの性能が部品のデータシートに規定された ESD 定格 (例: IEC 6100-4-2, HBM, CDM など) に関係している可能性があるというものです。データシートに規定されている ESD 定格は、デバイスのピンにエネルギーが放出されたとき、IC が損傷せず耐える能力のみを示すものです。この定格は、主にチップの取り扱いと組み立ての要件を理解するために役立ちますが、重要な点として、データの消失などアプリケーション レベルの性能の側面が考慮されていません。EMI/ESD ストレス下での高速銅リンクのアプリケーション レベルの性能は、システム設計に大きく関係していることを理解してください。このガイドには、強力なハードウェア設計手法とソフトウェア設定の最適化によって耐性を最大化するための、TI による推奨事項が記載されています。

2 代表的なテスト標準の概要

2.1 ISO 10605 標準

ISO 10605 標準は、車載環境でのシステム レベルの ESD 性能を評価するため、最も一般的に参照されているテストセットアップの 1 つです。この標準は、車両の電気システムに入り込む ESD をシミュレートし、システム性能を評価するよう特別に調整されたものです。FPD-Link システムの場合、テストセットアップには通常、ECU からカメラ モジュールへのエンドツー エンドのビデオリンクに加えて、カメラ データを表示するためのライブ画面など、システム性能をリアルタイムで評価する手段が含まれます。

ISO 10605 のセットアップは、直接 (DUT に対して) または間接 (DUT 付近のカッピング プレーンに対して) の放電方法を使用して実行でき、通常はこれらの設定が OEM 要件によって定義されています。ESD エネルギーは、指定された RC ネットワークとチップ形状を備えた ESD ジェネレータ ガンを介してテストセットアップに放電されます。テストには、空中放電と接触放電の 2 種類の放電タイプが規定されています。この 2 つのうち、一般に結合エネルギーの点では接触放電のほうがシステムへのストレスが大きくなります。これは、衝撃時に空気のイオン化によって消費されるエネルギーが少ないといためです。次のセットアップ例は、ISO 10605 附属書 F に従って構成されたもので、FPD ケーブルの下にカッピング プレートが使用されています。

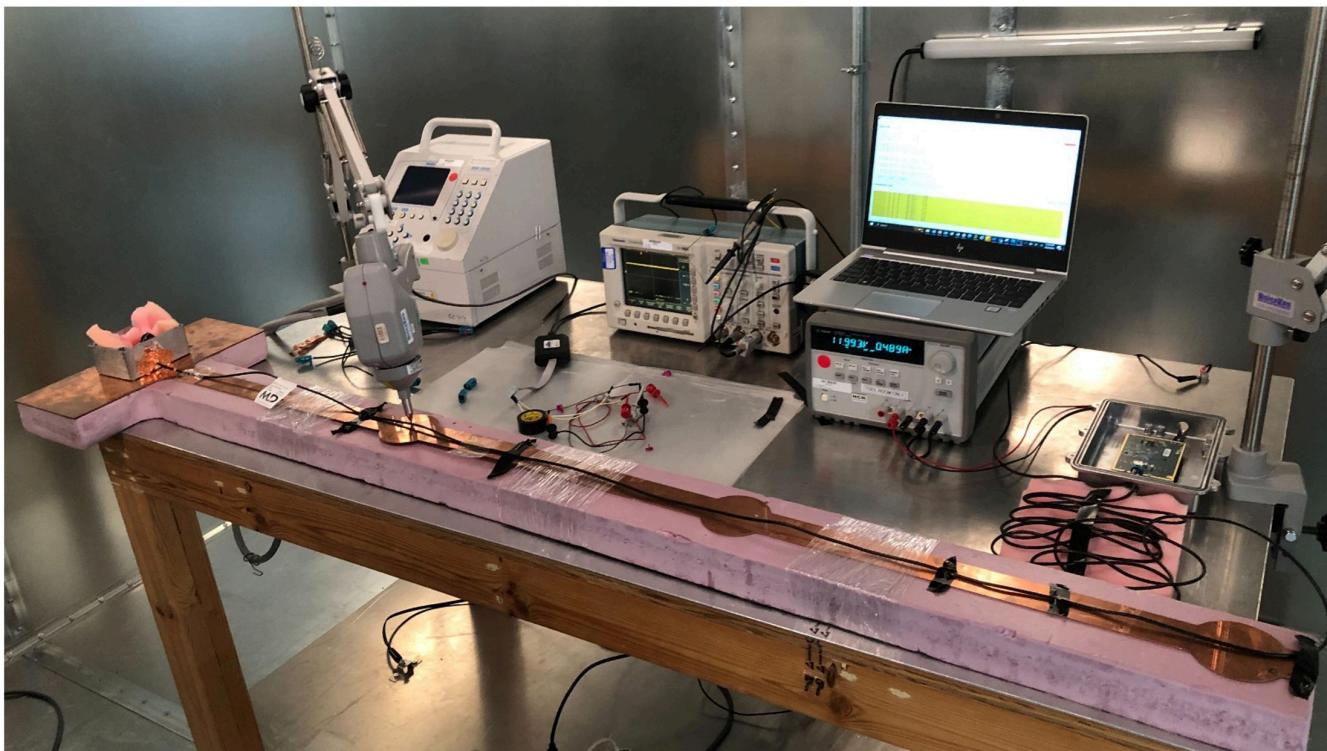


図 2-1. ISO 10605 附属書 F のセットアップ例

2番目の図は、プロセッサ プラットフォームのテスト ボード、完全なカメラ モジュール、および PC モニタ経由のライブビデオ表示を使用して、カメラ モジュールの性能を内部で評価するために TI で使用されるセットアップの概要です。このセットアップは、[セクション 5](#) でサンプル データを収集するために使用されました。デシリアルライザ システムには、一般的な ECU プラットフォームのような筐体は含まれていないことに留意してください。その結果、デシリアルライザ システムのグランディングとシールドは想定より弱くなり、平均性能が低下します。

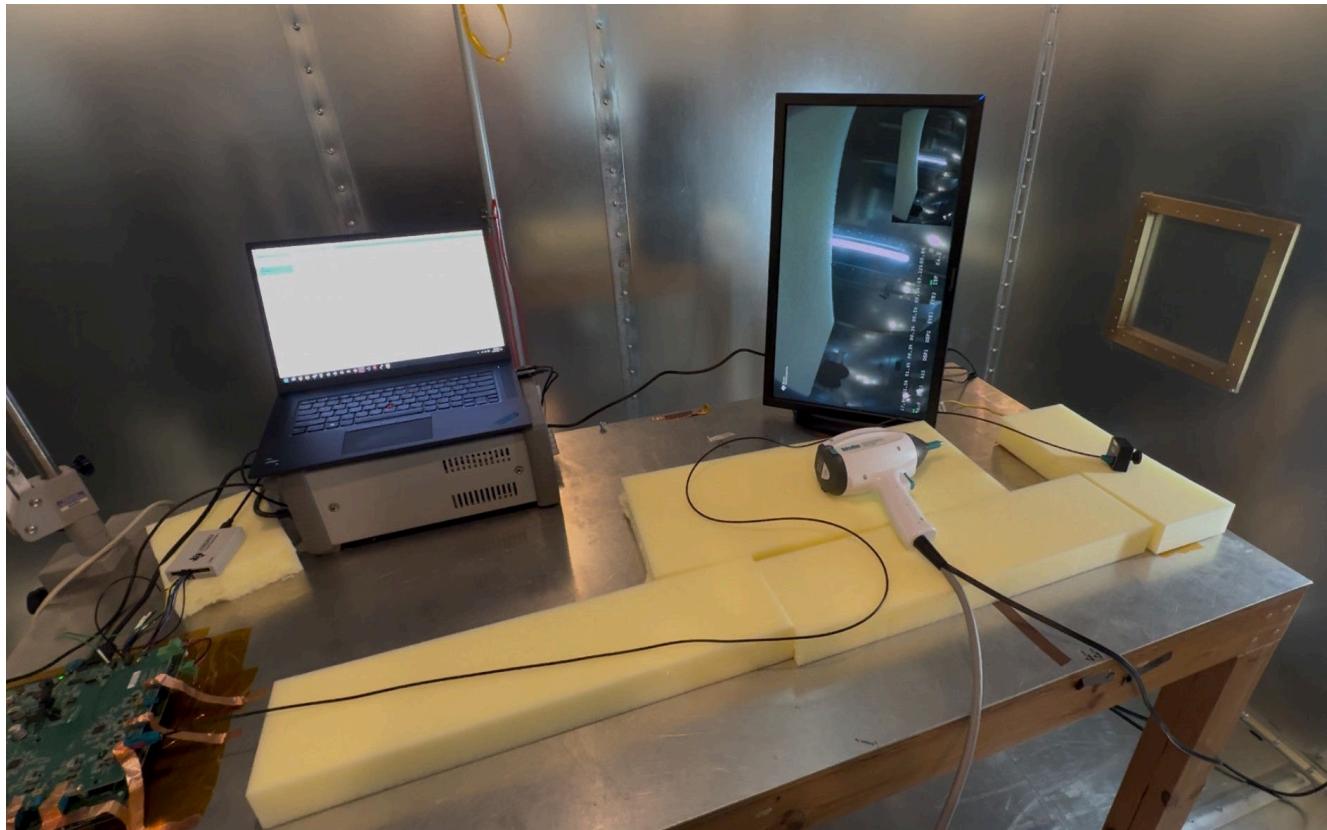


図 2-2. ISO 10605 のセットアップ例

2.2 性能ステータスの分類

システム レベルの ESD 性能は、レターリング システムを使用して各テストのレベルで分類されます。OEM は通常、さまざまなレベルでシステムの性能ステータスの要件を定義します。

表 2-1. 性能ステータスの表

ステータス	説明
A	性能からの逸脱なし (通常、カメラ システムでビデオに認識できる中断がないことを意味する)
B	一時中断するが、自動的に回復 (通常、瞬間的な視覚的グリッチが認識できることを意味する)
C	動作が中断して自動的に回復しない (通常、黒一色の画面になり、ビデオを回復するにはシステムの電源を入れ直すか、再初期化が必要があることを意味する)
D	DUT への永続的な損傷 (通常、IC または受動部品を含むシステムへの物理的な損傷を意味する)

3 FPD-Link ハードウェアの最適化

3.1 コネクタの接地

PoC を使用する ADAS アプリケーションでは、ケーブルシールドはリモート カメラに供給される DC 電力の帰路として機能し、2 つのシステム間のグランドリファレンスとしても機能します。ESD 衝撃や放射アンテナ エミッションなどの高エネルギー干渉イベントが原因で、シリアルライザとデシリアルライザのデバイス間のシステム グランドリファレンスで過渡的な中断が発生し、ビット エラーが発生する可能性があります。この影響を軽減するため、ケーブルシールドが FPD ケーブルの両側のシステム グランドに、できるだけ小さなインダクタンスで接続されていることを確認してください。TI は、コネクタのグランドピンを、サーマル リリーフ スプラインではなく、ソリッド接続を使用して、複数の層の PCB グランドプレーンに接続することを推奨しています。

推奨レイアウトでは、コネクタの GND ピンを大きなソリッドの GND プレーンに直接接続するだけでなく、GND プレーンを基板の他の層にある GND プレーンと接続する方法も紹介されています。この設計では、基板の 8 層すべてで、コネクタの下に GND プレーンがあります。

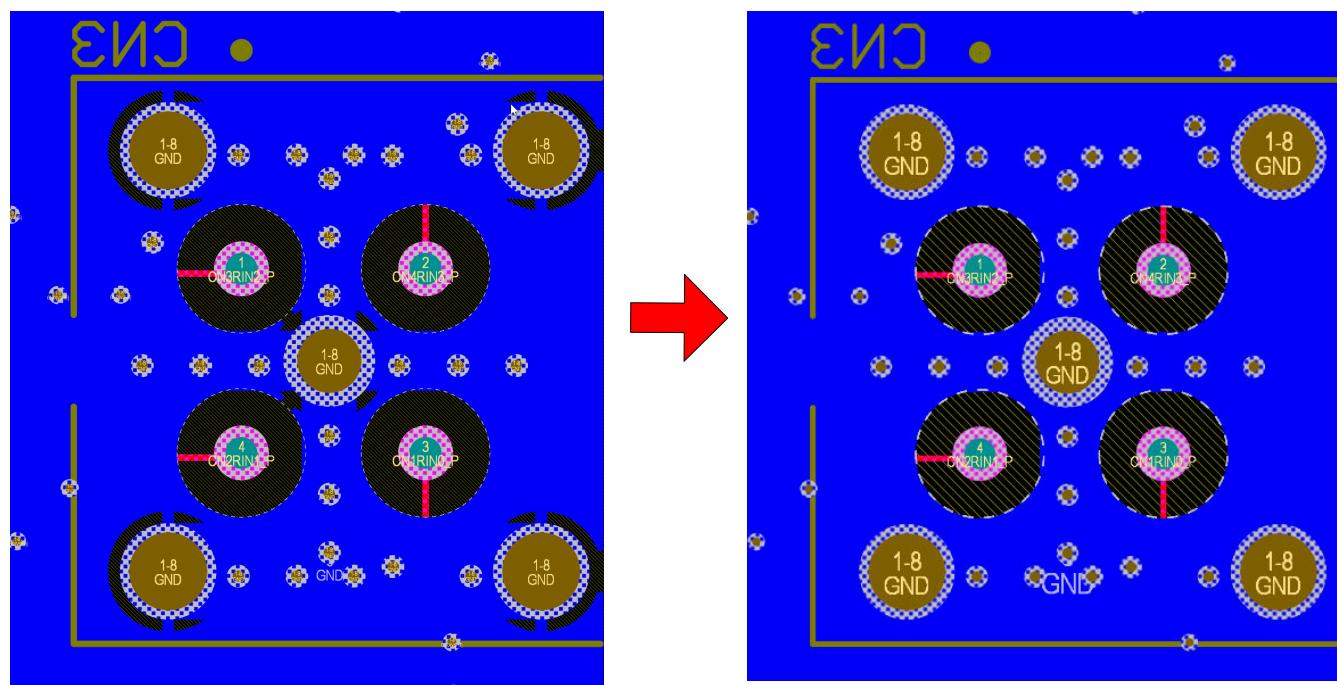
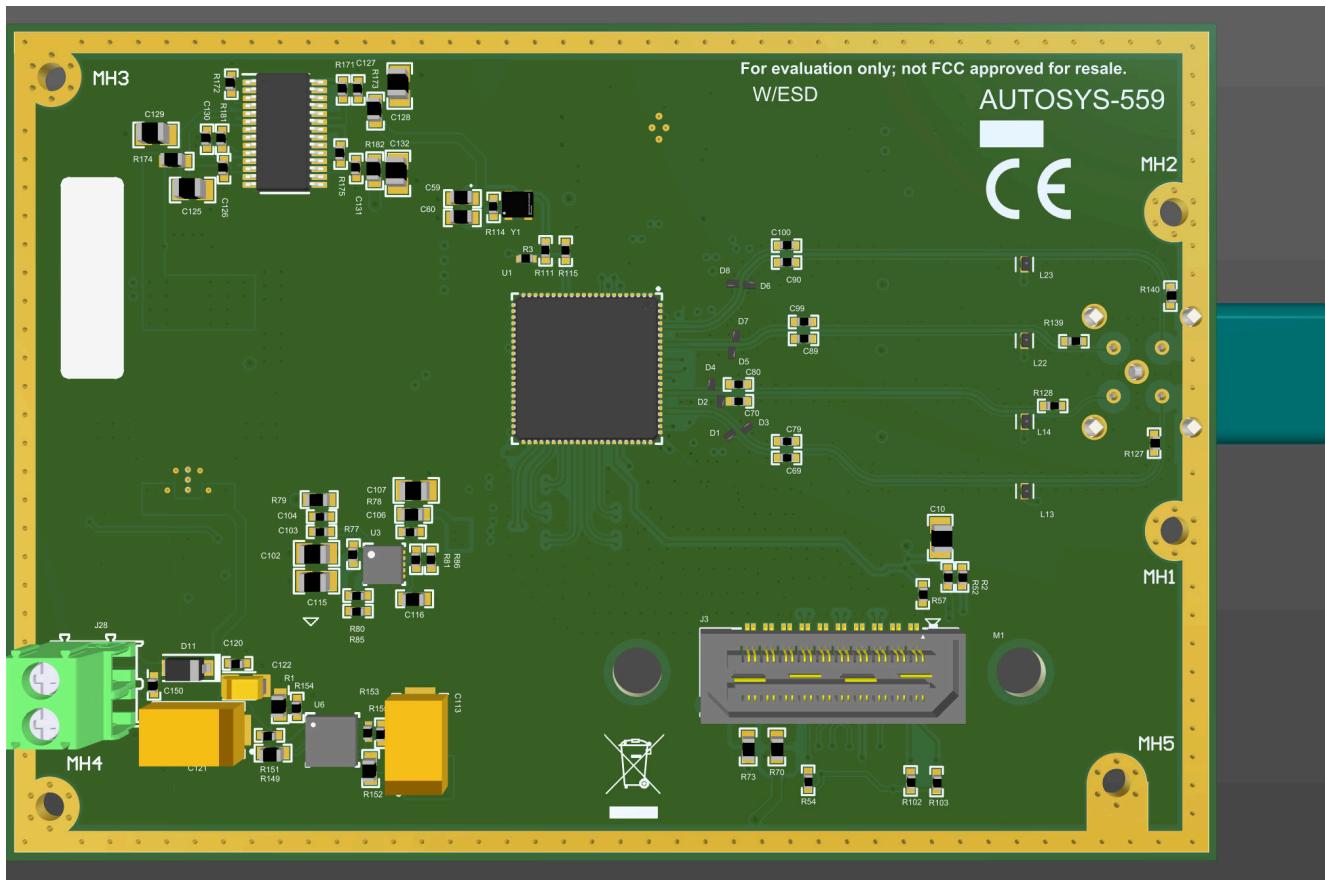


図 3-1. サーマル リリーフ レイアウト(左)、推奨のソリッド GND レイアウト(右)

3.2 PCB から筐体へのグラウンドィング

ESD が性能に及ぼす影響を低減するために、システム設計で最も重要な要素の 1 つは、低インダクタンスの放電パスを提供して、高速信号からエネルギーを遠ざけることです。これを実現するための最も効果的な方法の 1 つは、PCB グランドと、シャーシ グランドに接続されている筐体との間に、強力な接点を用意することです。PCB の周囲に露出したグランド リングを実装すると、スプリング接続、または導電性ネジとファスナーを使用して、筐体と基板のグランドとを複数の場所で接続できます。PCB の両側は、低インダクタンスの接続によって、筐体のグランドと接触している必要があります。これにより、基板のローカル セクション内でのグランド バウンスを防止できます。



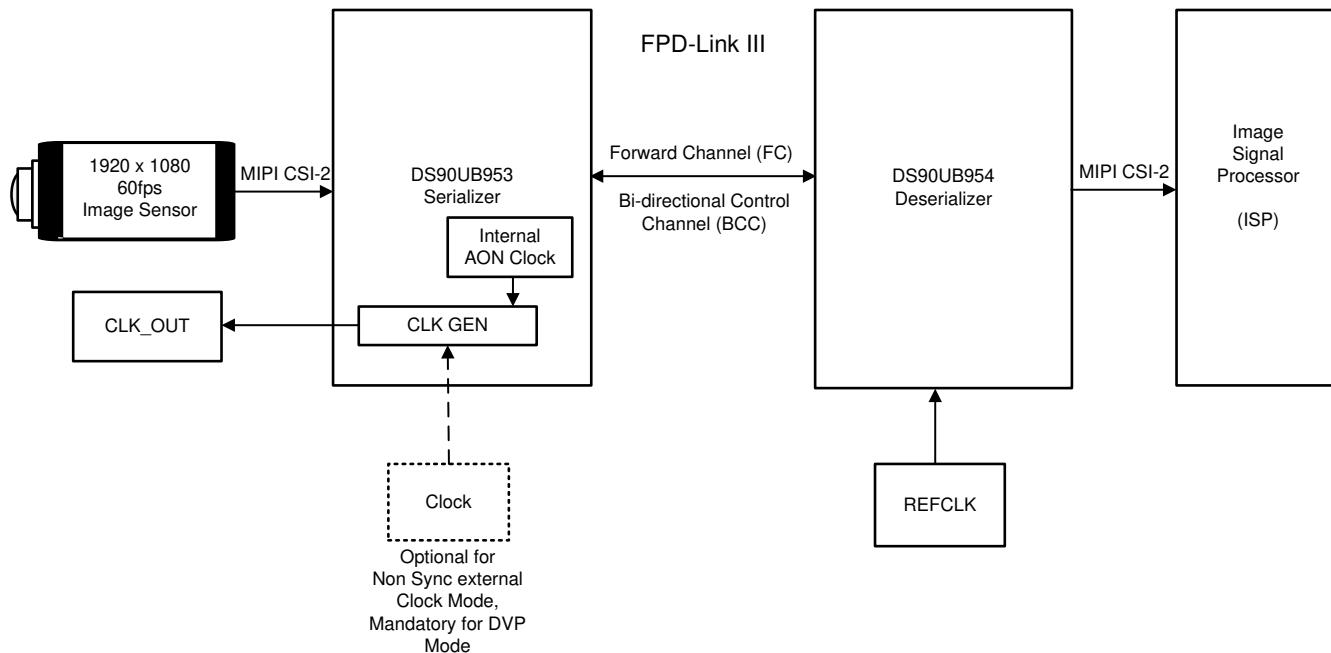


図 3-3. FPD-Link ADAS のクロック アーキテクチャ

FPD-Link III/IV ADAS シリアライザ デバイスでは、次の 2 種類の非同期モードを使用できます。

- 外部 CLKIN を使用する非同期モード
 - DS90UB953-Q1、DS90UB935-Q1、DS90UB635-Q1、DS90UB971-Q1 で、サポートされているすべてのレートで利用可能
 - CLKOUT 機能が利用可能
- 内部 AON クロックを使用する非同期モード
 - DS90UB953-Q1、DS90UB935-Q1、DS90UB635-Q1 でのみ利用可能
 - DS90UB971-Q1 では FPD-Link III モード (4Gbps) でのみ利用可能
 - CLKOUT 機能はディセーブル

TI は、ESD 認定テストのとき両方のモードを柔軟に評価できるよう、カメラ モジュールの設計で、どちらのモードもハードウェアでサポートするプロビジョニングを推奨しています。多くのシステム設計者は、部品数を削減し、高速バック チャネル GPIO を実現して、スペクトラム拡散クロック処理を可能にするために同期モードを好みます。データレートと CLKOUT 機能に関するシステムの必要に応じて、システム設計者は、非同期外部 CLKIN モードをオプションとして使用できるように、シリアル化に外部発振器入力を追加することもできます。それにより、ESD 認定試験のとき複数のモードでシステム性能を評価できます。システムで同期モードの使用を選択し、同期モードの ESD 性能がシステム設計者の要求を満たす場合は、最終ビルトで発振器を BOM から削除できます。

4 FPD-Link ソフトウェアの最適化

4.1 ロック検出のチューニング

FPD-Link ADAS フォワード チャネル プロトコルは、ビデオ ペイロード データのほか、GPIO、I2C、ステータス、クロック供給などの情報をシリアル フレームにパックし、ダウンストリームのデシリアルライザに送信します。受信データ フレームを適切にデコードするには、FPD-Link レシーバは、データの正しい遷移レートを判定することで受信高速信号にロックするとともに、受信データ フレームの整列（たとえば、各シリアル フレームの開始と終了）を適切に決定する必要があります。FPD-Link ADAS シリアルライザは、整列のタスクを行うため、フォワード チャネル データストリームにデコード サイクル アレイ（DCA）シーケンスと呼ばれる 130 文字のマルチ フレーム同期データ パターンと、フレームごとに 2 つの専用クロックビット（CLK0/CLK1）を挿入します。

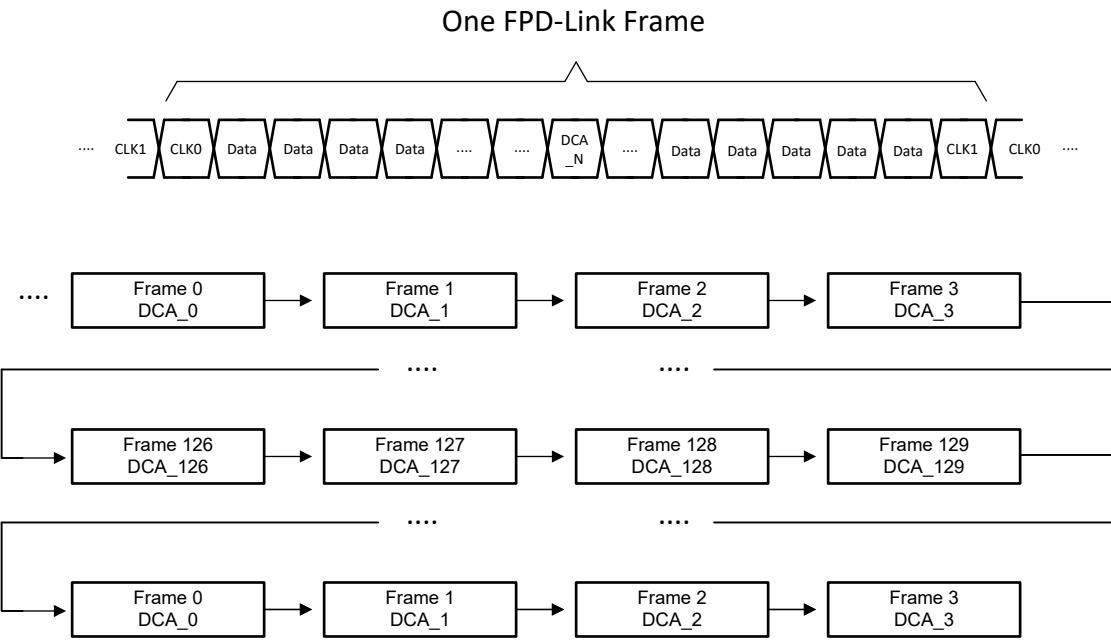


図 4-1. FPD-Link ADAS のフレームの構造とシーケンス

初期ロックが確立されると、デシリアルライザは受信シリアル フレームを継続的に監視し、各フレーム内の 2 つのクロックビットと 130 のフレーム DCA エンコード ビット パターンの予測される位置を検出します。デフォルトでは、130 受信フレームのスパンにおいてクロックビットとエンコードビットで 3 つのエラーが検出されると、FPD-Link ロック信号はドロップします。このデフォルト動作では、リンク内の BER が増加したことを示す優れた指標が得られます。同時にこのデフォルト動作では、リンクが ESD ストライクなどの過渡的なエラーイベントに対して非常に敏感になります。また、FPD フレームのクロックビットまたはエンコーディングビットで過渡的なエラーが発生しても、アプリケーションレベルの機能は中断されない可能性があることに注意してください。エラーが過渡的なものである限り、FPD-Link レシーバは自動的に動作と整列を維持します。

ESD ストレス時のリンクの堅牢性を向上させるため、TI は、デシリアルライザ内の LINK_ERR_THRESH 設定を増やすことを推奨します。この設定は、それぞれの 130 フレーム周期内で許容されるエラーの数を増やします。この値を超える数のエラーが周期内に検出されると、デシリアルライザはロックをドロップして再アクイジションを開始します。このスレッショルド設定は、レジスタ 0xB9 (LINK_ERROR_COUNT) にあります。設定されたスレッショルドを有効にするには、LINK_ERR_COUNT_EN も 1 に設定する必要があります。

ロックのドロップアルゴリズムの感度を低減するために適用できるもう 1 つの最適化は、リンク エラー カウンタの一部としてクロックビット エラー検出をディセーブルすることです。レジスタ 0xB6 = 0x1C を設定することで、クロックビットの検出はロックをドロップするためのマーカーとして使用されなくなります。この設定では、ロックをドロップするかどうかの検出に、繰り返される 130 フレームの DCA エンコーディング パターンのみが使用されます。

4.2 パリティ エラーの処理

FPD-Link のフォワード チャネル フレームのマルチ フレーム エンコーディングとクロック エラー チェックに加えて、FPD-Link の各フレームには既知のパリティも含まれています。すなわち、受信した各フレームにおいて 1 の数は必ず偶数です。これはデシリアルライザによってリアルタイムで検証されます。受信した FPD フレームで 1 の数が奇数であることが、デシリアルライザによって認識された場合、デシリアルライザは診断のためにエラーをフラグ付けし、パリティ エラー カウンタをインクリメントできます。

デフォルトでは、ADAS デシリアルライザは、パリティ エラーが含まれている FPD-Link のフォワード チャネル フレームを破棄するように設定されています。つまり、そのフレーム内に含まれる情報はダウンストリームのデータ パスに転送されません。このデフォルト動作では、エラーの伝播を防止できますが、アプリケーションに影響が及ぼない可能性がある過渡的な誤差に対して過敏になります。受信した FPD-Link フレームにシングル ビットのエラーが存在し、パリティ エラーのフラグが設定される例を考えてみましょう。

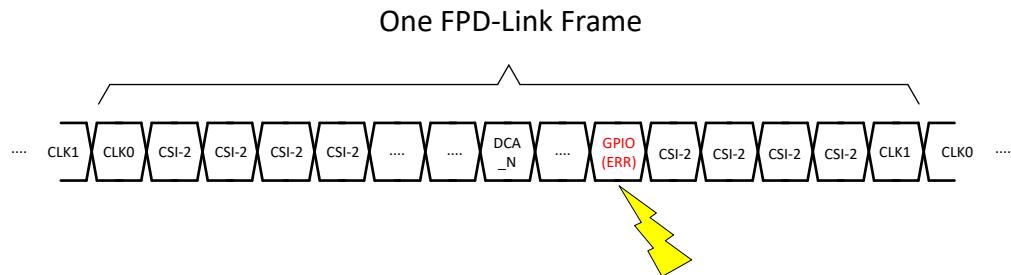


図 4-2. パリティ エラーの例

この例では、FPD-Link フレーム内のフォワード チャネル GPIO スロットの 1 つがビット エラーが影響を受けています。その結果、フレームの 1 の数が奇数に変わり、デシリアルライザはパケット全体を廃棄します。しかし、フレーム クロック ビットの 1 つのこの種の過渡的なエラーが発生しても、ロックの破壊や、フレーム境界の消失なしに許容可能な場合があります。さらに、この特定のエラーは、フレーム内の CSI-2 ビデオ データにも影響を与えていません。アプリケーションの使用事例によっては、このエラーが、システムのいかなる要素にも影響を及ぼさない可能性があります (たとえば、アプリケーションでフォワード チャネル GPIO を利用しない場合)。

FPD-Link III および IV CSI-2 の ADAS デシリアルライザ製品は、PORT_CONFIG2 レジスタで DISCARD_ON_PAR_ERR を 0 に設定することで、パリティ エラーのあるパケットを破棄しないように設定できます。この設定をディセーブルにすると、過渡的なエラー イベントに対する感度が低下し、耐性試験中にクラス A の性能を達成するため大きく役立ちます。

パリティ エラーのあるパケットを廃棄せずに処理できるようにしても、破損したビデオ データが見つけられずにデシリアルライザを通過するわけではありません。これは、FPD-Link ADAS デバイスが、CSI-2 パケットの全体をトンネリングして動作するためです。CSI-2 のペイロードとチェックサムは、両方ともシリアルライザとデシリアルライザを経由して変更されずに渡されから、ダウンストリームのアプリケーション プロセッサに到達します。CSI-2 のペイロード チェックサム (CRC) については、計算は行われません。その結果、FPD-Link チャネルのビット エラーによりビデオ ピクセル データが実際に破損すると、対応する CSI-2 パケットは、最終的な宛先で CRC チェックに失敗します。その後でアプリケーション プロセッサは、アプリケーションの必要に応じて、どのような方法でエラーを処理するのが最適かを決定できます。

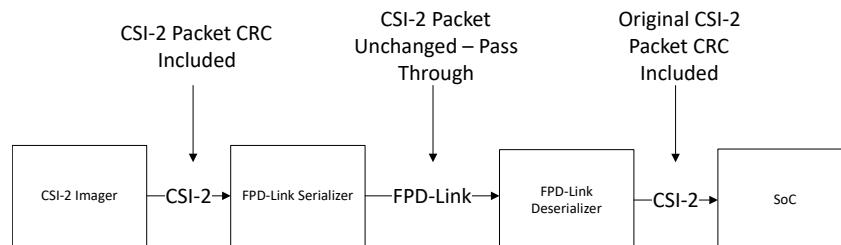


図 4-3. CSI-2 パケットのトンネリング

4.3 前方誤り訂正

FPD-Link III および FPD-Link IV 製品ファミリーの一部の FPD-Link デバイスには、FPD-Link フォワード チャネル パス内の前方誤り訂正 (FEC) 機能があります。前方エラー訂正では、チャネルの空き帯域幅を利用して、フォワード チャネルのデータペイロードとともにエラー訂正コード ワードを送信することで、ダウンストリームのデシリアライザが過渡的なエラーを修正できます。FEC 機能は、次のデバイスで使用できます。

- DS90UB971-Q1
- DS90UB971S-Q1
- DS90UB953-Q1
- DS90UB953A-Q1
- DS90UB935-Q1
- DS90UB635-Q1
- DS90UB960-Q1
- DS90UB962-Q1
- DS90UB662-Q1
- DS90UB9702-Q1
- DS90UB9722-Q1
- DS90UB9724-Q1
- DS90UB9724S-Q1
- DS90UB9742-Q1

次のデバイスでは FEC 機能を使用できません。

- DS90UB933-Q1
- DS90UB913A-Q1
- DS90UB913Q-Q1
- DS90UB934-Q1
- DS90UB954-Q1
- DS90UB936-Q1
- DS90UB638-Q1
- DS90UB964-Q1

FEC 機能を利用するには、リンク ペア内のシリアルライザ デバイスとデシrialライザ デバイスの両方が FEC をサポートしている必要があります。

選択可能な FEC モードは 3 つあり、それぞれ挿入される誤り訂正符号 (ECC) のコード サイズが異なります。

- 6 ビット FEC
- 7 ビット FEC
- 8 ビット FEC

FEC がイネーブルのとき、シリアルライザは複数の (X) データフレームを送信し、その後に ECC を含む複数の (Y) フレームを順に送信します。誤り訂正コード チャンク (Y) は、CSI-2 データ、GPIO データ、I₂C データを保護します。これにより、デシrialライザが 2 ビットのエラーを検出し、CSI-2 データのパケット ヘッダー内のデータ保護と同様に、データ チャンク (X) 内の 1 ビットエラーを訂正できます。

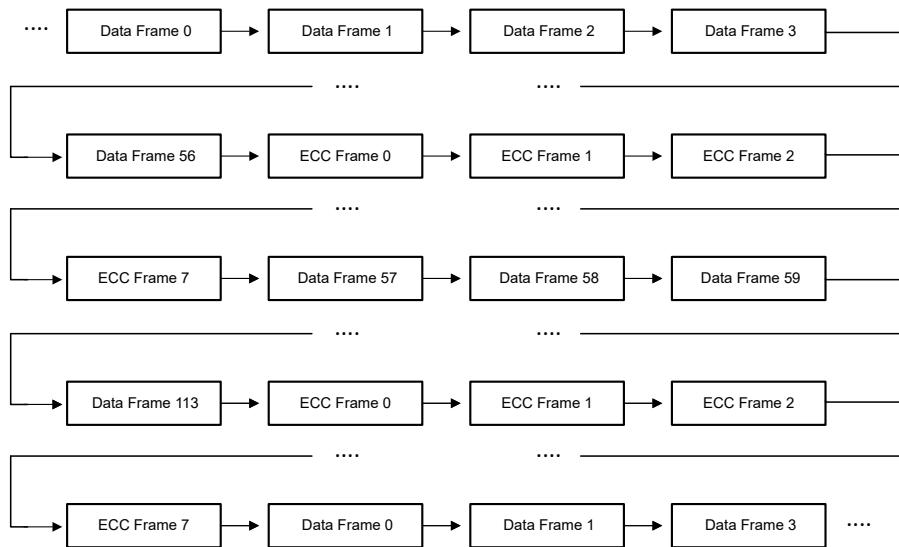


図 4-4. 8 ビット FEC シーケンスの例

各 FEC モードでは、130 フレームの DCA シーケンス内で送信される (X) データフレームと (Y) ECC フレームの比率が定義されています。6 ビットモードでは、データが小さなチャunkで保護されますが、訂正コードがより頻繁に送信されます。8 ビットモードでは、データがより大きなチャunkで保護され、訂正コードの送信頻度は少なくなります。この比率は次のように定義されます。

表 4-1. ECC コードの送信

FEC モード	データフレーム (X) の数	ECC フレーム (Y) の数	DCA シーケンスあたりの ECC フレームの数	DCA シーケンスあたりの データフレームの数	FEC オーバーヘッド
6 ビット	20	6	30	100	23%
7 ビット	36 と 37 を交互に送信	7	21	109	16%
8 ビット	57	8	16	114	12%

FEC オーバーヘッドにより、FPD-Link のフォワード チャネル パイプラインでビデオに使用できるペイロード能力が減少します。FEC がイネーブルになった後のリンクビデオ帯域幅が、アプリケーションの必要を満たすことを確認するよう注意する必要があります。

表 4-2. FEC オーバーヘッドの影響の例

デバイス	FPD-Link のレート (Gbps)	最大 CSI-2 帯域幅 (Gbps)	FEC モード	FEC がイネーブルなときの最大 CSI-2 帯域幅 (Gbps)
DS90UB971-Q1	7.55	6	6 ビット	4.8
DS90UB971-Q1	7.55	6	7 ビット	5.1
DS90UB971-Q1	7.55	6	8 ビット	5.3
DS90UB953-Q1	4	3.2	6 ビット	2.6
DS90UB953-Q1	4	3.2	7 ビット	2.7
DS90UB953-Q1	4	3.2	8 ビット	2.8
DS90UB935-Q1	4	2.528	6 ビット	2.528 (2.528Gbps と 3.2Gbps/1.23 の小さいほう)
DS90UB935-Q1	4	2.528	7 ビット	2.528 (2.528Gbps と 3.2Gbps/1.16 の小さいほう)
DS90UB935-Q1	4	2.528	8 ビット	2.528 (2.528Gbps と 3.2Gbps/1.12 の小さいほう)

FEC 機能をイネーブルにするには、デシリアルライザ レジスタ 0x4A (ポート固有) を設定します。

- 6 ビット FEC: 0x4A = 0x01
- 7 ビット FEC: 0x4A = 0x02
- 8 ビット FEC: 0x4A = 0x03

デシリアルライザは、FEC モードに移行するよう、シリアルライザにバック チャネルで自動的に通知します。

4.3.1 FEC テスト機能

サポート用のシリアルライザおよびデシリアルライザ デバイスには、FEC 機能が動作していることを確認する検証機能があります。シリアルライザはレジスタコマンドを通じて FEC エラーを注入でき、デシリアルライザは FPD-Link チャネルで検出された ECC1/ECC2 FEC エラーの数を診断できます。これらは、CSI_RX_STS レジスタで報告される ECC1/ECC2 CSI-2 エラーとは異なることに留意してください。

```
board.writeI2C(desAddr,0x76,0x01) # Force 1x FPD ECC1 error (Correctable)
board.writeI2C(desAddr,0x76,0x02) # Force 1x FPD ECC2 error (Detectable)
```

FEC エラー用のデシリアルライザ診断カウンタはポート固有です。診断結果を読み取る前に、レジスタ 0x4C で目的の RX ポートを選択します。

```
board.writeI2C(desAddr,0x4C,0x01) # Select RX port 0
board.ReadI2C(desAddr,0x48) # Read back count of FPD ECC1 Errors (Clear on Read)
board.ReadI2C(desAddr,0x49) # Read back count of FPD ECC2 Errors (Clear on Read)
```

5 最適化のテスト データ

ハードウェアとソフトウェアの最適化がシステム レベルの ESD 性能に及ぼす影響を示すため、TI は、FPD-Link IV DS90UB971-Q1 および DS90UB9702-Q1 デバイスを使用し、複数のデシリアルライザ構成で ISO 10605 テストを実施しました。

- 基本的なハードウェア設計 / ソフトウェア最適化なし
- 最適化されたハードウェア設計 / ソフトウェア最適化なし
- 最適化されたハードウェア設計 + ソフトウェア最適化

このテストは、7.55Gbps で動作する DS90UB971-Q1 カメラ モジュールを使用し、1.5m の Dacar 302 ケーブル経由でデシリアルライザ ハードウェア (DS90UB9702-Q1) に接続して実施されました。デシリアルライザ システムは、一般的な ECU サブシステムの設計をエミュレートするため、金属製の筐体に収容されています。デシリアルライザの筐体はテーブル GND プレーンに接続され、ケーブルとカメラ モジュールは絶縁ブロックを使用してグランド プレーンの上に固定されています。

この ISO 10605 テストの目標は、一般的な OEM EMC 承認セットアップをエミュレートすることです。ここでは、ESD 衝撃イベント発生時のシステムの性能がクラス A、B、C、D に分類されます。ハードウェアとソフトウェアが段階的に最適化された 3 つのシステムをテストすることで、典型的な改善の傾向を表しています。このテストは、すべてのシステム設計で期待される正確な結果を反映しているわけではなく、最適化されたハードウェアおよびソフトウェア設計によって実現可能となる相対的な性能向上を示すことを目的としています。

5.1 ベースライン ハードウェア - ソフトウェア最適化なし

テストの最初の 1 ラウンドは、このガイドの最初の方で説明した最適化が行われていない、ベースライン DS90UB9702-Q1 ハードウェア設計を使用して実行されました。システムは、過渡耐性のためのソフトウェア最適化が含まれていない基本的なリンク設定で構成されています。このテストの目標は、ESD 性能のベースラインを確立することです。このシステムは、カメラ モジュールとデシリアルライザ ボックス周辺のさまざまな衝撃ポイントで、接触放電と空中放電の両方を使用してテストされました。各ストレスレベルの後で診断レジスタを記録し、テスト中に FPD-Link ロックの消失や、訂正不可能な CSI-2 エラーが発生したかどうかが識別されました。これらのエラーは、ビデオ フィードのフリッカー、空白画面、または歪みの原因となる可能性があります。

表 5-1. ベースライン ハードウェア - ソフトウェア最適化なしの ESD 結果

衝撃の種類	レベル	一般的な OEM 要件	結果
空気	±4kV	クラス A	B
接触	±4kV	クラス A	B
空気	±6kV	クラス A	B
接触	±6kV	クラス A	B
空気	±8kV	クラス A	C
接触	±8kV	クラス B	C
空気	±15kV	クラス B	C
接触	±15kV	クラス B	C

- クラス A = ロック消失なし、訂正不可能な CSI-2 エラーなし。
- クラス B = ロックの一時的な消失または CSI-2 エラーが発生するが、自動的に回復
- クラス C = ビデオを再起動するために SoC のソフトリセットが必要

5.2 最適化されたハードウェア - ソフトウェア最適化なし

テストの 2 回目のラウンドは、高速信号パスを ESD 衝撃ノイズから絶縁することを目標として、デシリアライザ ハードウェアの最適化されたリビジョンを使用して実行されました。前のテストと同じ初期化ソフトウェアを使用して、ハードウェアのみからの相対的な改善を実証しました。適切なレイアウト手法と物理的な設計上の考慮事項を追加することで、システムはベースラインの例よりもわずかに高いレベルでクラス A の性能を達成できます。

表 5-2. 最適化されたハードウェア - ソフトウェア最適化なしの ESD 結果

衝撃の種類	レベル	一般的な OEM 要件	結果
空気	±4kV	クラス A	A
接触	±4kV	クラス A	A
空気	±6kV	クラス A	B
接触	±6kV	クラス A	B
空気	±8kV	クラス A	B
接触	±8kV	クラス B	B
空気	±15kV	クラス B	B
接触	±15kV	クラス B	B

- クラス A = ロック消失なし、訂正不可能な CSI-2 エラーなし。
- クラス B = ロックの一時的な消失または CSI-2 エラーが発生するが、自動的に回復

5.3 最適化されたハードウェアとソフトウェア

テストの最後のラウンドでは、性能をさらに向上させるため、デシリアライザの初期化に対してソフトウェアの最適化が行われました。ロック感度に微調整を加えることと、FPD-Link チャネルに FEC を追加することにより、過渡 ESD ストレス下での全体的な性能が大幅に向上升し、より高いレベルの ESD ノイズでもクラス A 性能を達成できるようになりました。

表 5-3. 最適化されたハードウェアとソフトウェアの ESD 結果

衝撃の種類	レベル	一般的な OEM 要件	結果
空気	±4kV	クラス A	A
接触	±4kV	クラス A	A
空気	±6kV	クラス A	A
接触	±6kV	クラス A	A
空気	±8kV	クラス A	A
接触	±8kV	クラス B	B
空気	±15kV	クラス B	B
接触	±15kV	クラス B	B

- クラス A = ロック消失なし、訂正不可能な CSI-2 エラーなし。
- クラス B = ロックの一時的な消失または CSI-2 エラーが発生するが、自動的に回復

6 ソフトウェア最適化のスクリプト例

レベル 1 の最適化

次のスクリプト例は、システム レベルの ESD テスト性能のために推奨されるソフトウェア最適化を組み合わせたものです。これらの設定は一般に、FEC 機能を備えた FPD-Link III/IV ADAS デバイスを使用するすべてのシステムに適用できます。この例では 7 ビット FEC を使用していますが、システムの必要とテスト性能に基づいて FEC 設定を調整できます。

```
board.writeI2C(desAddr,0x4C,0x0F) # Select all RX Ports
board.writeI2C(desAddr,0xB9,0x1F) # Increase LINK_ERR_THRESH
board.writeI2C(desAddr,0x4A,0x02) # Enable FPD FEC (7-bit)
board.writeI2C(desAddr,0x7C,0x00) # Disable FPD frame discard on parity error
```

レベル 2 の最適化

場合によっては、FPD-Link AEQ ゲインを調整するか、FPD-Link AEQ 範囲をより小さな値にロックすることで、性能をさらに向上できます。AEQ の調整は、他のすべてのハードウェアとレベル 1 のソフトウェア最適化を行った後で、最後の手段としてのみ使用する必要があります。AEQ を極端な値に調整すると、リンク性能に意図しない負の副作用を引き起こす可能性もあります。AEQ を手動で調整するときは、TI にお問い合わせください。

DS90UB960-Q1、DS90UB962-Q1、DS90UB662-Q1、DS90UB954-Q1、DS90UB936-Q1、DS90UB934-Q1、DS90UB964-Q1、DS90UB638-Q1 などの FPD-Link デバイスの場合、AEQ 値を強制的に小さくすることで性能を向上できます。ほとんどの場合、ケーブル長が 2 ~ 3m 未満なら、AEQ 値を強制的に 0 にすることで、システム レベルの ESD 性能が向上します。

```
# 960, 962, 662 AEQ Force for short cable length
board.writeI2C(desAddr,0x4C,0x0F) # Select all RX Ports
board.writeI2C(desAddr,0xD4,0x01) # Force AEQ = 0
```

DS90UB9702-Q1、DS90UB9722-Q1、DS90UB9724-Q1、DS90UB9742-Q1 など、FPD-Link IV CDR モードで動作する FPD-Link IV デシリアルライザの場合、AEQ 値を大きくすることで性能を向上できます。システム設計者が実験するときの開始値の例は、自動的に選択された AEQ に +5 を追加することです。AEQ を増加させるには、en_AEQ_LMS() 関数内で TI の初期化スクリプトに対して、次に示す下線の調整を行います。

```
def en_AEQ_LMS(first_time_power_up=1):
    if(first_time_power_up == 1):
        board.writeI2C(devAddr,0xB1,0x2C)
        read_aeq_init = board.readI2C(devAddr,0xB2)
        board.writeI2C(devAddr,0xB1,0x27)
        board.writeI2C(devAddr,0xB2,read_aeq_init + 5)
        board.writeI2C(devAddr,0xB1,0x28)
        board.writeI2C(devAddr,0xB2,read_aeq_init + 6)
    ...
    ...
```

7 システム レベルの追加ソフトウェア オプション

6 ~ 8kV レベルを上回るシステム レベルでクラス A の性能を実現する場合、ビデオ データを受信する SoC / プロセッサ 内に追加オプションを実装することもできます。より高い ESD ストレス レベルにわたってクラス A の性能を達成する一般的な方法は、エラーが発生したフレームや部分的に受信したフレームを破棄して視覚的な中断を防止できる、フレーム バッファリング方式を実装することです。この方法は、人間の視覚を使用してビデオ品質を判定するシステムで特に有効です。30Hz 以上のフレーム レートでは、1 フレームの繰り返しを検出することが困難なためです。この方法は、強力なシステム設計手法の代わりに使用する必要はありません。ベースラインのシステム性能が低い場合に、この方法と組み合わせると、視覚的に認識可能な遅延が発生する可能性があるためです。

ステップ 1

デシリアルライザ デバイスを、いずれかの RX ポートでエラーを検出したとき、割り込みを出力するように構成します。次の例では、クワッド チャネルのデシリアルライザが使用され、4 つの RX ポートがすべてアクティブなことを想定しています。

```
board.writeI2C(desAddr,0x23,0x8F) # Enable interrupts for all RX ports
```

ステップ 2

SoC 経由で割り込みピンを監視します。割り込みがトリガされると、割り込みソースをチェックして原因を識別し、ビデオ パスの破損を引き起こす可能性のある状態について、ポート固有のエラー フラグを生成します。パリティや ECC1 など一部のエラーはビデオに直接影響しないため、監視から除外されることに留意してください。

```
ERR = [0, 0, 0, 0] # RX Port-specific error flags
INT_STS = board.ReadI2C(desAddr,0x24) # Check which RX port triggered the interrupt
if INT_STS & 0x01 != 0: # RX0 interrupt
    board.writeI2C(desAddr,0x4C,0x01) # Select RX0
    RX_PORT_STS1 = board.ReadI2C(desAddr,0x4D)
    RX_PORT_STS2 = board.ReadI2C(desAddr,0x4E)
    CSI_RX_STS = board.ReadI2C(desAddr,0x7A)
    if RX_PORT_STS1 & 0x10 != 0: # LOCK_STS_CHG
        ERR[0] = 1
    if RX_PORT_STS2 & 0xC1 != 0: # LINE_LEN_CHG, LINE_CNT_CHG
        ERR[0] = 1
    IF CSI_RX_STS & 0XOE != 0: # LENGTH_ERR, CKSUM_ERR, ECC2_ERR
        ERR[0] = 1
if INT_STS & 0x02 != 0: # RX1 interrupt
    board.writeI2C(desAddr,0x4C,0x12) # Select RX1
    RX_PORT_STS1 = board.ReadI2C(desAddr,0x4D)
    RX_PORT_STS2 = board.ReadI2C(desAddr,0x4E)
    CSI_RX_STS = board.ReadI2C(desAddr,0x7A)
    if RX_PORT_STS1 & 0x10 != 0: # LOCK_STS_CHG
        ERR[1] = 1
    if RX_PORT_STS2 & 0xC1 != 0: # LINE_LEN_CHG, LINE_CNT_CHG
        ERR[1] = 1
    IF CSI_RX_STS & 0XOE != 0: # LENGTH_ERR, CKSUM_ERR, ECC2_ERR
        ERR[1] = 1
if INT_STS & 0x04 != 0: # RX2 interrupt
    board.writeI2C(desAddr,0x4C,0x24) # Select RX2
    RX_PORT_STS1 = board.ReadI2C(desAddr,0x4D)
    RX_PORT_STS2 = board.ReadI2C(desAddr,0x4E)
    CSI_RX_STS = board.ReadI2C(desAddr,0x7A)
    if RX_PORT_STS1 & 0x10 != 0: # LOCK_STS_CHG
        ERR[2] = 1
    if RX_PORT_STS2 & 0xC1 != 0: # LINE_LEN_CHG, LINE_CNT_CHG
        ERR[2] = 1
    IF CSI_RX_STS & 0XOE != 0: # LENGTH_ERR, CKSUM_ERR, ECC2_ERR
        ERR[2] = 1
if INT_STS & 0x08 != 0: # RX3 interrupt
    board.writeI2C(desAddr,0x4C,0x38) # Select RX3
    RX_PORT_STS1 = board.ReadI2C(desAddr,0x4D)
    RX_PORT_STS2 = board.ReadI2C(desAddr,0x4E)
    CSI_RX_STS = board.ReadI2C(desAddr,0x7A)
    if RX_PORT_STS1 & 0x10 != 0: # LOCK_STS_CHG
        ERR[3] = 1
    if RX_PORT_STS2 & 0xC1 != 0: # LINE_LEN_CHG, LINE_CNT_CHG
        ERR[3] = 1
    IF CSI_RX_STS & 0XOE != 0: # LENGTH_ERR, CKSUM_ERR, ECC2_ERR
        ERR[3] = 1
```

ステップ 3

ポート固有のエラー フラグに基づき、SoC をトリガして、エラーが発生したポートの現在のビデオ フレームを破棄し、前のフレームを 2 回表示します。

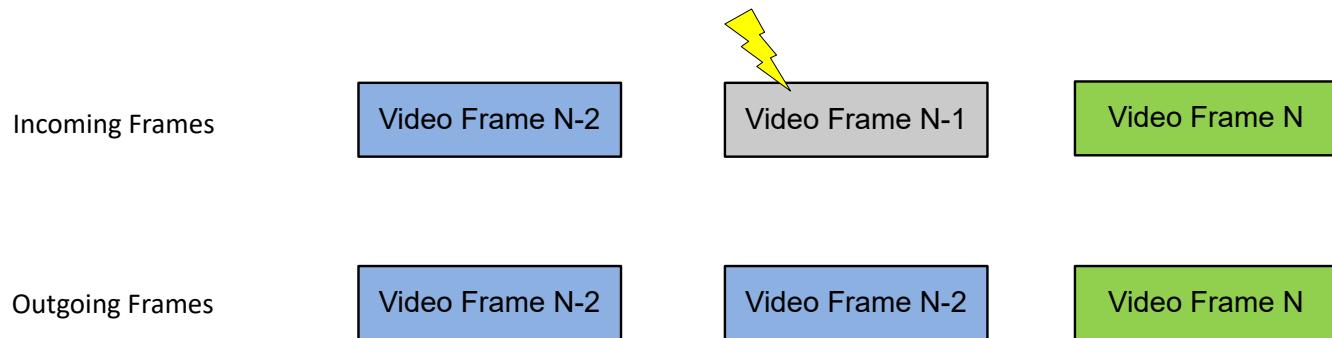


図 7-1. フレーム破棄と繰り返しの例

8まとめ

FPD-Link を使用するシステム レベルの ESD 性能は、ハードウェアとソフトウェアの両方の設計手法に関係します。ハードウェア レベルでは、設計で高速信号をシールドし、システム筐体と PCB との間に強力な接地を組み込む必要があります。ソフトウェア側では、FEC やロック検出チューニングなどのツールを使用して、ビデオ パスの実効 BER を向上させることができます。このアプリケーション ノートで記述されている手法は、システム レベルの ESD テストを対象としていますが、同じ手法を BCI や放射耐性など他のノイズ注入 EMC テストに適用し、性能を向上させることもできます。

9 参考資料

- ISO、『[10605:2023 道路車両 - 静電気放電による電気的障害のテスト方法](#)』

10 改訂履歴

Changes from Revision A (October 2024) to Revision B (October 2025)	Page
• DS90UB971S-Q1 および DS90UB9724S-Q1 を FEC がサポートされるデバイスのリストに追加.....	9
• ESD 性能への影響がないため、ソフトウェア最適化からロック検出用の CLK0/CLK1 ディセーブルを削除.....	13

Changes from Revision * (October 2024) to Revision A (October 2024)	Page
• ドキュメント全体にわたって表、図、相互参照の採番方法を更新.....	1
• 著者を追加.....	1
• パリティ エラーの例のテキストを更新 (偶数と奇数).....	8

重要なお知らせと免責事項

TIは、技術データと信頼性データ(データシートを含みます)、設計リソース(リファレンス デザインを含みます)、アプリケーションや設計に関する各種アドバイス、Webツール、安全性情報、その他のリソースを、欠陥が存在する可能性のある「現状のまま」提供しており、商品性および特定目的に対する適合性の默示保証、第三者の知的財産権の非侵害保証を含むいかなる保証も、明示的または默示的にかかわらず拒否します。

これらのリソースは、TI製品を使用する設計の経験を積んだ開発者への提供を意図したもので、(1)お客様のアプリケーションに適したTI製品の選定、(2)お客様のアプリケーションの設計、検証、試験、(3)お客様のアプリケーションに該当する各種規格や、その他のあらゆる安全性、セキュリティ、規制、または他の要件への確実な適合に関する責任を、お客様のみが単独で負うものとします。

上記の各種リソースは、予告なく変更される可能性があります。これらのリソースは、リソースで説明されているTI製品を使用するアプリケーションの開発の目的でのみ、TIはその使用をお客様に許諾します。これらのリソースに関して、他の目的で複製することや掲載することは禁止されています。TIや第三者の知的財産権のライセンスが付与されている訳ではありません。お客様は、これらのリソースを自身で使用した結果発生するあらゆる申し立て、損害、費用、損失、責任について、TIおよびその代理人を完全に補償するものとし、TIは一切の責任を拒否します。

TIの製品は、[TIの販売条件](#)、[TIの総合的な品質ガイドライン](#)、[ti.com](#)またはTI製品などに関連して提供される他の適用条件に従い提供されます。TIがこれらのリソースを提供することは、適用されるTIの保証または他の保証の放棄の拡大や変更を意味するものではありません。TIがカスタム、またはカスタマー仕様として明示的に指定していない限り、TIの製品は標準的なカタログに掲載される汎用機器です。

お客様がいかなる追加条項または代替条項を提案する場合も、TIはそれらに異議を唱え、拒否します。

Copyright © 2025, Texas Instruments Incorporated

最終更新日：2025年10月