

Application Note

Linux での TUSB7320/7340 向け USB 2.0 コンプライアンス モード



Mike Yu

概要

このアプリケーション ノートは、TUSB7320 または TUSB7340 コントローラを搭載したシステムの電氣的検証とコンプライアンス認証を行うハードウェア設計エンジニア、シグナル インテグリティ (SI) エンジニア、テスト技術者、およびシステム インテグレータを対象としています。Linux コマンド ライン、PCI Express (PCIe) コンフィギュレーション空間、および一般的な xHCI アーキテクチャに関する基本知識を有していること前提としています。

目次

1 はじめに.....	2
2 ハードウェアの構成とセットアップ.....	3
3 ソフトウェアの準備と環境のセットアップ.....	4
4 メモリマップト I/O (MMIO) レジスタ構成.....	5
5 まとめ.....	7
6 参考資料.....	7

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

目的と背景

TUSB7320 または TUSB7340 xHCI ホストコントローラを使用するプラットフォームのハードウェア開発段階では、USB-IF 物理層の仕様に厳密に準拠するためには、USB 2.0 の電氣的コンプライアンス テスト (アイダイアグラム、ジッタ、信号品質分析など) を実行することが非常に重要です。これらの測定を円滑に進めるため、高帯域幅オシロスコープなどの物理テスト装置では、ホストコントローラのダウンストリーム ポートが特定のテスト パターン (たとえば、テスト パケット、テスト J、テスト K) を継続的に送信する必要があります。

Windows® 環境では、開発者は通常、公式の USB-IF Hardware Start of Frame Evaluation and Test Tool (HSETT) を使用して、このプロセスをグラフィカル ユーザー インターフェイス (GUI) で自動化します。ただし、HSETT ユーティリティは Windows 環境専用であり、組込みシステム、産業用 PC、エンタープライズ サーバーなどの Linux® ベースのプラットフォームでは実行できません。したがって、Linux オペレーティング システムでは、ホストコントローラの内部レジスタを直接操作するために、低レベルのコマンドライン インターフェイスを使用して、USB 2.0 コンプライアンス モードに手動で移行する必要があります。このアプリケーション レポートは、Linux 上で USB 2.0 コンプライアンス テスト パターンを正常にトリガするための、標準化されたコマンドベースの手法を紹介します。

説明と範囲

このドキュメントで説明している手順と設定は、以下に示すテキサス インストルメンツの SuperSpeed USB 3.0 ホストコントローラに適用できます。

- TUSB7320:2 ポート USB 3.0 xHCI ホストコントローラ
- TUSB7340:4 ポート USB 3.0 xHCI ホストコントローラ

これらのデバイスは、互換性の高いレジスタ アーキテクチャとメモリマップト I/O (MMIO) 空間構成を共有するため、ここで詳述するレジスタレベルのコマンドは、両方のデバイスに等しく適用されます。このガイドでは、ネイティブの Linux カーネル インターフェイスと標準的なオープンソース コマンドライン ユーティリティを活用し、プロプライエタリ ソフトウェアやベンダ固有のドライバへの依存を排除します。

2 ハードウェアの構成とセットアップ

テスト設定の概要

USB 2.0 の電氣的コンプライアンス試験で正確な波形をキャプチャするには、被試験デバイス (DUT)、テストフィクスチャ、測定機器の間に適切な物理的接続が必要です。テスト構成では、USB 2.0 信号ライン (D+ と D-) を分離し、オシロスコープが TUSB7320 または TUSB7340 ダウンストリーム ポートによって生成される連続的なテストパターンをプローブできるようにする必要があります。

必要な機器と材料

評価を実行するには、以下のハードウェア コンポーネントが必要です。ホスト システム (DUT): TUSB7320 または TUSB7340 ホスト コントローラを搭載し、機能する Linux ディストリビューションを実行するハードウェア プラットフォーム。

- USB 2.0 コンプライアンス テスト フィクスチャ: D+ ラインと D- ラインを引き出すための、標準的な USB-IF 認定済みテスト ボード (USB 2.0 信号品質テスト ボードなど)。
- デジタル ストレージ オシロスコープ: コンプライアンス分析ソフトウェアを搭載した高帯域幅オシロスコープ (USB 2.0 High-Speed コンプライアンス テストの場合、通常 2GHz の帯域幅)。
- プローブ / ケーブル: フィクスチャをオシロスコープ チャンネルに接続するための適切な帯域幅定格を持つ、高品質の SMA ケーブルまたはマッチング済み差動プローブ。

ハードウェア接続の手順および要件

TUSB7320/TUSB7340 コントローラが指示されたコンプライアンス状態に正常に遷移することを確認するために、物理的な接続手順を以下に従って実施する必要があります。

1. システムの電源を切り、接地します。静電放電 (ESD) による損傷やグランド ループが高速信号測定に歪みを生じさせないように、DUT とオシロスコープが共通のシャーシ グランドを共有していることを確認します。
2. テスト フィクスチャを接続します。USB-IF コンプライアンス テスト フィクスチャを、評価対象の TUSB7320 または TUSB7340 の対象ダウンストリーム USB ポートに直接接続します。
3. 測定プロービングを確立します。テスト フィクスチャのテスト ポイント (選択したポートに対応) から、SMA 出力または差動プローブをオシロスコープの入力チャンネルに接続します。
4. テスト フィクスチャの終端を確認します。特定のテストスイート (たとえば、テスト パケット、テスト J/K) で必要とされる標準の 50Ω 終端 (または 90Ω 差動終端) に、テスト フィクスチャが設定されていることを確認します。

重要なハードウェア上の注意事項

TUSB7320 および TUSB7340 は、内部ポート ロジックを利用して、ライン状態の遷移および接続イベントを検出します。Linux ソフトウェア コマンドを実行する前に、物理テスト フィクスチャをテスト対象のポートに接続する必要があります。コントローラが D+ / D- ラインの正しい終端負荷を検出しない場合、ポート構成ロジックがテスト状態への遷移を拒否したり、ポートを自動的にリセットしてしまい、コンプライアンス パターンの生成が行われなくなることがあります。

3 ソフトウェアの準備と環境のセットアップ

組込みユーティリティの要件 (BusyBox のセットアップ)

TUSB7320 または TUSB7340 を搭載したプラットフォーム上で動作する多くの組込み Linux ディストリビューション、エンタープライズ アプライアンス、または最小限の OS 構成では、標準コア ユーティリティが削減されていることがあります。低レベルのメモリ操作を実行するには、必須の `devmem` アプレットを含む **BusyBox** スイートがシステムに必要となります。

レジスタの設定に進む前に、エンジニアは **BusyBox** が利用可能であることを確認し、ネイティブ システム イメージに **BusyBox** がない場合はインストールする必要があります。

1. 既存の **BusyBox** と `devmem` アプレットを確認します。プラットフォーム上に **BusyBox** がすでに存在しているかどうかを確認し、次のように実行して組み込みコマンドリストを調べます。

```
busybox --help
```

2. コマンドラインから **BusyBox** をインストールします。導入対象のプラットフォームがインターネット接続またはローカルパッケージ リポジトリへのアクセスを利用できる場合は、標準のパッケージ マネージャ コマンドを実行して、**BusyBox** 環境全体を取得、インストールします。

```
sudo apt-get install busybox
```

3. パス検証。インストール後、ユーティリティが標準実行パスに正しく応答することを確認します。システムのシンボリックリンクがパッケージ マネージャによって自動的に生成されない場合、アプレットは、次の **Busybox** プレフィックスを使用して、コマンドラインから常に明示的に呼び出すことができます。

```
sudo busybox devmem --help
```

ハードウェア マッピング手順に進む前に、コマンドが物理メモリ アドレスの読み取りと書き込みのシンタックスを出力することを確認してください。

4 メモリマップト I/O (MMIO) レジスタ構成

高レベルのカーネルドライバに頼らずに TUSB7320/TUSB7340 のダウンストリーム ポートを強制的に USB 2.0 コンプライアンス モードにするため、ソフトウェアは、システム メモリにマップされた xHCI オペレーショナル レジスタを直接操作できます。このセクションでは、**busybox devmem** ユーティリティを使用して、物理メモリ アドレスに対して低レベルの読み取り / 書き込み操作を直接実行します。

lspci を実行して PCI バス上のホストコントローラを特定する

ハードウェア メモリ空間とやり取りをする前に、PCI バスのトポロジをスキャンして、システム アーキテクチャによってテキサス インスツルメンツのコントローラに割り当てられている、特定のバス、デバイス、および機能 (BDF) アドレスを識別します。

PCI レーンに接続されているすべての周辺コンポーネントを一覧表示するには、基本的な **lspci** ユーティリティを実行します。

```
lspci
```

対象エントリ (例:01:00.0) が見つかったら、特定の BDF スロットと詳細な 16 進ダンプ用パラメータを指定してコマンドを再実行し、64 ビット ベース アドレス レジスタ (BAR) マッピングを抽出します。

```
sudo lspci -s <BUS:DEV.FUN> -vxx
```

出力フィールドから、xHCI オペレーショナル レジスタに対応する 64 ビット メモリ ベース アドレスを識別します。コントローラは複数のメモリ領域を割り当てることに注意してください。**[size=64K]** で指定されたアドレス エントリを選択します (例: Memory at f79f0000)。このアドレスは、それ以降のすべてのレジスタ オフセット計算における基本のベース アドレスとして機能します。

xHCI コントローラを停止する

個々のポートを強制的に電氣的コンプライアンス テスト モードにする前に、動作中のソフトウェア ループや割り込みルーチンによってレジスタの状態がオーバーライドされないように、xHCI コントローラを停止する必要があります。

Run/Stop ビットを 0 にクリアして、USB コマンド (USBCMD) レジスタ (ベース アドレスから 0x20 をオフセット) を変更します。次の 8 ビットの **devmem** コマンドを実行します。

```
sudo busybox devmem <Base_Address + 0x20> 8 4
```

対象のダウンストリーム ポートの電源をオフにする

ポート ロジックを適切にサイクルし、トランシーバをコンプライアンス モードへの移行に備えさせるためには、特定のダウンストリーム ポートへの電源を無効にする必要があります。

対象ポートのポート ステータスおよびコントロール (PORTSC) レジスタのオフセットを確認します。32 ビット幅のコマンドを利用して書き込み、ポートの電源ビットをクリアします。

```
sudo busybox devmem <Base_Address + Port_Offset> 32 0x0A000080
```

USB 2.0 テスト パターンを送信する

ポートが正常にシーケンスされたら、関連するテスト レジスタ領域 (ベース ポート オフセットから + 0x4 オフセット) 内のポート テスト コントロール フィールドに書き込み、目的のコンプライアンス テスト モードを選択します。

USB 2.0 High-Speed テスト パケット パターンを連続的に送信するには (電氣的アイダイアグラム分析に必要な)、次の 32 ビット コマンドを実行します。

```
sudo busybox devmem <Base_Address + Port_Offset + 0x4> 32 0x40000000
```

正常に実行されると、対象ポートの物理層 (PHY) は直ちに連続パターン生成を開始します。

物理ポート アドレス

代替物理ポートをテストするには、次の表に基づいて対象メモリ アドレスをシフトします。

表 4-1. TUSB7320/TUSB7340 ダウンストリーム ポート レジスタのオフセット

ポート	コントロール アドレス	テスト コマンド アドレス
1	0x0420	0x0424
2	0x0430	0x0434
3	0x0440 ⁽¹⁾	0x0444 ⁽¹⁾
4	0x0450 ⁽¹⁾	0x0454 ⁽¹⁾

1. TUSB7340 のみ

実行例

このセクションでは、ポート 1 の実際の実装について説明します。この手順例は、標準の Ubuntu プラットフォームを使用した実際のテスト環境で取得されたものです。

1. lspci 検出コマンドを実行すると次のような結果が得られます。

```
user@Ubuntu-OptiPlex-960: ~
File Edit View Search Terminal Help
user@Ubuntu-OptiPlex-960:~$ lspci
00:00.0 Host bridge: Intel Corporation 4 Series Chipset DRAM Controller (rev 03)
00:01.0 PCI bridge: Intel Corporation 4 Series Chipset PCI Express Root Port (rev 03)
00:02.0 VGA compatible controller: Intel Corporation 4 Series Chipset Integrated Graphics Controller (rev 03)
00:02.1 Display controller: Intel Corporation 4 Series Chipset Integrated Graphics Controller (rev 03)
00:03.0 Communication controller: Intel Corporation 4 Series Chipset HECI Controller (rev 03)
00:03.2 IDE interface: Intel Corporation 4 Series Chipset PT IDE Controller (rev 03)
00:03.3 Serial controller: Intel Corporation 4 Series Chipset Serial KT Controller (rev 03)
00:19.0 Ethernet controller: Intel Corporation 82567LM-3 Gigabit Network Connection (rev 02)
00:1a.0 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #4 (rev 02)
00:1a.1 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #5 (rev 02)
00:1a.2 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #6 (rev 02)
00:1a.7 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB2 EHCI Controller #2 (rev 02)
00:1b.0 Audio device: Intel Corporation 82801JD/DO (ICH10 Family) HD Audio Controller (rev 02)
00:1c.0 PCI bridge: Intel Corporation 82801JD/DO (ICH10 Family) PCI Express Port 1 (rev 02)
00:1c.1 PCI bridge: Intel Corporation 82801JD/DO (ICH10 Family) PCI Express Port 2 (rev 02)
00:1d.0 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #1 (rev 02)
00:1d.1 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #2 (rev 02)
00:1d.2 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB UHCI Controller #3 (rev 02)
00:1d.7 USB controller: Intel Corporation 82801JD/DO (ICH10 Family) USB2 EHCI Controller #1 (rev 02)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev a2)
00:1f.0 ISA bridge: Intel Corporation 82801JDO (ICH10DO) LPC Interface Controller (rev 02)
00:1f.2 RAID bus controller: Intel Corporation SATA Controller [RAID mode] (rev 02)
00:1f.3 SMBus: Intel Corporation 82801JD/DO (ICH10 Family) SMBus Controller (rev 02)
01:00.0 USB controller: Texas Instruments TUSB73x0 SuperSpeed USB 3.0 XHCI Host Controller (rev 02)
```

図 4-1. PCI アドレスの検出コマンド

```
user@Ubuntu-OptiPlex-960:~$ sudo lspci -s 01:00.0 -vxx
01:00.0 USB controller: Texas Instruments TUSB73x0 SuperSpeed USB 3.0 XHCI Host Controller (rev 02) (prog-if 30 [XHCI])
    Flags: bus master, fast devsel, latency 0, IRQ 16
    Memory at f79e0000 (64-bit, non-prefetchable) [size=64K]
    Memory at f79e0000 (64-bit, non-prefetchable) [size=8K]
    Capabilities: [40] Power Management version 3
    Capabilities: [48] MSI: Enable+ Count=1/8 Maskable- 64bit+
    Capabilities: [70] Express Endpoint, MSI 00
    Capabilities: [c0] MSI-X: Enable+ Count=8 Masked-
    Capabilities: [100] Advanced Error Reporting
    Capabilities: [150] Device Serial Number 08-00-28-00-00-20-00-00
    Kernel driver in use: xhci_hcd
00: 4c 10 41 82 06 04 10 00 02 30 03 0c 10 00 00 00
10: 04 00 9f f7 00 00 00 00 04 e0 9e f7 00 00 00 00
20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
30: 00 00 00 00 40 00 00 00 00 00 00 00 0b 01 00 00
```

図 4-2. ベース アドレス用コマンド

抽出された主要なターミナル出力:

- PCI アドレス (BDF): **01:00.0**
 - 64 ビット ベース アドレス (BAR): **0xf79f0000** (Memory at f79f0000 から)
2. テスト パケットをトリガするための完全なコマンド シーケンス。抽出されたベース アドレス (0xf79f0000) を使用して、テスト パケット モードにポート 1 (コントロール アドレス 0x420、テスト コマンド アドレス 0x424) をトリガする正確なレジスタ実行ログを以下に示します。

```
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0020 8
0x05
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0020 8 4
0x04
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0420 32
0xA0002A0
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0420 32 0xA0000000
0x00000000
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0420 32
0xA0000000
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0424 32
0x00000000
user@Ubuntu-OptiPlex-960:~$ sudo busybox devmem 0xf79f0424 32 0x40000000
0x40000000
```

図 4-3. テスト パケットをトリガするコマンド

5 まとめ

HSETT のような標準的なグラフィカル ユーティリティを使用できない Linux プラットフォームで、USB 2.0 の電氣的コンプライアンス検証を実行するには、堅牢な低レベル アプローチが必要です。ネイティブの PCI ユーティリティ (lspci) と、BusyBox (devmem) による直接メモリ操作を活用することで、ハードウェア検証エンジニアおよびシグナル インテグリティ エンジニアは、TUSB73x0 ホスト コントローラ ファミリー全体で必須のテスト パターンを一貫してトリガできます。このアプリケーション レポートで概要を示した構成手順に従うことで、再現可能で正確な測定ワークフローが検証され、USB-IF 認証 取得に向けた明確な道を切り開くことができます。

6 参考資料

1. テキサス インスツルメンツ、[「TUSB7320 SuperSpeed USB 3.0 2 ポート xHCI ホスト コントローラ」](#)、製品ページ。
2. テキサス インスツルメンツ、[「TUSB7340 SuperSpeed USB 3.0 4 ポート xHCI ホスト コントローラ」](#)、製品ページ。
3. Keysight、[「Understanding the USB 2.0 PID/VID Method for Testing High Speed Embedded Hosts \(高速組み込みホストを試験するための USB 2.0 PID/VID 手法の理解\)」](#)、Web ページ。
4. USB、[「USB 2.0 Electrical Compliance Test Specification \(USB 2.0 電氣的コンプライアンス テストの仕様書\)」](#)、Web ページ。

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黙示保証、第三者の知的財産権の非侵害保証を含みいかなる保証も、明示的または黙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す。(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、TI は一切の責任を拒否します。

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 © 2026, Texas Instruments Incorporated

最終更新日：2025 年 10 月