注意：この日本語版文書は参考資料としてご利用ください。
最新情報は必ずオリジナルの英語版をご参照願います。

8 ビット PIC®マイクロコントローラを使った I3C®マルチプロトコル トランスレータの実装

AN5712

はじめに

著者: Irene Mary Abraham, Microchip Technology Inc.

I3C® (Improved Inter-Integrated Circuit)は、アプリケーション プロセッサへのセンサおよびデバイスの統合を改善するために MIPI® Alliance によって開発された 2 線式シリアルバス インターフェイスです。I3C は I^2C (Inter-Integrated Circuit)インターフェイスを基盤として構築され、1 つのアプリケーション内で I^2C、SPI (Serial Peripheral Interface)、UART (Universal Asynchronous Receiver Transmitter)プロコルを使用する際に頭を悩まされてきた問題を克服します。I3C バス上のデバイスは、コントローラまたはターゲット デバイスのどちらからでも通信を開始できるコントローラ/ターゲット環境内で通信します。I3C 仕様の詳細は MIPI® Alliance ウェブサイトを参照してください。

コンピューティング、通信、車載等を含む多くの業界は、従来の I^2C/SPI/UART 通信インターフェイスから高速かつシンプルな MIPI I3C プロトコルへと徐々に移行しつつあります。しかし、多くのセンサデバイスは依然として従来型プロトコルを介して通信するため、I3C には後方互換性も要求されます。複数の異なる通信インターフェイスが 1 つのシステム内に存在する場合、I3C バスを介して I^2C/SPI/UART デバイスをホストデバイスへ接続するために**マルチプロトコル トランスレータ**または **I3C-to-Serial コンバータ**が役立ちます。I3C ターゲットモジュールを備えた 8 ビット PIC®マイクロコントローラは、I3C コントローラと I^2C/SPI/UART デバイス間のブリッジとして機能します。

本書では、I3C ターゲット機能を備えた 8 ビット PIC マイクロコントローラを使ってマルチプロトコル トランスレータを実装する方法を説明します。このマルチプロトコル トランスレータを使った応用例は、MPLAB® Discover(下記リンク)から入手できます。

クリックして MPLAB® DISCOVER でサンプルコードを見る

© 2025 Microchip Technology Inc. and its subsidiaries

目次

はじめに 1
1. 関連文書 3
2. 対応デバイス 4
3. マルチプロトコル トランスレータの目的 5
3.1 混成通信バスを使う従来型システム 5
3.2 マルチプロトコル トランスレータを使ってアップグレードしたシステム 6
4. PIC® MCU を使ったマルチプロトコル トランスレータの実装 8
4.1 概要 8
4.2 マルチプロトコル トランスレータの特長 9
4.3 機能説明 10
5. ファームウェアの概要 18
6. アプリケーション デモ 20
7. まとめ 21
8. 改訂履歴 22
Microchip 社の情報 23
商標 23
法律上の注意点 23
Microchip 社のデバイスコード保護について 23

© 2025 Microchip Technology Inc. and its subsidiaries

1. 関連文書

- 8 ビット MCU 内の I3C®モジュール
- I3C®モジュールに関連するサンプルコードと参考文書
- TB3340: Using the I3C® Target Module on 8-bit PIC® Microcontrollers
- I3C ターゲット ドライバ向け MCC Melody API リファレンス

© 2025 Microchip Technology Inc. and its subsidiaries

2. 対応デバイス

マルチプロトコル トランスレータは、I3C ターゲット機能を備えると共にその他のシリアル通信プロトコル(I^2C、SPI、UART 等)もサポートする 8 ビット PIC マイクロコントローラを使って実装します。互換 PIC デバイスについては Microchip 社製品ページを参照してください。マルチプロトコル トランスレータを使って完全なシステムを実装するには、この PIC マイクロコントローラを介して I^2C/SPI/UART デバイスと通信する I3C 対応コントローラが必要です。I^2C/SPI/UART インターフェイス上の全てのデバイスは、この PIC マイクロコントローラへ接続できます。

© 2025 Microchip Technology Inc. and its subsidiaries

3. マルチプロトコル トランスレータの目的

I3C は I²C プロトコルを基盤として構築されたシリアル通信インターフェイスであり、高速/高性能/低消費電力ソリューションを提供します。I3C は I²C が持つシンプルさ、少ピン数、容易な基板設計、マルチドロップといった長所を持つだけでなく、SPI が持っている高データレート化、ピン構成のシンプル化、低消費電力化もサポートします。I3C プロトコルは、従来の通信プロトコルに利点をもたらす機能も備えています。このため、多くのコンピューティング、モバイル、車載セクタでは、システム設計を I3C ベースの通信へとアップグレードする動きが生じています。I3C は多くの業界で人気を得つつありますが、現在市場で入手可能なセンサの大部分は I²C、SPI、UART インターフェイスを介して通信します。このような状況において、システム内のホストデバイスは混成 I3C バス(I3C デバイスと I²C デバイスが混在するバス)または異なる複数の通信インターフェイスを使用する場合があります。以下では、そのようなシステムの短所と、マルチプロトコル トランスレータを使ってこの短所を克服するための方法を説明します。

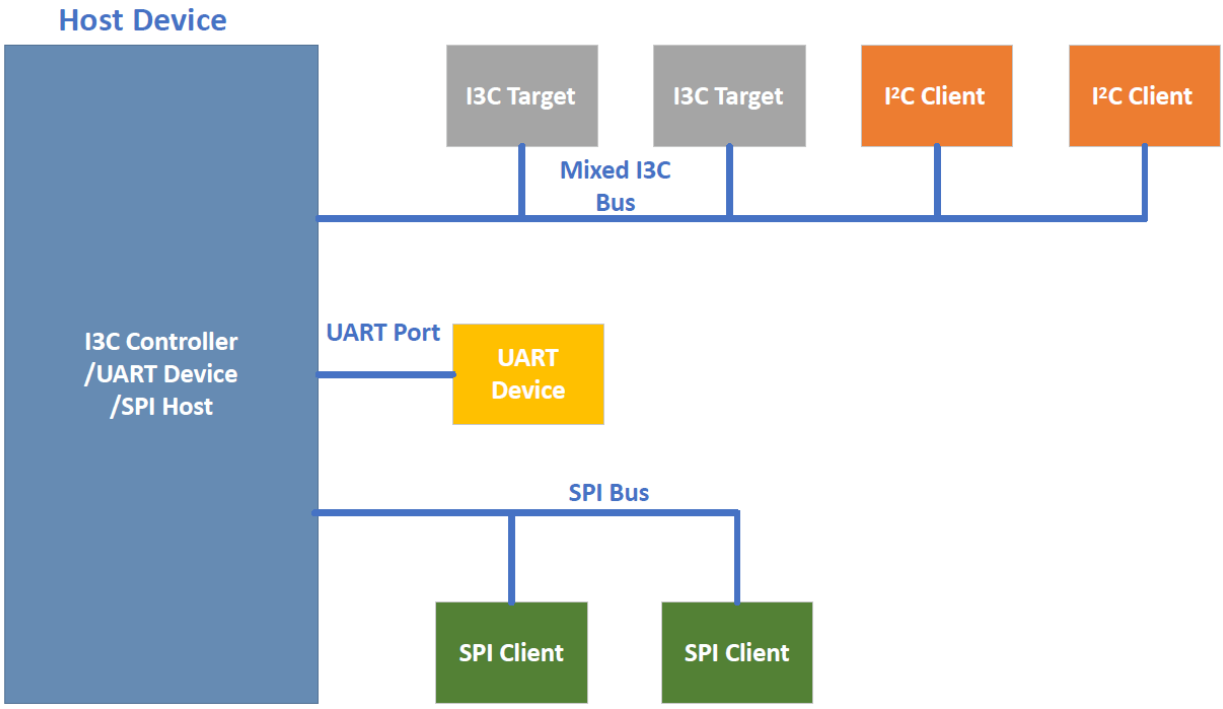
3.1 混成通信バスを使う従来型システム

図 3-1 に、I3C コントローラ機能を持つホストデバイスを使ったシステムを示します。I3C プロトコルは I²C との後方互換性を有するため、I²C クライアント デバイスは I3C バス内で共存できます。I3C は最高 12.5 MHz のデータレートをサポートしますが、バス上の I²C デバイスと通信するために I3C コントローラは通信速度を I²C 互換速度まで落とす必要があります。これによりバスの総データスループットは大きく低下します。I²C デバイスを I3C バスに追加する場合、高クロックレートの I3C トラフィックを無視するために、I²C デバイスは SDA および SCL ピンに 50 ns スパイクフィルタを備える必要があります。スパイクフィルタなしで I²C デバイスを追加した場合、I3C バス速度は最も低速の I²C クライアントの速度によって制限されます。I3C バス上の I²C デバイスでは、I3C の先進バス管理機能(インバンド割り込み、動的アドレス割り当て、ホットジョイン対応等)を使う事はできません。従って、バス管理の効率は制限され、I3C コントローラによる追加の処理が要求されます。I3C デバイスと I²C デバイスが混在するバスでは、両方のデバイスと通信する必要があるため、I3C コントローラのバス管理とソフトウェア処理が複雑になります。

加えて、ホストデバイスが SPI/UART デバイスと通信する必要がある場合、SPI バスと UART ポートを構成するためにホストデバイスは追加のピンを必要とします(図 3-1 参照)。また、ホストデバイスは SPI および UART プロトコルをサポートすると共に、それらの通信インターフェイスを管理する必要があります。I²C/SPI 接続のクライアント デバイスは、ホストデバイスへ割り込み要求を送信するために外部信号線用端子を使います。また、それらのデバイスはホストからの外部信号を使ってリセットされるため、リセット用にホスト側のピン数が増加します。混成バスを使うこの複雑なシステムは、純粋な I3C バス(I3C デバイスのみ接続したバス)と、I²C/SPI/UART バスを備えたブリッジデバイスを使う事によってシンプルにできます(3.2 マルチプロトコル トランスレータを使ってアップグレードしたシステム参照)。

© 2025 Microchip Technology Inc. and its subsidiaries

図 3-1. ホストデバイスが複数の異なる通信バスを持つ従来型システム



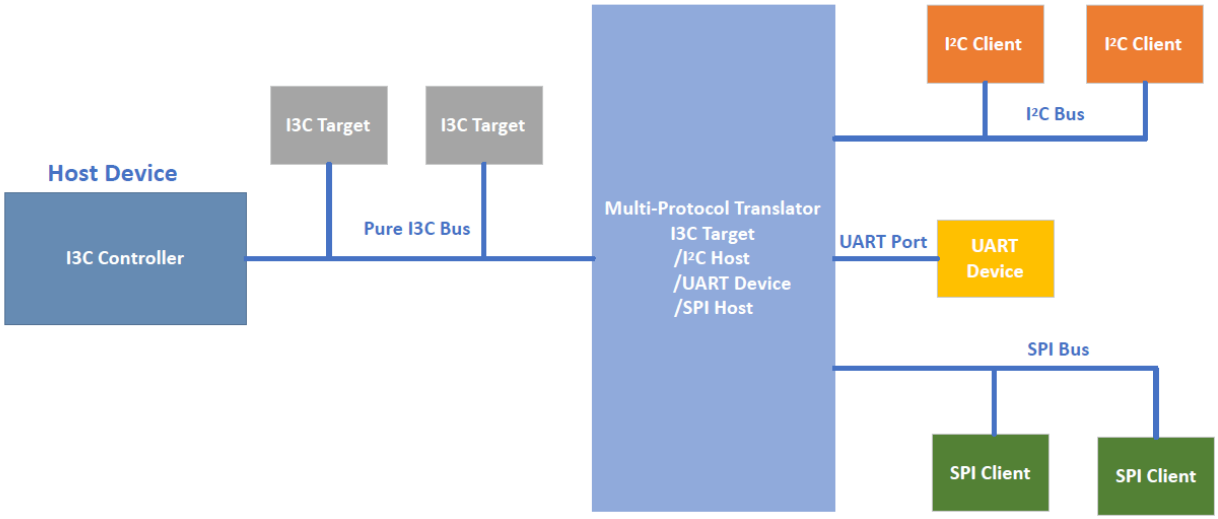
3.2 マルチプロトコル トランスレータを使ってアップグレードしたシステム

図 3-2 に、混成バス方式の欠点を克服するために役立つマルチプロトコル トランスレータを示します。このシステムには I3C ターゲット機能を備えたマイクロコントローラが追加され、プロトコル トランスレータとして機能します。ホストデバイスは I3C 対応コントローラであり、I3C 通信のみサポートします。I3C ターゲット機能と I^2C/SPI/UART インターフェイスを備えたプロトコル トランスレータは I3C to Serial コンバータとして機能します。これはブリッジデバイスとして機能し、I^2C/SPI/UART 通信デバイスを I3C コントローラから切り離します。これにより、I3C コントローラは純粋な I3C バスを維持したまま、マルチプロトコル トランスレータを介して他のプロトコルで接続されたデバイスと通信する事ができます。

ホストデバイスは純粋な I3C バスのみを使うため、I3C コントローラのバス管理とソフトウェア処理の全体的な複雑さは、異なる複数の通信インターフェイスを扱う場合に比べて軽減されます。マルチプロトコル トランスレータの I3C ターゲット モジュールは、インバンド割り込み(IBI) 機能を使って I^2C/SPI クライアント デバイスからの全ての割り込み信号を I3C コントローラへ中継するため、コントローラ側で追加のピンを使う必要はありません。I3C コントローラは、ターゲット リセットパターン機能を使ってクライアント デバイスをリセットできます。コントローラは SCL ラインと SDA ラインのみを使ってリセットパターンを送信するため、追加のピンは不要です。従って、システムはホスト側で 2 本のピン (SCL および SDA ライン) のみを必要とします。

© 2025 Microchip Technology Inc. and its subsidiaries

図 3-2. マルチプロトコル トランスレータを使ってアップグレードしたシステム



© 2025 Microchip Technology Inc. and its subsidiaries

4. PIC® MCU を使ったマルチプロトコル トランスレータの実装

4.1 概要

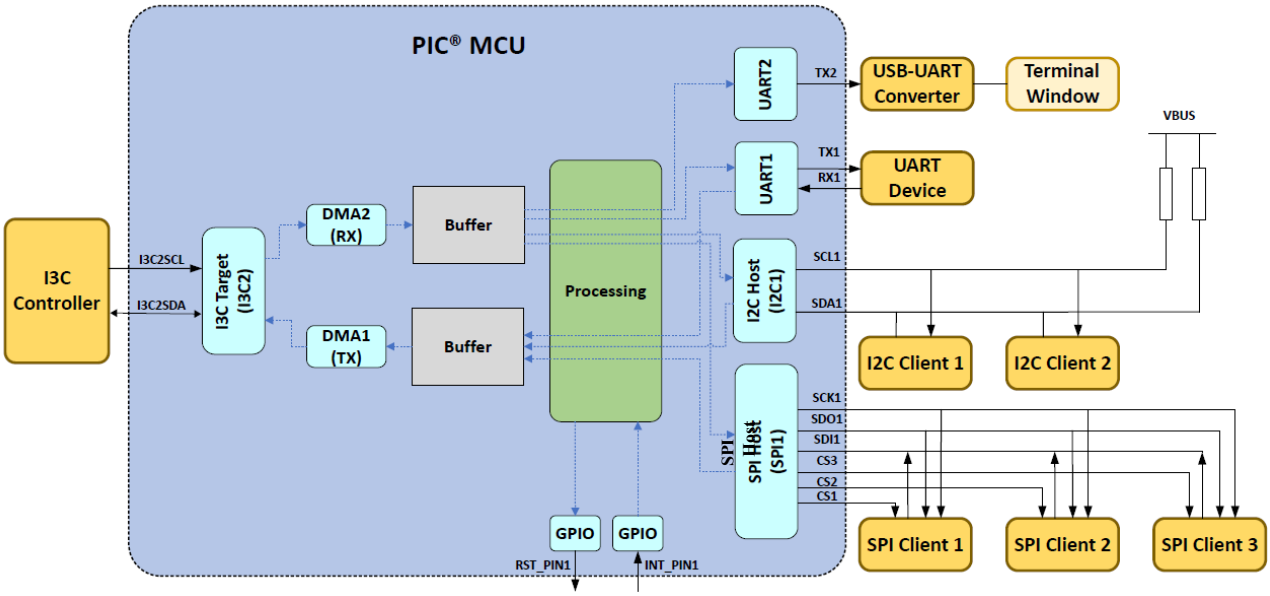
マルチプロトコル トランスレータは I3C ターゲット、I²C、SPI、UART モジュールを備えた 8 ビット PIC マイクロコントローラを使って実装します。PIC MCU は、純粋な I3C バスと I²C/SPI/UART バスの間でブリッジとして機能します。従って、この MCU は I3C ターゲット、I²C ホスト、SPI ホストとして設定されます。この MCU の UART ポートも設定されます。図 4-1 に、8 ビット PIC MCU を使って実装したマルチプロトコル トランスレータのブロック図を示します。

この実装では、マルチプロトコル トランスレータは受信データを一切処理せずに、I3C バスと I²C/SPI/UART デバイスの間でデータをそのまま転送します。I3C バスは最高 12.5 MHz の高速なデータ転送レートをサポートするため、PIC MCU は DMA チャンネルを使って I3C 送信バッファおよび受信バッファを読み書きします。 これにより、プロトコル トランスレータは高速な I3C バス周波数に追従します。プロトコル トランスレータは、コントローラからのプライベート書き込みトランザクション中に受信したデータを保存するために 1 つのソフトウェア バッファを備え、ターゲットからコントローラへ送信するインバンド割り込み(IBI)トランザクションのペイロードを保存するためにもう 1 つのバッファを備えています。マルチプロトコル トランスレータは、I²C/SPI デバイスからの割り込み信号を集約し、追加のピンを使わずに、それらを IBI トランザクションを介して I3C コントローラへ送信します。同様に、コントローラは I3C SCL ピンおよび SDA ピンでターゲット リセットパターンを送信する事により、任意の I²C/SPI クライアント デバイスをリセットできます。プロトコル トランスレータがサポート可能なクライアント デバイス上の割り込み信号とリセット信号の数は、PIC マイクロコントローラ内で利用可能な GPIO ピンの数に基づきます。

サポート可能な SPI クライアント デバイスの数は、CS (Client Select)信号向けに利用可能な I/O ピンの数によって決まります。I²C バスに接続可能な I²C クライアントの数は、サポートされるアドレス指定モードに基づきます。接続可能な UART デバイスの数は、マイクロコントローラ内で利用可能な UART モジュールの数によって決まります。

PIC MCU 上の MVIO (Multi-Voltage I/O)モジュールは、I3C バスと I²C/SPI/UART バスが異なるバス電圧で動作する事を可能にします。例えば、I3C バスは低電圧(最低 1V)で動作しながら、I²C/SPI/UART インターフェイスは I²C/SPI/UART デバイスと互換の電圧(最高 5V)で動作し続ける事ができます。

図 4-1. PIC® MCU を使って実装したマルチプロトコル トランスレータのブロック図



次表に、このマルチプロトコル トランスレータに必要な全てのピン接続を示します。これは、Discover 応用例で提供される最新ファームウェア実装に基づきます。

© 2025 Microchip Technology Inc. and its subsidiaries

名称	機能
I3C2SCL	I3C バス シリアルクロック入力
I3C2SDA	I3C バス シリアルデータ入力/出力
TX1	UART1 送信出力
RX1	UART1 受信入力
TX2	UART2 送信出力 [1]
SCK1	SPI シリアルクロック出力
SDO1	SPI シリアルデータ出力
SDI1	SPI シリアルデータ入力
CS1	SPI クライアント選択 1 出力(アクティブ Low)
CS2	SPI クライアント選択 2 出力(アクティブ Low)
CS3	SPI クライアント選択 3 出力(アクティブ Low)
SCL1	I^2C バス シリアルクロック出力
SDA1	I^2C バス シリアルデータ入力/出力
INT_PIN1	割り込みピン 1 (アクティブ High 入力): 割り込みは立ち上がりエッジで発生
RST_PIN1	リセットピン 1 (アクティブ Low 出力): ピンを Low に設定する事によりクライアント デバイスをリセット

Note:

1. デバッグメッセージの出力用に UART2 TX ピンも使います。メッセージを出力するために UART-USB コンバータを接続します。UART2 の受信機能は無効にされます。

4.2 マルチプロトコル トランスレータの特長

以下では、最新のファームウェア実装によるマルチプロトコル トランスレータの特長について説明します。詳細は Discover 応用例を参照してください。この実装はアプリケーション要件、機能、マルチプロトコル トランスレータとして使われる PIC マイクロコントローラのピン数に応じて拡張可能です。

- I3C バスの動作電圧レンジ: 0.95～1.62V
  これは Curiosity Nano ボードのレイアウトに基づいて選定されました。最高 3.6V をサポートするよう変更可能です。I3C SCL および SDA ピンは、MVIO モジュールによって給電される異なった電圧ドメインで動作します。
- I3C インターフェイスは最高 12.5 MHz で動作
- SPI、I^2C、UART バスはデバイス V_{DD} で動作
- I^2C ホスト インターフェイスは最高 100 kHz で動作
- 7 ビットの I^2C クライアント アドレス指定をサポート
- 2x UART ポート (baud レート: 9600)
- SPI Mode 0
- SPI ホスト インターフェイスは 2 MHz で動作
- 3x チップセレクト ピン: 最大で 3 つの SPI クライアント デバイスを接続可能
- 1x 汎用入出力(GPIO)ピンによる I^2C/SPI クライアント デバイスからの割り込み検出
- 1x GPIO ピンによる I^2C/SPI クライアント デバイスのリセット
- I^2C/SPI バス内で最大 500 バイトのデータをサポート
- コントローラへのデータ送信用バッファ(500 バイト)
- コントローラからのデータ受信用バッファ(503 バイト)
- エラー信号をコントローラへ送信するための 1 バイトバッファ

© 2025 Microchip Technology Inc. and its subsidiaries

4.3 機能説明

初期化後に、I3C ターゲットは I3C バスに対してホットジョイン要求を生成し、動的アドレスが割り当てられます。I3C コントローラは、プライベート書き込みトランザクションを介して I3C ターゲットへ特定のコマンドを送信する事により、I²C/SPI クライアント デバイスに対して読み出し/書き込み/リセットのいずれかを開始し、UART デバイスに対して読み出しまたは書き込みを開始します。このプライベート 書き込みトランザクションでは、ペイロードの最初のバイトはファンクション ID を格納します。ファンクション ID は I²C/SPI クライアント向けのクライアント ID、UART デバイス向けのデバイス ID、コマンドで構成されます。

表 4-1. ファンクション ID の構成

ビット	機能
b7: Res	予約済み
b6:5 コマンド[1:0]	00 - 予約済み 01 - 読み出し 10 - 書き込み 11 - クライアント リセット
b4:3 デバイス ID [1:0]	01 - UART1 10 - UART2 11 - 無効な UART チャンネル選択 00 - I²C または SPI デバイス (クライアント ID に基づく)
b:2:0 クライアント ID [2:0]	000 - I²C デバイス (次のバイトで I²C デバイスのアドレスを指定) 001、010....111 - 各 SPI デバイスの ID b0 = 1 - CS1 を選択 b1 = 1 - CS2 を選択 b2 = 1 - CS3 を選択

PIC MCU は、I3C コントローラによって送信されたファンクション ID に基づいて実行するアクションを決定します。各コマンドに対応する一連の動作については、後で詳細に説明します。例えば、I²C クライアント デバイスに対して書き込みを実行する場合、コントローラはプライベート書き込みトランザクション内でペイロードの先頭バイト(ファンクション ID = 0x40)の後に I²C クライアント アドレスと送信データを続けて送信する必要があります。マイクロコントローラは先頭バイトのファンクション ID に基づいて I²C 書き込み動作を開始します。

Note: I3C コントローラは以下の情報を知る必要があります。

- I²C クライアント アドレス
- SPI クライアントに対応する CS ピン
- リセットおよび割り込みピンに接続されている I²C/SPI クライアント デバイス
- 接続されている UART ポート
- UART 受信動作向けの Stop バイト データ

以下では、マルチプロトコル トランスレータが実行する各動作を、データシーケンス図を使って説明します。

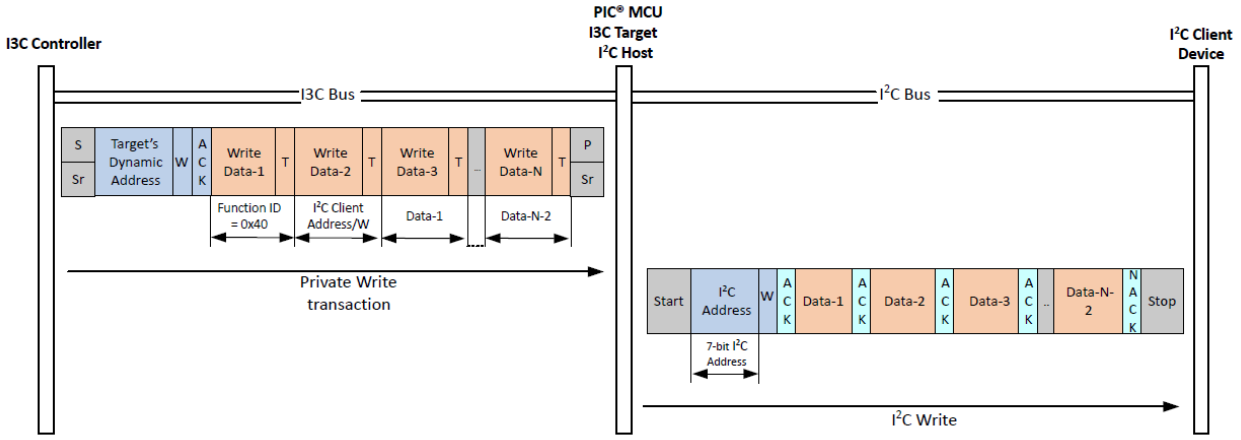
4.3.1 I²C 書き込み

I3C コントローラは、以下の手順により I²C クライアント デバイスへデータを送信します。

I3C コントローラはプライベート書き込みトランザクションのデータワード内でファンクション ID、クライアント アドレス、I²C クライアント デバイスへの書き込みデータを送信します。PIC MCU は、ファンクション ID に基づいて対応するクライアント向けに I²C 書き込みを開始します。

© 2025 Microchip Technology Inc. and its subsidiaries

図 4-2. I²C クライアントへの書き込み動作



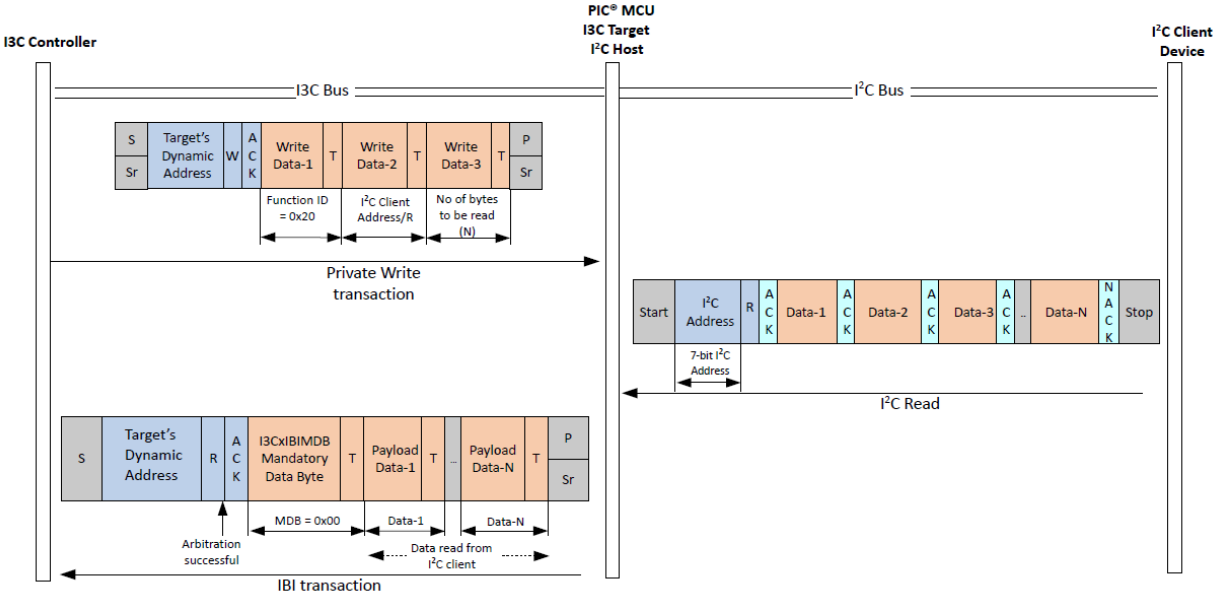
4.3.2 I²C 読み出し

I3C コントローラは、以下の手順により I²C クライアント デバイスからデータを読み出します。

I3C コントローラはプライベート書き込みトランザクションのデータワード内でファンクション ID、クライアント アドレス、I²C クライアントからの読み出しバイト数を送信します。PIC MCU は、ファンクション ID に基づいて対応するクライアント向けに I²C 読み出しを開始し、読み出したデータをバッファに保存します。

I²C 読み出しが成功すると、I3C ターゲットは IBI 要求を I3C コントローラへ送信し、そのペイロードにクライアントから受信したデータを格納します。

図 4-3. I²C クライアントからの読み出し動作

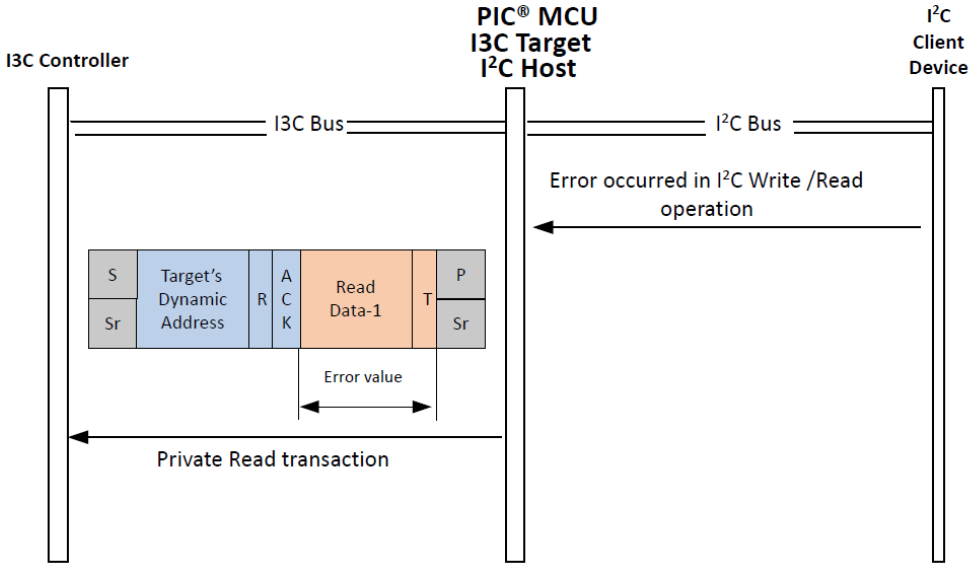


4.3.3 I²C 読み出し/書き込み中のエラー報告

I²C の読み出し/書き込み動作中にエラーが発生した場合、以下の手順により、エラー信号(1 バイトデータ)がプライベート読み出しトランザクションを介してコントローラへ送信されます。I3C ターゲットはエラー信号データをバッファ内で保持し、コントローラからのプライベート読み出し要求を待機します。コントローラは、直前の I²C トランザクションのステータスを取得するために、プライベート読み出しトランザクションを要求します。

© 2025 Microchip Technology Inc. and its subsidiaries

図 4-4. I²C 読み/書き動作中に発生したエラーの報告動作



エラー信号値	エラーの内容
0x1	I²C バスでタイムアウト エラーが発生した
0x2	I²C 書き込み中にタイムアウト エラーが発生した
0x3	I²C 読み出し中にタイムアウト エラーが発生した
0x4	I²C クライアントがアドレスに対して NACK を返した
0x5	I²C クライアントがデータに対して NACK を返した
0x6	I²C バスで衝突エラーが発生した

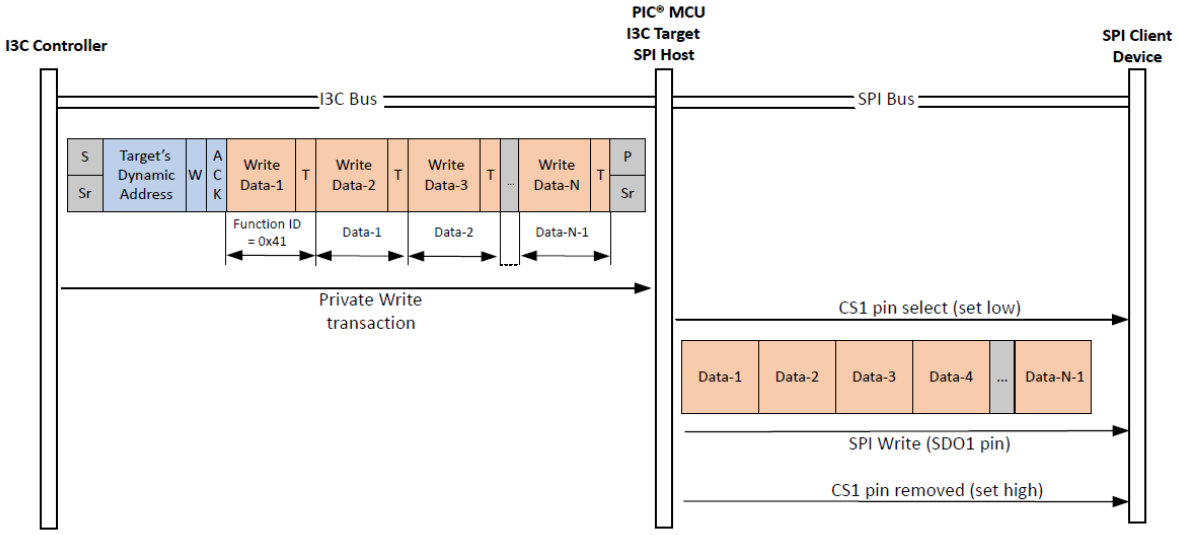
4.3.4 SPI 書き込み

I3C コントローラは、以下の手順により SPI クライアント デバイスにデータを書き込みます。

I3C コントローラはプライベート書き込みトランザクションのデータワード内でファンクション ID とクライアント デバイスへの書き込みデータを送信します。PIC MCU は、ファンクション ID 内のクライアント ID に基づいて SPI クライアント デバイスを選択(CS ピンを Low に設定)し、データの書き込みを開始します。書き込みが完了すると、SPI クライアントの選択が解除(CS ピンが High に設定)されます。

© 2025 Microchip Technology Inc. and its subsidiaries

図 4-5. SPI クライアントへの書き込み動作



4.3.5 SPI 読み出し

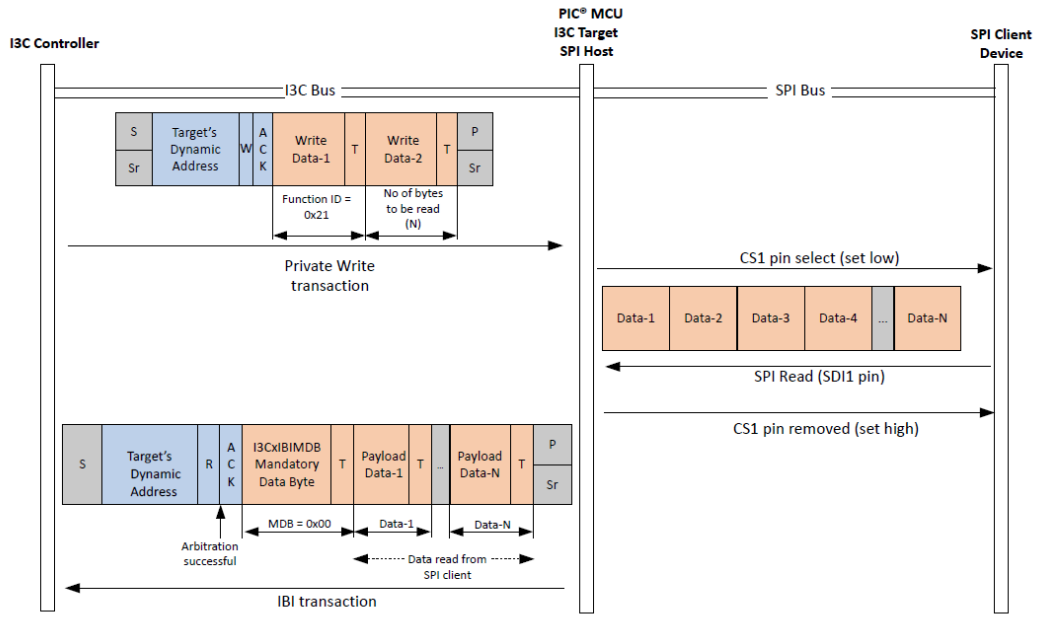
I3C コントローラは、以下の手順により SPI クライアント デバイスからデータを読み出します。

I3C コントローラは、プライベート書き込みトランザクションのデータワード内でファンクション ID と SPI クライアントからの読み出しバイト数を送信します。PIC MCU は、ファンクション ID 内のクライアント ID に基づいて SPI クライアント デバイスを選択(CS ピンを Low に設定)します。I3C コントローラは、そのクライアントからの SPI 読み出しを開始し、データをバッファに保存します。その後に、SPI クライアントの選択が解除(CS ピンが High に設定)されます。

SPI 読み出しが完了すると、I3C ターゲットは IBI 要求を I3C コントローラへ送信し、そのペイロードにクライアント デバイスからの受信データを格納します。

© 2025 Microchip Technology Inc. and its subsidiaries

図 4-6. SPI クライアントからの読み出し動作

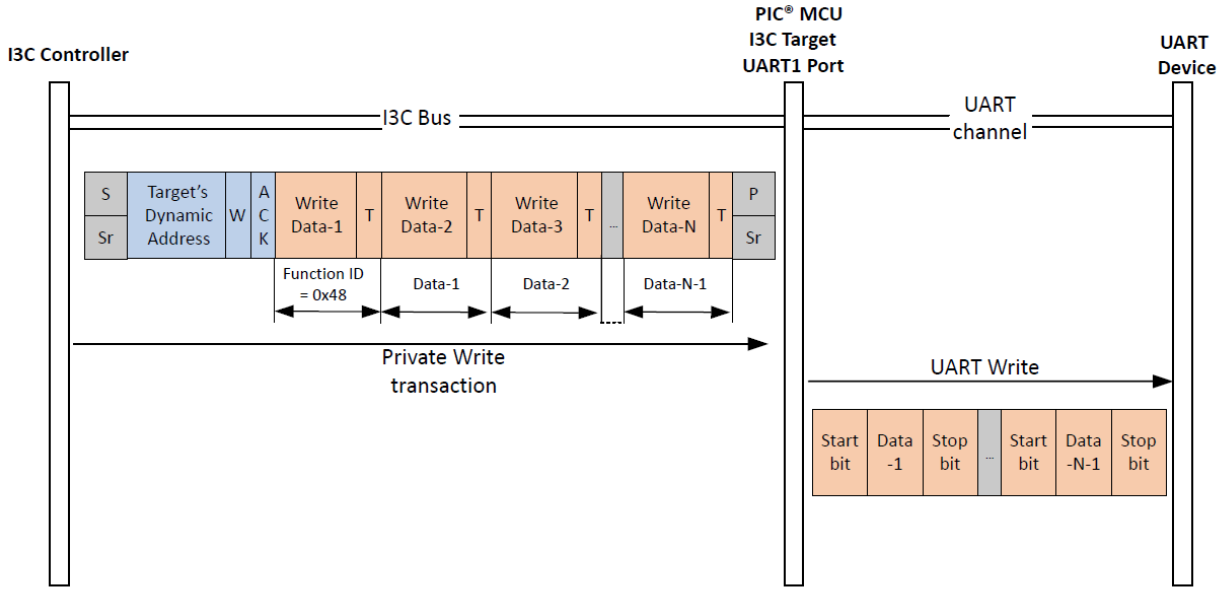


4.3.6 UART 書き込み

I3C コントローラは、以下の手順により UART ポートに接続された UART デバイスへデータを書き込みます。

I3C コントローラは、プライベート書き込みトランザクションのデータワード内でファンクション ID と UART チャンネルへの書き込みデータを送信します。PIC MCU は、ファンクション ID に基づいて対応する UART チャンネル向けに UART 書き込み動作を開始します。

図 4-7. UART 書き込み動作



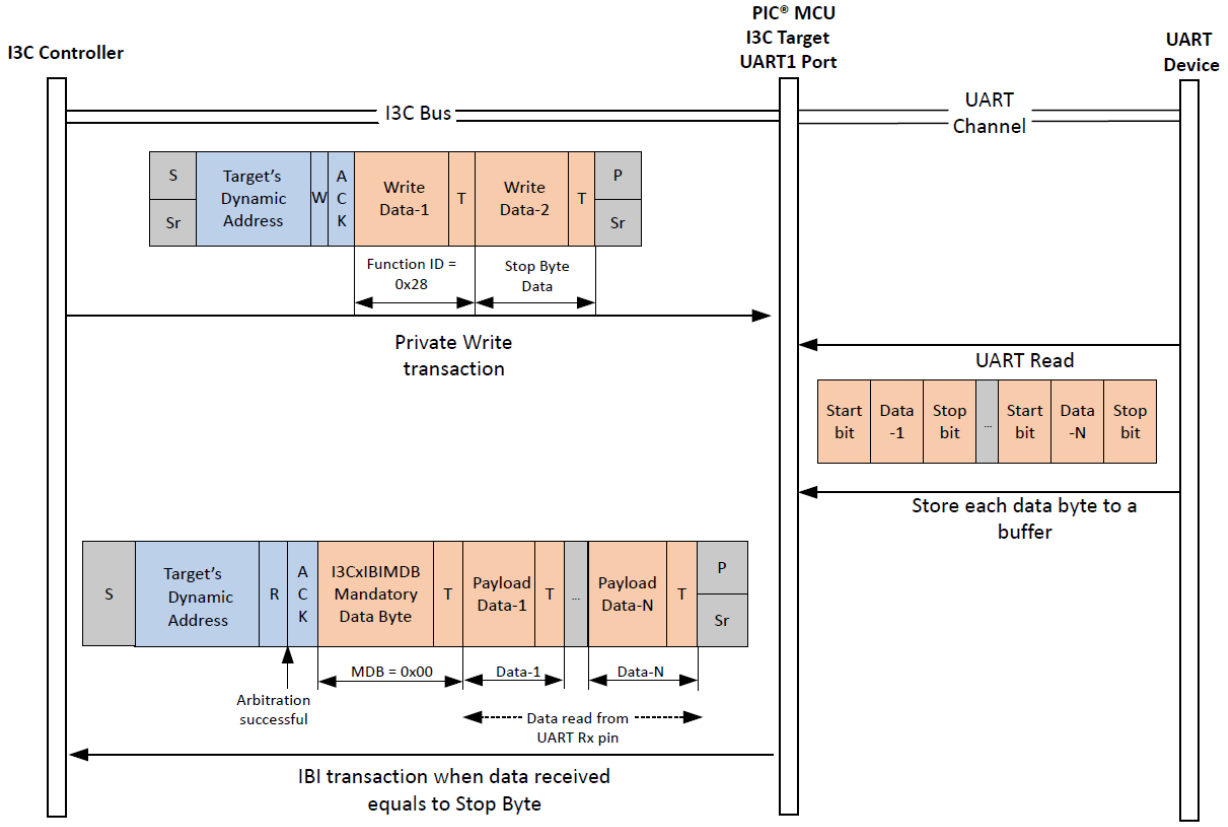
© 2025 Microchip Technology Inc. and its subsidiaries

4.3.7 UART 読み出し

I3C コントローラは、以下の手順により UART デバイスからデータを読み出します。

PIC MCU は、UART ポートの RX ラインを介して UART デバイスから連続的にデータを受信します。この実装では、Stop バイトはデータ転送の終了を示すターミネータとして使われます。I3C コントローラはこの Stop バイトを送信する必要があります。I3C コントローラは最初にプライベート書き込みトランザクションのデータワード内でファンクション ID と Stop バイトを送信します。I²C/SPI 読み出しとは異なり、I3C コントローラは UART 読み出しを開始するためにプライベート書き込みトランザクションを常に送信する必要はありません。このトランザクションは 1 度だけ、またはユーザが Stop バイト値を変更する必要がある時にのみ送信できます。UART RX ピンで受信したデータはバッファに保存されます。この受信データが Stop バイト値と等しい場合、プロトコル トランスレータは IBI トランザクションを介してバッファに保存されているデータを I3C コントローラへ送信します。

図 4-8. UART 読み出し動作

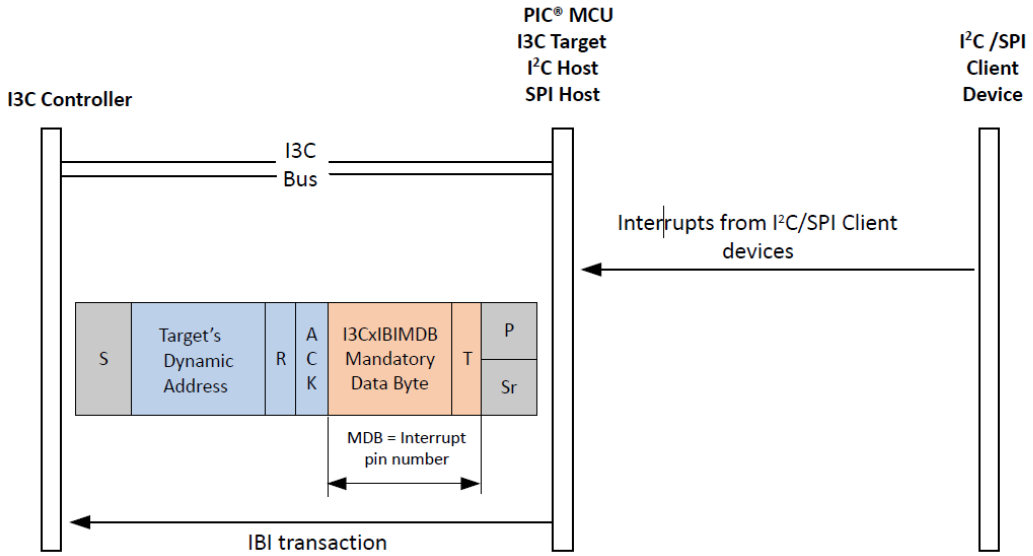


4.3.8 I²C/SPI クライアント デバイスからの割り込み

PIC MCU は、I²C/SPI クライアント デバイスからの割り込みを監視するための専用ピンを提供します。これらの GPIO には、クライアント デバイス上の割り込みピンを接続できます。このピンで立ち上がりエッジが検出されると、I3C ターゲット モジュールは IBI 要求を I3C コントローラへ送信します。IBI ペイロード内の MDB (Mandatory Byte)は割り込みピン番号に関する情報を格納し、どのクライアント デバイスから割り込みが発生したのかをコントローラに伝えます。この実装では、1 つの割り込み信号向けに 1 本のピンのみが使われます。クライアント デバイスからの割り込み信号の数は、PIC MCU 内で利用可能なピンの数に応じて増やす事ができます。

© 2025 Microchip Technology Inc. and its subsidiaries

図 4-9. クライアント デバイスからの割り込み信号の受信動作

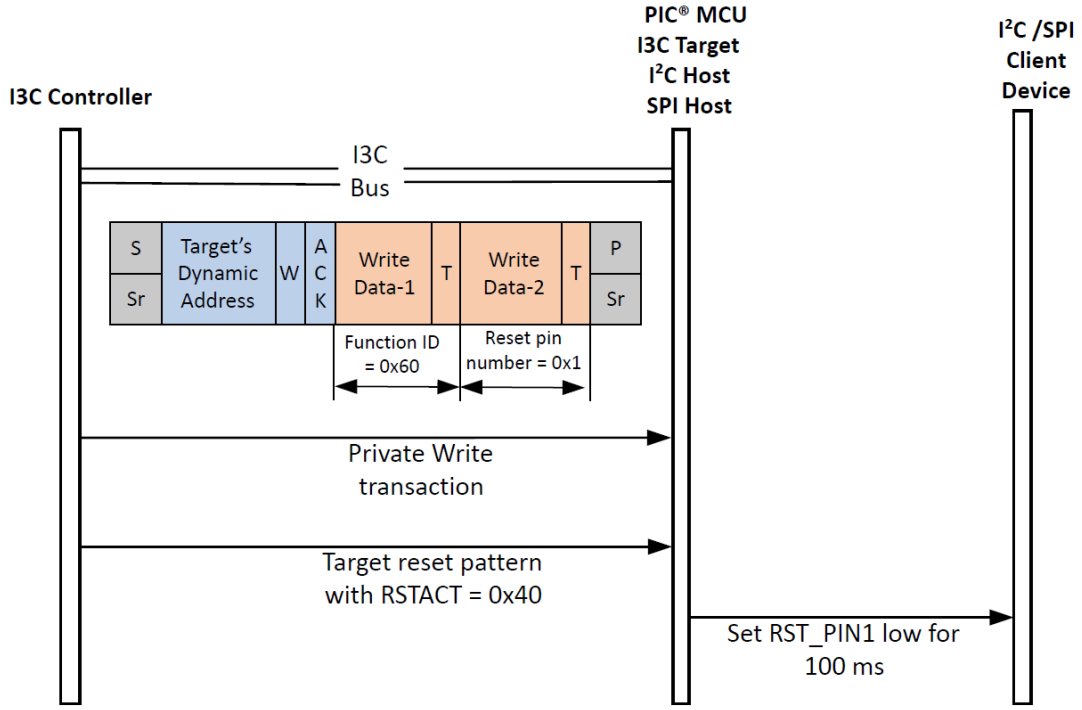


MDB 値	内容
0x1	INT_PIN1 に接続されたクライアント割り込みピンで割り込みが発生した

4.3.9 I²C/SPI クライアント デバイスのリセット

PIC MCU は、I²C/SPI クライアント デバイスをリセットするための専用ピンを提供します。これらの GPIO ピンには、クライアント デバイス上のリセットピンを接続できます。リセットピンの既定値状態は High です。実装内のリセットピンの数は、PIC MCU 内で利用可能なピンの数に応じて増やす事ができます。

図 4-10. クライアント デバイスのリセット動作



© 2025 Microchip Technology Inc. and its subsidiaries

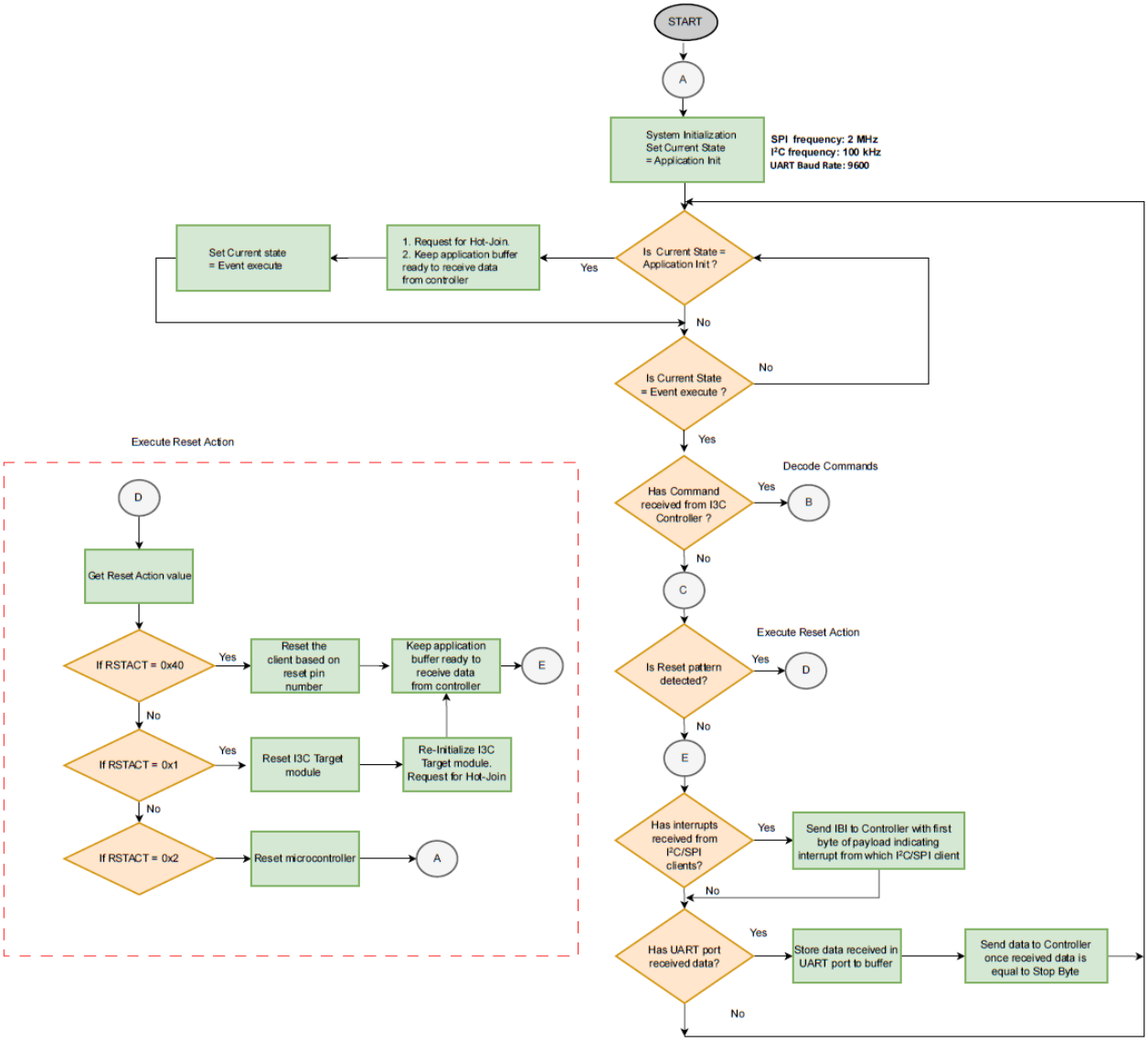
リセットピン番号	内容
0x1	RST ピンが RST_PIN1 に接続されているクライアントをリセットする

ターゲット リセットパターンが検出されると、PIC MCU はターゲット リセット アクション(RSTACT)の値に基づいてリセットを実行します。

RSTACT	内容
0x40	リセットピン番号に基づいてクライアント デバイスをリセットする(リセットピンを 100 ms 間 Low に設定)
0x01	I3C ターゲット モジュールのみリセットする
0x02	マイクロコントローラをリセットする

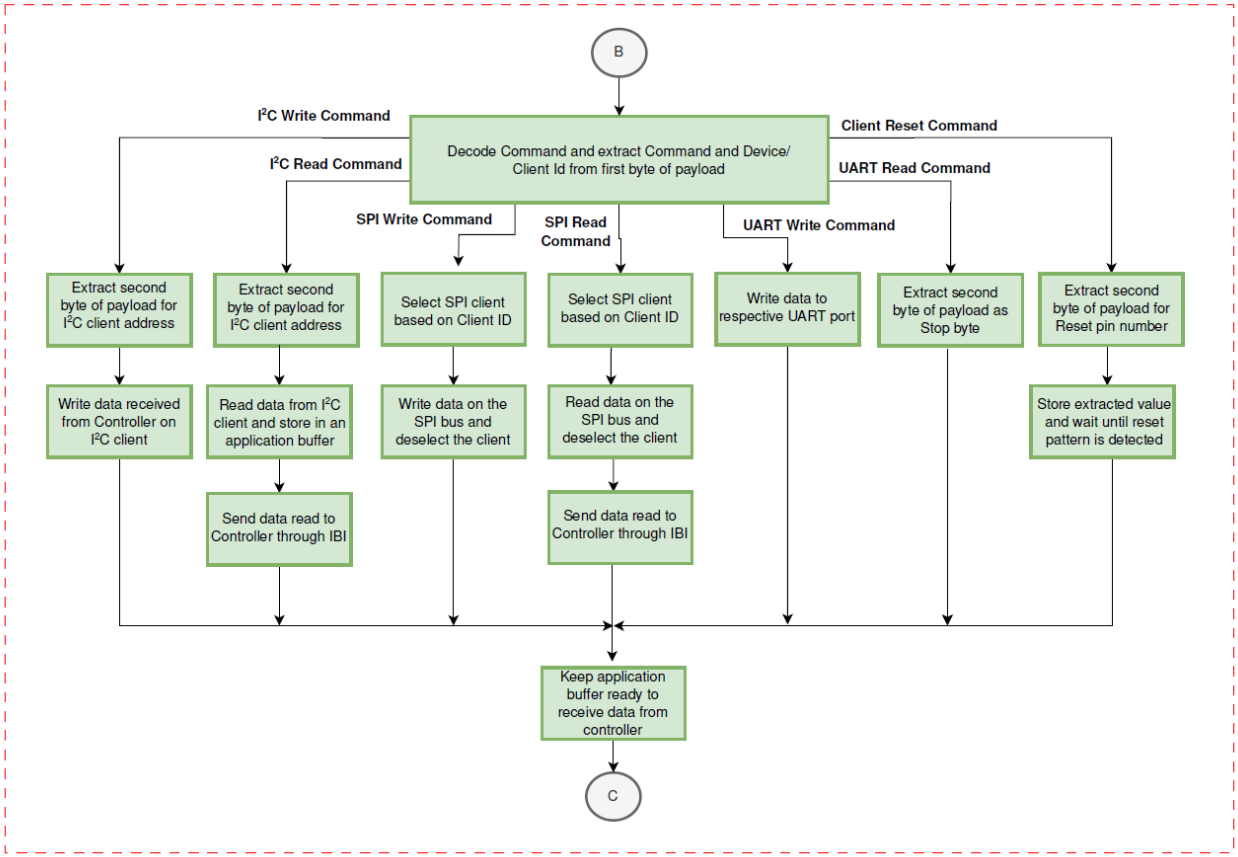
© 2025 Microchip Technology Inc. and its subsidiaries

5. ファームウェアの概要



© 2025 Microchip Technology Inc. and its subsidiaries

Decode Commands



© 2025 Microchip Technology Inc. and its subsidiaries

6. アプリケーション デモ

アプリケーションを実装する際に、ユーザは自由に I3C コントローラを選定できます。プロトコル トランスレータはデータを一切処理しないため、I^2C、SPI、UART バスを介して通信する全てのセンサデバイスをプロトコル トランスレータに接続できます。下記のサンプルコードは、8 ビット PIC マイクロコントローラ PIC18F16Q20 を使って実装されたアプリケーションを実行します。ハードウェアのセットアップと動作手順の詳細は、下のリンクを参照してください。

クリックして MPLAB® DISCOVER でサンプルコードを見る

© 2025 Microchip Technology Inc. and its subsidiaries

7. まとめ

本書では、8 ビット PIC マイクロコントローラを使ったマルチプロトコル トランスレータについて説明しました。また、I3C コントローラと従来型シリアル プロトコル インターフェイスの間でブリッジとして機能するプロトコル トランスレータの必要性を強調しました。さらに、プロトコル トランスレータの動作を詳細に説明し、シリアル プロトコルをサポートする 8 ビット PIC マイクロコントローラを使ってマルチプロトコル トランスレータを実装する方法を示しました。

© 2025 Microchip Technology Inc. and its subsidiaries

8. 改訂履歴

リビジョン	日付	内容
A	2025 年 1 月	本書は初版です。

© 2025 Microchip Technology Inc. and its subsidiaries

Microchip 社の情報

商標

Microchip 社の名称とロゴ、「M」ロゴ、その他の名称、ロゴ、ブランドは米国および/またはその他の国における Microchip Technology Incorporated またはその支社および/または子会社の登録または未登録商標です。これらの Microchip 社商標に関する情報は、https://www.microchip.com/en-us/about/legal-information/microchip-trademarks でご覧になれます。

ISBN: 979-8-3371-1355-5

法律上の注意点

本書および本書に記載されている情報は、Microchip 社製品を設計、テスト、お客様のアプリケーションと統合する目的を含め、Microchip 社製品に対してのみ使う事ができます。それ以外の方法でこの情報を使用する事はこれらの条項に違反します。デバイス アプリケーションの情報は、ユーザの便宜のためにのみ提供されるものであり、更新によって変更となる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。その他のサポートについては、弊社または代理店にお問い合わせになるか、www.microchip.com/en-us/support/design-help/ client-support-services をご覧ください。

Microchip 社は本書の情報を「現状のまま」で提供しています。Microchip 社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。

いかなる場合も Microchip 社は、本情報またはその使用に関連する間接的、特殊的、懲罰的、偶発、的または必然的損失、損害、費用、経費のいかんにかかわらず、また Microchip 社がそのような損害が生じる可能性について報告を受けていた場合あるいは損害が予測可能であった場合でも、一切の責任を負いません。法律で認められる最大限の範囲を適用しようとも、本情報またはその使用に関連する一切の申し立てに対する Microchip 社の責任限度額は、使用者が当該情報に関連して Microchip 社に直接支払った額を超えません。

Microchip 社の明示的な書面による承認なしに、生命維持装置あるいは生命安全用途に Microchip 社の製品を使う事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、Microchip 社は擁護され、免責され、損害をうけない事に同意するものとします。特に明記しない場合、暗黙的あるいは明示的を問わず、Microchip 社が知的財産権を保有しているライセンスは一切譲渡されません。

Microchip 社のデバイスコード保護について

Microchip 社製品のコード保護機能について以下の点にご注意ください。

- Microchip 社製品は、該当する Microchip 社データシートに記載の仕様を満たしています。
- Microchip 社では、通常の条件ならびに仕様に従って使った場合、Microchip 社製品のセキュリティ レベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- Microchip 社はその知的財産権を重視し、積極的に保護しています。Microchip 社製品のコード保護機能の侵害は固く禁じられており、デジタル ミレニアム著作権法に違反します。
- Microchip 社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、Microchip 社が製品を「解読不能」として保証するものではありません。コード保護機能は常に進歩しています。Microchip 社では、常に製品のコード保護機能の改善に取り組んでいます。

© 2025 Microchip Technology Inc. and its subsidiaries