binho

アプリケーションノート

Binho Supernova によるバスの 3.3 V SMBus から 1.8 V I3C への移行

AN0018

文書情報

キーワード	AN0018, Supernova, I3C, SMBus, 7Eh, DISEC, SETDASA, ENTDAA, ダイナミックアドレス割り当て, バス電圧, プルアップ, NVMe-MI, pyCosmicSDK
概要	3.3 V の SMBus バス上で起動し、1.8 V の I3C に移行するデバイスでは、ホストがそのデバイスを見つけ、 SMBus 専用のデバイスがバス上に残っていないことを確認し、バス電圧を下げ、ダイナミックアドレスを 割り当てる必要があります。Supernova はこれらすべてを 1 つの I3C ポートから実行します。本アプリケーション ノートでは、5 つのステップ（最後のステップは SETDASA または ENTDAA）を実行するスクリプトを提供し、 2 台目の Supernova が I3C デバイスの役割を果たすベンチで動作を示します。結果は SETDASA で 20 回中 20 回、 ENTDAA で 20 回中 20 回成功し、バスのハイレベルは移行前に 3.34 V、移行後に 1.79 V と測定されました。

Binho Inc.
42 Dore Street, Unit A, San Francisco, CA 94103

改訂 1.0
6 NOVEMBER 2026

1 はじめに

一部のデバイスは 3.3 V の SMBus ターゲットとして起動し、1.8 V の I3C に移行できます。移行のタイミングは ホストが決定します。ホストはまず、バス上に残るすべてのデバイスが移行に追従できることを確認し、その後 バス電圧を下げてダイナミックアドレスを割り当てます。SMBus しか使えないデバイスはこの移行に耐えられない ため、そのようなデバイスが見つかった場合、ホストは SMBus モードのままにします。

Supernova の I3C ポートは、同じ 2 本のピンで SMBus（レガシー I2C）と I3C の両方を扱え、その電圧は ソフトウェアから設定します。そのため、一連の手順全体を 1 つのポートと 1 つのスクリプトで実行できます。

1.1 対象範囲

本書では以下を扱います。

- 一連の手順の 5 つのステップと、各ステップの背後にある pyCosmicSDK の呼び出し
- 配線、およびバスのプルアップが電圧変更に追従するための接続先
- SETDASA または ENTDAA をアドレス割り当てに使用して一連の手順を実行するスクリプト
- Supernova を I3C デバイスとして使用したベンチでの測定結果

PCI-SIG のアウトオブバンド管理仕様における SMBus から I3C Basic への検出フロー（ホットジョイン （Hot-Join）、ARP、SMBus リセット）は AN0014 で扱うため本書では扱いません。また、MCTP および NVMe-MI のメッセージも扱いません。

1.2 対象デバイス

表 1. 対象デバイス

役割	部品	識別情報	根拠
コントローラ	Binho Supernova	ハードウェアリビジョン C、ファームウェア 4.6.2、I3C HV ポート	測定済み、本書
コントローラ	Binho Supernova	ファームウェア 4.6.0 以降	想定：使用するすべての呼び出しが 4.6.0 に存在
I3C デバイス	I3C ターゲットとしての Binho Supernova	ハードウェアリビジョン C、ファームウェア 4.6.2	測定済み、本書
I3C デバイス	スタティックアドレスで SMBus に応答し、SETDASA または ENTDAA をサポートする任意のデバイス		未測定

1.3 関連資料

表 2. 関連資料

資料	タイトル
MIPI I3C	MIPI Alliance Specification for I3C, version 1.1.1
SMBus 3.2	System Management Bus Specification, version 3.2
AN0014	Single-port I2C/SMBus/I3C operation for NVMe-MI on Binho Supernova

2 システム概要

表 3 に、各ステップとそれを実行する呼び出しを示します。

表 3. 一連の手順のステップ

ステップ	動作	pyCosmicSDK の呼び出し
1	3.3 V の SMBus モードで動作する	`set_voltage(voltage_mv=3300)`、`bring_up(...)`、`ccc.rstdaa()`
2	7Eh コマンドをブロードキャストする。ACK は、バス上に少なくとも 1 つの I3C デバイスがあることを意味する	`ccc.disec_broadcast(I3cEvent.HJ)`。NACK の場合は `FW_I3C_NACK_ADDRESS` を伴う `I3cError` が発生する
3	すべての SMBus アドレスをプローブする。応答し、かつ既知の I3C デバイスではないデバイスは SMBus 専用である。そのようなデバイスが残っている場合は SMBus モードのままにする	08h から 77h に対して `legacy_i2c_read(address, 1)`
4	バスとそのプルアップを 3.3 V から 1.8 V に移行する	`set_voltage(voltage_mv=1800)`
5	SETDASA または ENTDAA でダイナミックアドレスを割り当て、各デバイスの PID を読み出す	`ccc.setdasa(static=, dynamic=)` または `init_bus()`、その後 `ccc.getpid(address)`

一連の手順は、次の 3 つの事実に基づいています。

- **ステップ 2 は SMBus 転送ではなく CCC です。** オープンドレインで送信されるのは 7Eh ヘッダのみである ため、SMBus 専用デバイスには自分のものではないアドレスが見え、それを無視します。使用する CCC は DISHJ を指定した DISEC で、ホストがまだ SMBus モードにある間に I3C デバイスがホットジョインすることも防ぎます。
- **ステップ 3 には I3C デバイスのスタティックアドレスのリストが必要です。** スタティックアドレスを持つ I3C デバイスは、そのアドレスへの SMBus 読み出しにも応答するため、応答があるだ

けではそのデバイスが SMBus 専用であることの証明になりません。スクリプトは I3C デバイスのスタティックアドレス（`--i3c-static`）を受け取り、それ以外の応答をすべて SMBus 専用として扱います。

- **ステップ 4 でプルアップが移行するのは、プルアップがポート電圧から供給されている場合のみです。** Supernova の SDA プルアップは I3C ポート電圧に追従します。外部プルアップが追従するのは、ポートの VTARG ピンに接続されている場合のみです（3.3 節）。

3 ハードウェアのセットアップ

3.1 必要な機材

表 4. 必要な機材

項目
Binho Supernova、ファームウェア 4.6.0 以降
USB-C ケーブル
被試験デバイス、またはその役割を果たす 2 台目の Supernova（6 章）
バスにプルアップがない場合は、VTARG に接続するプルアップ抵抗

3.2 接続

表 5. Supernova I3C HV ポートとデバイス間の接続

Supernova I3C HV ポート	デバイス	必須
SCL	SCL	必須
SDA	SDA	必須
GND	GND	必須
VTARG	プルアップ抵抗、またはデバイスの I/O 電源	バスに外部プルアップがある場合は必須

3.3 バスに関する考慮事項

I3C ポートの SCL にはプルアップがなく、SMBus にはプルアップが必要です。SMBus のプルアップは、固定の 3.3 V レールではなく、ポートの VTARG ピンに接続してください。VTARG は `set_voltage` に追従するため、 ステップ 4 でプルアップはバスとともに 3.3 V から 1.8 V に移行します。固定の 3.3 V レールに接続した プルアップでは、ステップ 4 の後もバスが 3.3 V に保たれ、1.8 V デバイスが許容する電圧を超えてしまいます。

> **注意：** 3.3 V トレラントでないデバイスは、ステップ 1 から 3 の間、バス上に接続しないでください。
>
> **注意：** バス上のコントローラは 1 つだけにしてください。

4 ソフトウェアのセットアップ

スクリプトを実行するのと同じ Python で、PyPI から pyCosmicSDK の最新リリースをインストールします。

```
python -m pip install --upgrade pycosmicsdk
```

Mission Control で Supernova をファームウェア 4.6.0 以降に更新します。

5 手順

バス上の各 I3C デバイスのスタティックアドレスを指定してスクリプトを実行します。デフォルトは、SMBus が 3300 mV、I3C が 1800 mV で、SETDASA により 08h からダイナミックアドレスを割り当てます。

```
python smbus_to_i3c_daa.py --i3c-static 0x3A --daa setdasa
python smbus_to_i3c_daa.py --i3c-static 0x3A --daa entdaa
```

出力には各ステップが示されます。6 章のベンチでは次のとおりです。

```
FW-4.6.2 controller | target FW-4.6.2
1. SMBus mode, bus at 3300 mV
2. DISEC to 7Eh: ACK, an I3C device is present
3. SMBus probe 08h-77h: 3Ah -> SMBus-only: none
4. Bus and pull-ups at 1800 mV
5. SETDASA: dynamic 08h, GETPID 0B0000000A01
```

7Eh に ACK するデバイスがない場合はステップ 2 の後で、SMBus 専用デバイスが応答した場合はステップ 3 の後で スクリプトは停止します。どちらの場合も、バスは SMBus の電圧のままです。アドレスを割り当てたすべての デバイスが PID を返した場合、スクリプトは 0 で終了します。

一連の手順を独自のコードで使用するには、`run` 関数をコピーしてください。各ステップは数行で、表 3 の 呼び出しのみを使用します。

表 6. smbus_to_i3c_daa.py のオプション

オプション	デフォルト	意味
`--serial`	任意	コントローラ Supernova のシリアル番号（またはその先頭部分）
`--i3c-static`	必須	I3C デバイスのスタティックアドレス。それ以外の SMBus 応答はすべて SMBus 専用として扱われる
`--daa`	`setdasa`	`setdasa` はスタティックアドレスごとに 1 つのアドレスを割り当てる。`entdaa` はファームウェアにすべてのデバイスへの割り当てを任せる
`--first-dynamic`	0x08	SETDASA が割り当てる最初のダイナミックアドレス
`--smbus-mv`	3300	ステップ 1 から 3 のバス電圧
`--i3c-mv`	1800	バス電圧（ステップ 4 以降）
`--settle-ms`	10	電圧変更後の待ち時間
`--emulate-target`	なし	最初の `--i3c-static` アドレスで I3C デバイスの役割を果たす 2 台目の Supernova のシリアル番号
`--rounds`	1	一連の手順をこの回数だけ実行し、成功した回数を表示する

6 測定結果

6.1 テスト構成

表 7. テスト構成

項目	値
コントローラ	Binho Supernova、ハードウェアリビジョン C、ファームウェア 4.6.2（`FW-4.6.2`）、I3C HV ポート
I3C デバイス	2 台目の Supernova、ファームウェア 4.6.2、`--emulate-target` による I3C ターゲット、スタティックアドレス 3Ah、PID 0B0000000A01、BCR 06h、DCR C5h。ポート電圧はコントローラのポート電圧と合わせて設定
ホストソフトウェア	PyPI の pyCosmicSDK 1.6.0、Ubuntu 22.04
バス	I3C HV ポート間を SCL、SDA、GND で配線
アナライザ	Saleae Logic Pro 16、SCL と SDA でアナログおよびデジタルを取得

6.2 結果

表 8. 結果

実行	結果
SETDASA、20 回	20 回中 20 回：7Eh ACK、SMBus 上は 3Ah のみ、ダイナミックアドレス 08h、PID を読み出し
ENTDAA、20 回	20 回中 20 回：同上、ダイナミックアドレス 08h はファームウェアが割り当て
バスのハイレベル、アナライザで 1 回実行	SCL はステップ 1 から 3 で 3.34 V、ステップ 4 の後で 1.79 V

ステップ 3 の SMBus 専用分岐は、このベンチに SMBus 専用デバイスがないため実行していません。 これらは 1 つのベンチで当社が測定した値であり、仕様ではありません。

7 トラブルシューティング

表 9. トラブルシューティング

症状	考えられる原因と対処
バス上に I3C デバイスがあるのに、ステップ 2 が NACK を報告する	デバイスに電源が入っていないか、SMBus の電圧になっていません。`--smbus-mv` とデバイスの電源を確認してください。
ステップ 3 が I3C デバイスを SMBus 専用として表示する	そのスタティックアドレスが `--i3c-static` に含まれていません。追加してください。
ステップ 3 で何も見つからないが、ステップ 5 は成功する	デバイスがスタティックアドレスで SMBus 読み出しに応答しません。一連の手順には影響しません。
ステップ 5 の GETPID が `FW_I3C_NACK_ADDRESS` で失敗する	バスがまだデバイスの I/O 電圧より高いか、デバイスがその DAA 方式をサポートしていません。プルアップが VTARG に接続されていることを確認するか、もう一方の `--daa` を試してください。
ステップ 4 の後もバスが 3.3 V のまま	外部プルアップが固定の 3.3 V レールに接続されています。VTARG に接続してください（3.3 節）。

8 略語

表 10. 略語

用語	意味
I3C	MIPI Improved Inter-Integrated Circuit
SMBus	System Management Bus（システムマネジメントバス）
CCC	I3C 共通コマンドコード（common command code）
DISEC	ターゲットイベント（ここではホットジョイン、DISHJ）を無効にする CCC
SETDASA	スタティックアドレスを持つデバイスにダイナミックアドレスを与える CCC
ENTDAA	すべてのデバイスに対してダイナミックアドレス割り当てを開始する CCC
RSTDAA	すべてのダイナミックアドレスをリセットする CCC
GETPID	デバイスの 48 ビットのプロビジョナル ID を読み出す CCC
PID	プロビジョナル ID（Provisional ID）
BCR、DCR	I3C のバス特性レジスタおよびデバイス特性レジスタ
VTARG	ポート電圧で出力される Supernova I3C ポートの電源出力

9 本書のソースコードに関する注記

本書に示し、付属のアーカイブで提供するサンプルコードは、評価および参考を目的として提供されます。 MIT ライセンスの下で公開されており、その本文はアーカイブに含まれています。

アーカイブ `AN0018-assets.zip` には `smbus_to_i3c_daa.py` と `LICENSE` が含まれています。

10 改訂履歴

表 11. 改訂履歴

リビジョン	日付	説明
0.1	23 October 2026	社内レビュー用のドラフト。ファームウェア 4.6.2 と pyCosmicSDK 1.6.0 で測定。
1.0	6 November 2026	公開。

法的情報（英語）

Definitions

Draft

A document marked draft is under internal review and subject to formal approval. Binho Inc. makes no representation or warranty as to its accuracy or completeness and accepts no liability arising from its use.

Disclaimers

Limited warranty and liability

Information in this document is believed to be accurate and reliable. Binho Inc. makes no representation or warranty, express or implied, as to its accuracy or completeness, accepts no liability for the consequences of its use, and takes no responsibility for content originating outside Binho Inc.

In no event shall Binho Inc. be liable for any indirect, incidental, punitive, special or consequential damages, including without limitation lost profits, lost savings, business interruption, or costs of removing or replacing any product, whether or not based on tort, warranty, breach of contract or any other legal theory. Notwithstanding any damages a customer might incur, the aggregate and cumulative liability of Binho Inc. is limited in accordance with its terms and conditions of commercial sale.

Right to make changes

Binho Inc. reserves the right to change the information in this document, including specifications and product descriptions, at any time and without notice. This document supersedes all information supplied prior to its publication.

Suitability for use

Binho Inc. products are not designed, authorized or warranted for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure of a Binho Inc. product can reasonably be expected to result in personal injury, death, or severe property or environmental damage. Any such inclusion or use is at the customer's own risk and Binho Inc. accepts no liability for it.

Applications

Applications described here are illustrative only. Binho Inc. makes no warranty that they are suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their own applications and products, for appropriate design and operating safeguards, and for determining whether a Binho Inc. product is suitable and fit for their purpose. Binho Inc. accepts no liability for assistance with applications or customer product design.

Third-party products and specifications

This document refers to products, boards, specifications and documentation supplied by third parties, which remain the responsibility of their respective owners. Binho Inc. makes no warranty regarding them and their behavior may change without notice. Register maps, protocol details and device identifiers reproduced here were observed on the specific hardware and firmware versions stated in this document and must be confirmed against the vendor's own documentation for the reader's part.

Example code

Source code supplied with this document is provided as an example, for evaluation and reference. It is not qualified for production use, has not been developed under a functional safety or security process, and is licensed under the terms accompanying it in the distribution archive.

Terms and conditions of commercial sale

Binho Inc. products are sold subject to the general terms and conditions of commercial sale published by Binho Inc., unless otherwise agreed in a valid written individual agreement, in which case only the terms of that agreement apply.

Export control

This document and the items it describes may be subject to export control regulations. Export may require prior authorization from the competent authorities.

Security

All products may be subject to unidentified vulnerabilities or may support security standards with known limitations. The customer is responsible for the design and operation of its applications and products throughout their lifecycle to reduce the effect of such vulnerabilities, and for compliance with all legal, regulatory and security requirements concerning its products.

Trademarks

All referenced brands, product names, service names and trademarks are the property of their respective owners. **Binho** and the Binho logo are trademarks of Binho Inc. **NVMe** および **NVMe-MI** は NVM Express, Inc. の商標です。**Saleae** および **Logic** は Saleae, Inc. の商標です。