参考資料

*Subsystem Design*

# *MSPM0 MCU での SPI から CAN へのブリッジ*

*Kalicheti Vishnuvardhan Reddy, Ashwini Gopinath, Shubham Saurabh*

## 目次

**1 はじめに**.....1
1.1 サポート機能.....2
1.2 CAN フレーム フォーマット.....2
1.3 SPI メッセージフレームフォーマット.....3
**2 実装**.....4
2.1 SPI メッセージ形式.....4
2.2 タイムアウト機能.....10
2.3 エラー表示.....10
2.4 ビジーステータスの表示.....10
2.5 メッセージ RAM の構成.....11
2.6 テスト環境.....12
**3 参考資料**.....12

## 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

コントローラエリアネットワーク (CAN) とシリアルペリフェラルインターフェイス (SPI) は、最新のマイクロコントローラユニット (MCU) で広く採用されている 2 つの通信プロトコルです。CAN プロトコルは、堅牢なエラー処理と信頼性により車載アプリケーションで優位性がありますが、多くのローエンドマイコンやセンサは CAN でサポートできません。一方、SPI は、マイコンデバイス全体でほぼ汎用的に使用されています。この技術的なギャップを埋めるため、SPI-CAN インターフェイスは SPI ベースのデバイスと CAN ネットワーク間のシームレスな通信を可能にし、低コストのセンサやマイクロコントローラを CAN ベースのシステムに統合します。

SPI-CAN ブリッジ機能は、MSPM0G3507 デバイスを使用して実装されます。このデバイスは、外部 SPI コントローラと CAN ネットワークの間のインターフェイスとして機能します。外部 SPI コントローラが通信フレームを開始したとき、MSPM0G3507 はこれらを CAN プロトコル命令に変換し、CAN ネットワークに送信します。

図 1-1 に、MSPM0G3507 デバイスと外部インターフェイスとの相互接続を示すシステムアーキテクチャを示します。このデバイスは、4 線式 SPI プロトコルを使用して外部コントローラと通信しつつ、CAN 物理層送受信回路 (CAN-PHY) を介して CAN バスに接続されます。



English Document: SLAAEQ4
Copyright © 2025 Texas Instruments Incorporated

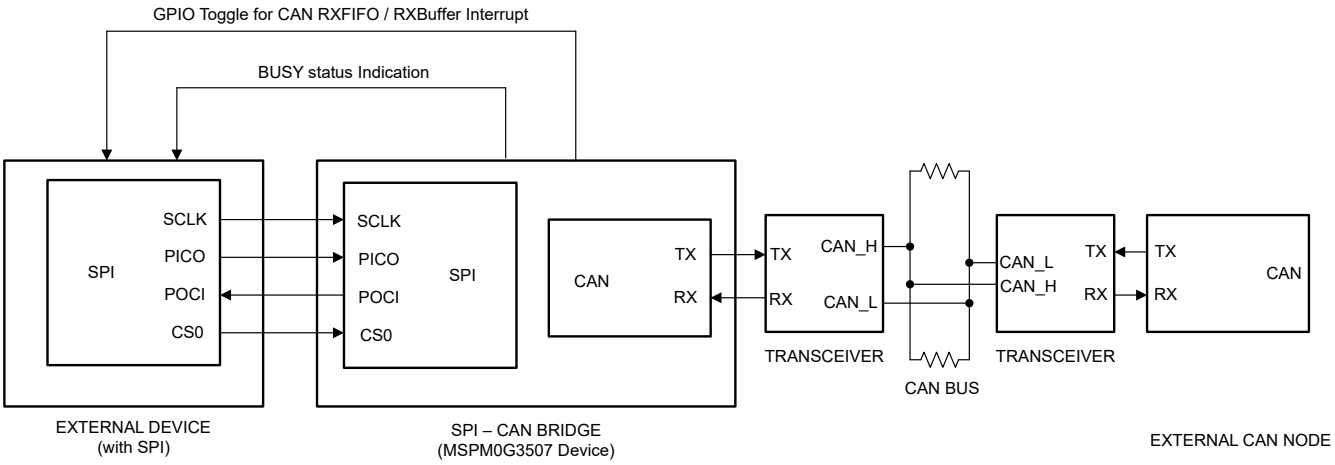



**図 1-1. SPI - CAN ブリッジのブロック図**

## 1.1 サポート機能

CAN FD のサポート

- 最大 1MbpS 公称ビット レート、5MbpS データ ビット レート
- 受信専用および送信専用バッファ (各側に 4 つ)
- 送信および受信 FIFO (それぞれ深度 4)
- 拡張および標準フィルタエレメント (送信側および受信側に各 8 個)

SPI

- 最大 8MHz の周波数をサポート
- 4 線式 Motorola Phase0 Polarity0 モード向けに実装

エラー表示

- 命令エラーおよびタイムアウトステータスの表示

SPI から CAN へのブリッジは、Classical CAN と CAN-FD の各フォーマットをサポートします。CAN と CAN-FD のフレーム フォーマットの違いについては、表 1-1 に詳述します。

**表 1-1. CAN と CAN-FD の違い**

| 機能 | CAN 2.0 | CAN 2.0B | CAN FD |
|---|---|---|---|
| 識別子: | 標準 (11 ビット) | 標準 (11 ビット)<br>拡張 (29 ビット) | 標準 (11 ビット)<br>拡張 (29 ビット) |
| データ レート | 最大 1MbpS | 最大 1MbpS | 最大 8MbpS |
| ペイロード | 0 - 8 バイト | 0 - 8 バイト | 0 - 64 バイト |

## 1.2 CAN フレーム フォーマット

図 1-2、図 1-3、図 1-4 に CAN-FD のフレーム フォーマットを示します。ビット値 *1* はリセッシブとみなされ、値 *0* はドミナントとみなされます。

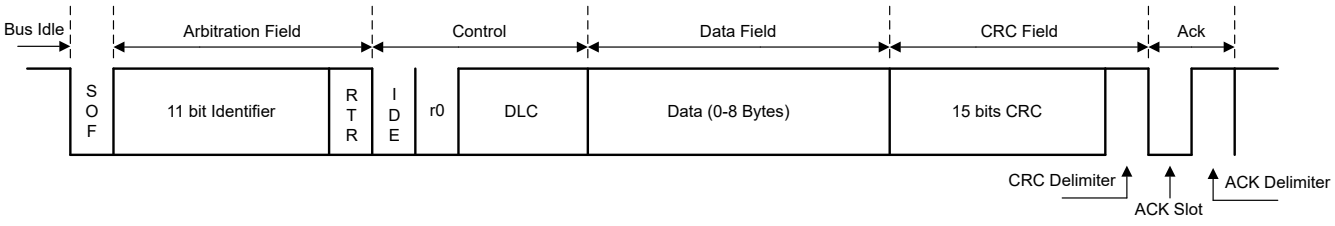



**図 1-2. CAN 2.0A フレームフォーマット**


Copyright © 2025 Texas Instruments Incorporated

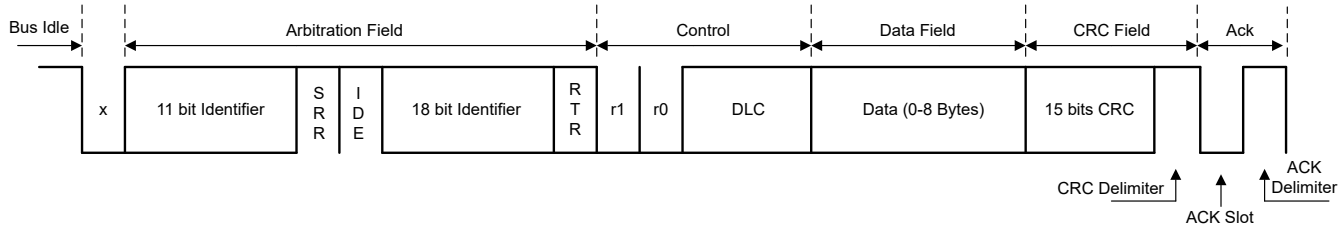



**図 1-3. CAN 2.0B フレームフォーマット**

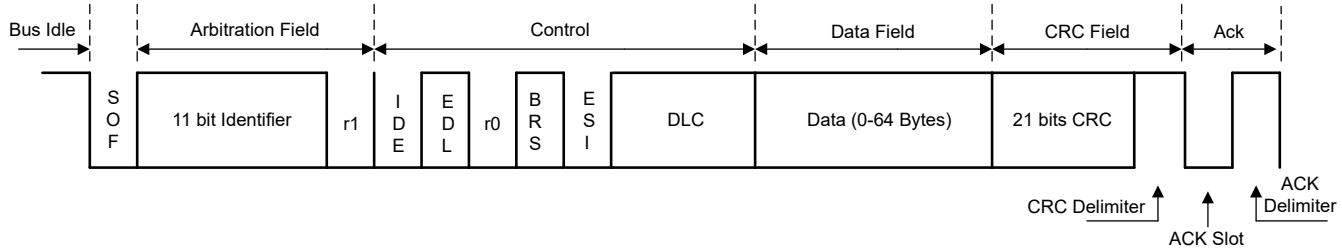



**図 1-4. CAN-FD フレームフォーマット**

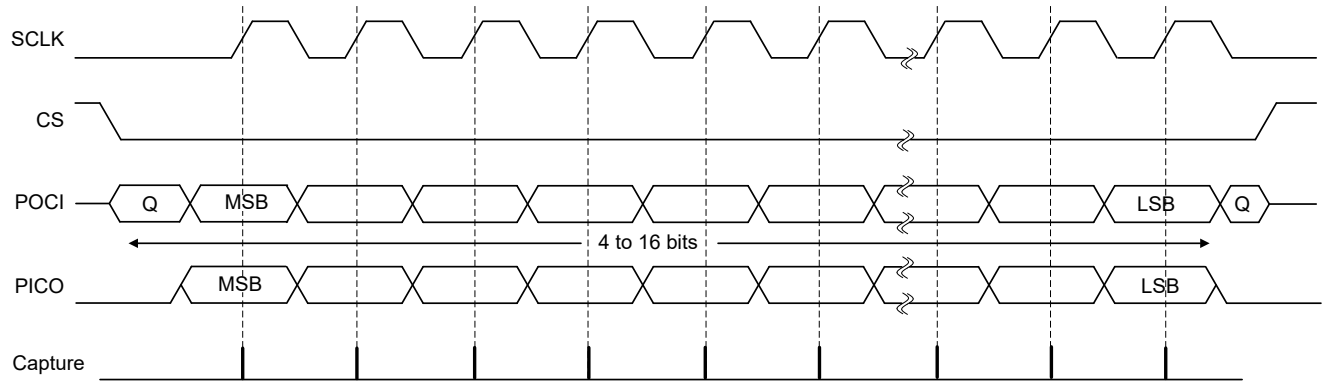
**表 1-2. CAN-FD フレームのビットと定義**

| ビット | 説明 |
|---|---|
| SOF | フレームの開始 |
| IDE | 識別子拡張 |
| FDF | FD フォーマットインジケータ |
| BRS | ビットレートのスイッチング |
| ESI | エラーステータスインジケータ |
| DLC | データ長コード |
| CRC | 巡回冗長検査。 |

- IDE = リセッシブ、拡張 (29 ビット) 識別子を示す、IDE = ドミナント、標準 (11 ビット) 識別子を示す。
- FDF = リセッシブ、CAN-FD 形式を示す、FDF = ドミナント、CAN フレームを示す
- BRS = リセッシブ、ビットレートのより高速なスイッチングを示す、BRS = ドミナントで、レートの変化がないことを示す

## 1.3 SPI メッセージフレームフォーマット

SPI は、4 ラインの同期通信プロトコルです (オプションのチップセレクト (CS) オプション、クロック (SCLK)、ペリフェラル出力コントローラ入力 (POCI)、ペリフェラル入力およびコントローラ出力 (PICO)。SPI-CAN ブリッジは、最大 8MHz までの SPI データ転送をサポートします。図 1-5 に、SPI フレームの例を示します。




Q is undefined

**図 1-5. SPO = 0 および SPH = 0 での Motorola SPI フレームフォーマット**


Copyright © 2025 Texas Instruments Incorporated

# 2 実装

SPI-CAN ブリッジアプリケーションは、次の 2 つのステージで構成されています。

1. MSPM0G3507 は、ペリフェラルモードで動作する SPI とトランザクション開始モードで動作する CAN でセットアップされています。
2. CAN パラメータと設定は、外部マスタコントローラからの SPI コマンドを使用してリモートで構成されます。

## 2.1 SPI メッセージ形式

図 2-1 に、SPI メッセージフレームの例を示します。各メッセージフレームは、以下のように分割されます。

- 命令: 16 ビットフレームの下位 8 ビットが使用されます。
- アドレス オフセット:16 ビット (アドレスの下位 16 ビットのみを送信)。レジスタについての記述とそのアドレスについては、「MSPM0G3507 テクニカルリファレンスマニュアル」を参照してください。
- データマスク:32 ビット – LSB ファーストで 2 つの 16 ビットフレーム。
- データ:32 ビット – LSB ファーストで 2 つの 16 ビットフレーム。

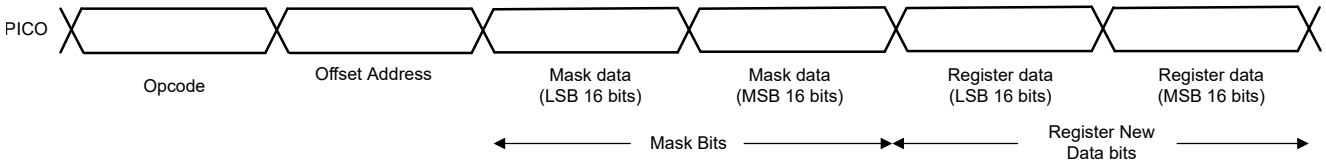



**図 2-1. SPI メッセージ形式**

### *2.1.1 SPI コマンド*

CAN モジュールの構成でサポートされるコマンドを表 2-1 に一覧します。

**表 2-1. SPI コマンド**

| 手順 | Opcode | オフセット アドレス | データ サイズ | マスクデータ LSB | マスクデータ MSB | データ LSB | データ MSB | 詳細 |
|---|---|---|---|---|---|---|---|---|
| リセット | 16'h0020 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | CAN モジュールをリセットする |
| 電源有効化と CAN 構成 | 16'h0030 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | CAN モジュールの電源有効化および CAN メッセージ RAM の構成 |
| 書き込みレジスタ | 16'h0080 | CAN ベースアドレスからのレジスタオフセット | 未使用 | 未使用 | 未使用 | LSB 16 ビット | MSB 16 ビット | CAN レジスタにデータを書き込む |
| 読み取りレジスタ | 16'h0090 | CAN ベースアドレスからのレジスタオフセット | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | CAN レジスタからデータを読み取る |
| ビット変更レジスタ | 16'h0010 | CAN ベースアドレスからのレジスタオフセット | 未使用 | LSB データ向けマスクビット | MSB データ向けマスクビット | LSB 16 ビット | MSB 16 ビット | CAN レジスタの個別ビットを変更する |
| 送信バッファをロードする | 16'h0040 | 未使用 | データサイズ (bytes) | 未使用 | 未使用 | 未使用 | 未使用 | CAN フレームを CAN TX バッファに書き込む |
| 送信 FIFO をロードする | 16'h0050 | 未使用 | データサイズ (bytes) | 未使用 | 未使用 | 未使用 | 未使用 | CAN フレームを CAN TX FIFO に書き込む |
| 送信バッファの送信リクエスト | 16'h0060 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | TX バッファ要素へのバッファ追加リクエスト |
| 送信 FIFO の送信リクエスト | 16'h0070 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | 未使用 | TXFIFO 要素へのバッファ追加リクエスト |
| 受信 FIFO の読み出し | 16'h00B0 | 未使用 | データサイズ (bytes) | 未使用 | 未使用 | 未使用 | 未使用 | RXFIFO からのデータの読み取り |


English Document: SLAAEQ4
Copyright © 2025 Texas Instruments Incorporated

表 2-1. SPI コマンド (続き)

| 手順 | Opcode | オフセット アドレス | データ サイズ | マスクデータ LSB | マスクデータ MSB | データ LSB | データ MSB | 詳細 |
|---|---|---|---|---|---|---|---|---|
| 受信バッファの読み取り | 16'h00D0 | 未使用 | データサイズ (bytes) | 未使用 | 未使用 | 未使用 | 未使用 | RXBUFFER からのデータの読み取り |
| ステータスレジスタの読み取り | 16'h00F0 | 未使用 | データサイズ (bytes) | 未使用 | 未使用 | 未使用 | 未使用 | 受信ステータスビットの読み取り |

レジスタの読み取り、受信 FIFO の読み取り、受信バッファの読み取り、ステータスレジスタの読み取り命令の後は、受信データの取得命令が続く必要があります。

### 2.1.2 命令セット

**ビット変更命令**

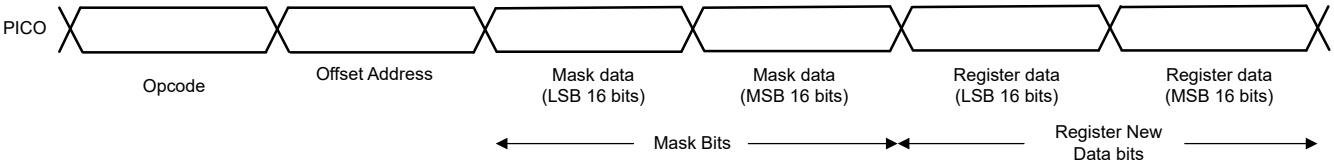



図 2-2. ビット変更命令フォーマット

Opcode : 0b0000000000010000

ビット変更命令は、特定の CAN レジスタの個々のビットをセットまたはクリアするために使用されます。MASK ビットは、レジスタ内の変更可能なビットを特定します。マスクのビット 1 を使用すると、レジスタの対応するビットを変更でき、ビット 0 を使用するとビットの変更を阻止できます。

**リセット命令**

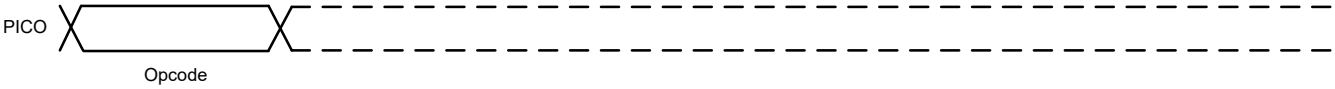



図 2-3. リセット命令のフォーマット

Opcode : 0b0000000000100000

リセット命令は、SPI–CAN ブリッジデバイスの CAN モジュールをリセットします。CAN モジュールをリセットするには、ユーザーは、SPI プロトコルを通じて対応する opcode を送信する必要があります。

**電源有効化と CAN 構成命令:**

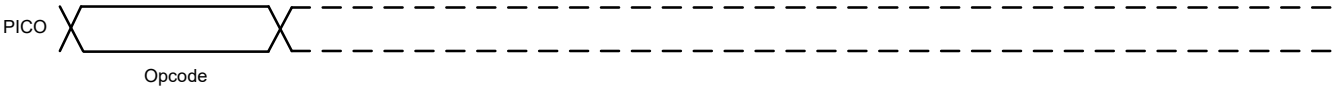



図 2-4. 電源有効化と CAN 構成命令のフォーマット

Opcode : 0b0000000000110000

電源有効化および CAN 構成命令は、ブリッジデバイスの CAN モジュールに電力を供給するために使用されます。また、CAN モジュールが有効になった後で、事前定義された CAN メッセージ RAM 構成が行われます。

**TX バッファ読み込み命令:**

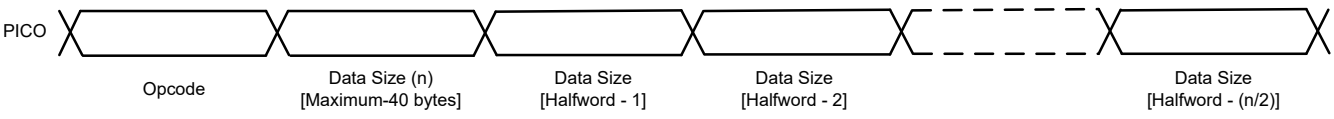



図 2-5. TX バッファ読み込み命令のフォーマット


Copyright © 2025 Texas Instruments Incorporated

Opcode：0b00000000010000ab

TX バッファ読み込み命令は、選択した送信バッファに CAN フレームを読み込むために使用されます。送信バッファの選択は、opcode でエンコードされます。4 つの送信バッファのうちの 1 つは、命令フィールドの a、b ビットに基づき選択できます。

**表 2-2. TX バッファ読み込みのマッピング**

| a | b | バッファ |
|---|---|---|
| 0 | 0 | バッファ – 0 |
| 0 | 1 | バッファ – 1 |
| 1 | 0 | バッファ – 2 |
| 1 | 1 | バッファ – 3 |

**TX FIFO 読み込み命令：**

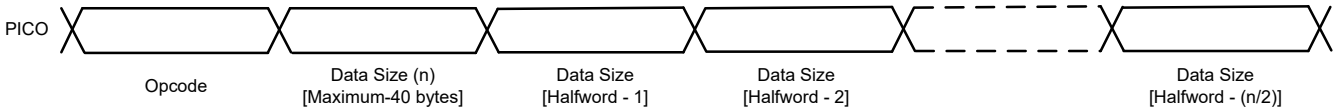



**図 2-6. TX FIFO 読み込み命令**

Opcode：0b0000000001010000

TX FIFO 読み込み命令は、送信 FIFO に送信する CAN フレームを読み込むために使用されます。CAN フレームを送信 FIFO にロードする前に、ユーザーは、送信 FIFO が一杯にならないように注意する必要があります。

**TX バッファ送信リクエスト命令**

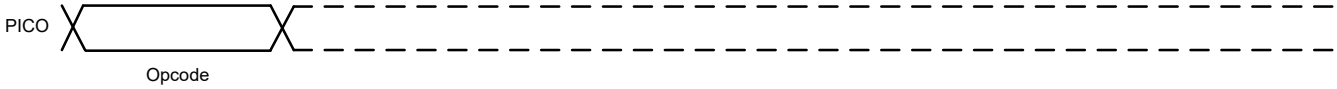



**図 2-7. TX バッファ送信リクエスト命令のフォーマット**

Opcode：0b00000000011000ab

TX バッファ送信リクエスト命令は、4 つの送信バッファの 1 つに対する送信リクエストを追加します。送信バッファの選択は、opcode でエンコードされます。4 つの送信バッファのうちの 1 つは、命令フィールドの a、b ビットに基づき選択できます。

**表 2-3. TX バッファ送信リクエストのマッピング**

| a | b | バッファ |
|---|---|---|
| 0 | 0 | バッファ – 0 |
| 0 | 1 | バッファ – 1 |
| 1 | 0 | バッファ – 2 |
| 1 | 1 | バッファ – 3 |

**TXFIFO 送信リクエスト命令**

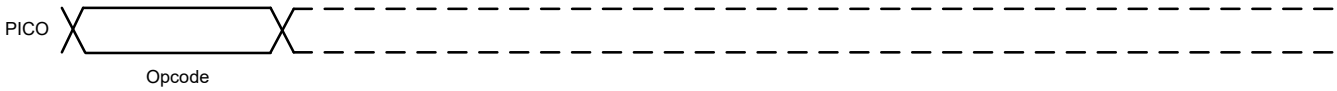



**図 2-8. TXFIFO 送信リクエスト命令のフォーマット**

Opcode：0b0000000001110000


Copyright © 2025 Texas Instruments Incorporated

TX FIFO 送信リクエスト命令は、送信 FIFO 配置インデックスに基づき、送信 FIFO 素子の 1 つに対する送信リクエストを追加します。ユーザーは、バッファ追加リクエストを送信する前に、送信 FIFO に CAN フレームがあることを確認する必要があります。

**レジスタ書き込み命令**

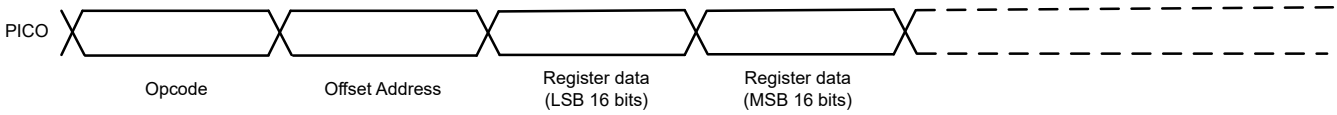



**図 2-9. レジスタ書き込み命令のフォーマット**

Opcode：0b0000000010000000

オフセットアドレス – レジスタデータを書き込むレジスタアドレスを指定します。

レジスタ書き込み命令は、SPI ペリフェラル経由で CAN レジスタを構成するために使用されます。データは、オフセットアドレスに基づき、該当するレジスタに書き込まれます。

**レジスタデータ読み取り命令：**

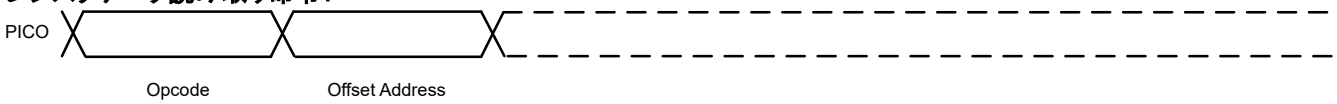



**図 2-10. レジスタデータ読み取り命令のフォーマット**

Opcode：0b0000000010000000

オフセットアドレス – レジスタデータを読み取るためのレジスタアドレスを指定します。

レジスタデータ読み取り命令は、特定の CAN レジスタからデータを読み取るために使用されます。レジスタデータ読み取り命令は、レジスタの読み取りを行いますが、送信は行いません。したがって、レジスタデータ読み取り命令の後は、レジスタデータフェッチ命令が続く必要があります。

**レジスタデータフェッチ命令：**

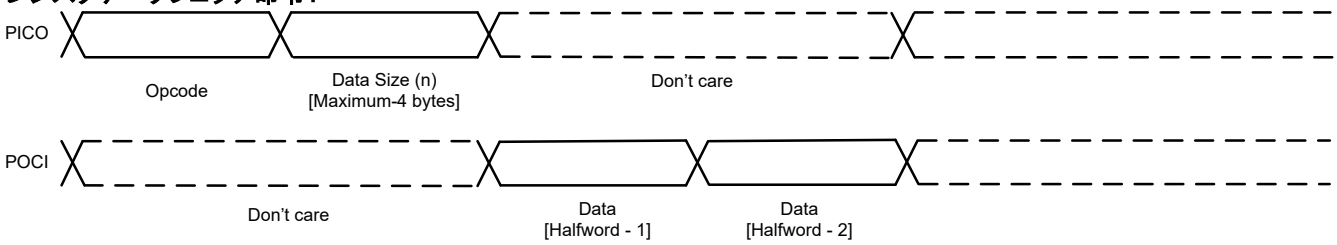



**図 2-11. レジスタデータフェッチ命令のフォーマット**

Opcode：0b0000000010100000

レジスタデータフェッチ命令は、レジスタデータ読み取り命令を使用して読み取ったレジスタ値の送信を行います。

**RX バッファデータ読み取り命令：**

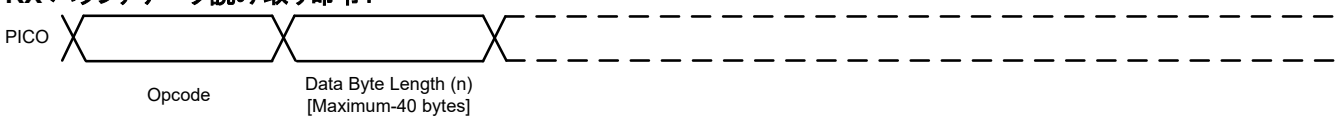



**図 2-12. RX バッファデータ読み取り命令のフォーマット**

Opcode：0b000000001011100xy


Copyright © 2025 Texas Instruments Incorporated

SPI-CAN ブリッジが CAN ネットワークから新しい CAN フレームを受信すると、図 2-12 に示すように、SPI-CAN ブリッジは GPIO ピンをトグルし、フレームが SPI コントローラで受信されたことを示します。

SPI-CAN ブリッジで CAN モジュール RX バッファから受信した CAN フレームを取得するには、次のシーケンスに従う必要があります。

1. まず、RX ステータスデータ読み取り命令と、RX ステータスデータフェッチ命令を使用し、受信した CAN フレームがどの RX バッファに含められるかを特定します。
2. 次に、RX バッファ読み取り命令を実行し、バッファの内容にアクセスします。この命令はバッファの場所を読み取るだけで、データの送信は行わないことに注意してください。
3. 最後に、RX バッファデータフェッチ命令を実行し、バッファからデータを取得します。

データの送信を完了するには、各 RX バッファ読み取り命令を対応する RX バッファデータフェッチ命令と組み合わせる必要があります。

RX バッファの選択は、opcode でエンコードされます。4 つの RX バッファのうちの 1 つは、命令の x、y ビットに基づいて選択できます。

**表 2-4. RX バッファ読み取りのマッピング**

| X | y | バッファ |
|---|---|---|
| 0 | 0 | バッファ – 0 |
| 0 | 1 | バッファ – 1 |
| 1 | 0 | バッファ – 2 |
| 1 | 1 | バッファ – 3 |

**RX バッファデータフェッチ命令**

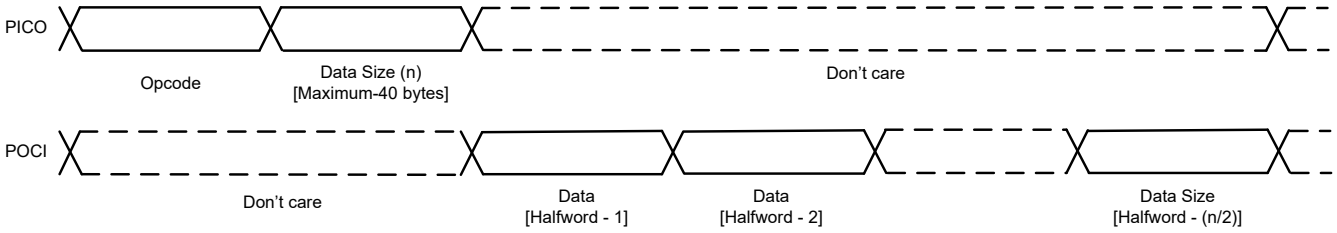



**図 2-13. RX バッファデータフェッチ命令のフォーマット**

Opcode：0b0000000011000000

RX バッファデータフェッチ命令は、RX バッファ読み取り命令を使用して読み取られた RX バッファデータを送信します。

**RX FIFO データ読み取り命令：**

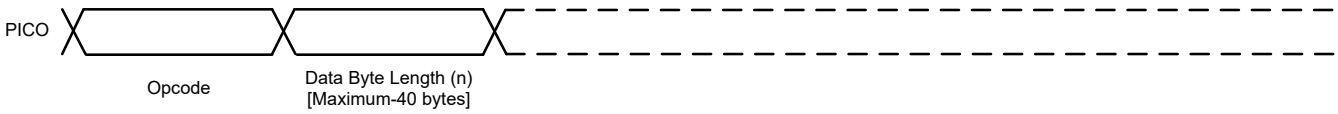



**図 2-14. RXFIFO データ読み取り命令のフォーマット**

Opcode：0b0000000011010000

SPI-CAN ブリッジが CAN ネットワークから新しい CAN フレームを受信すると、図 2-14 に示すように、SPI-CAN ブリッジは GPIO ピンをトグルし、フレームが SPI コントローラで受信されたことを示します。

SPI-CAN ブリッジ内で CAN モジュールの RX FIFO から受信した CAN フレームを取得するには、次のシーケンスに従う必要があります。


Copyright © 2025 Texas Instruments Incorporated

1. 最初に、RX ステータスデータ読み取り命令と、RX ステータスデータフェッチ命令を使用して、RXFIFO の状態に関する情報を取得します。
2. 次に、RX FIFO データ読み取り命令を実行し、FIFO 素子の内容にアクセスします。RX FIFO 取得インデックスに基づき、RX FIFO 素子が選択され、データが読み取られ、アレイに格納されます。この命令は FIFO 素子の位置を読み取るだけで、データの送信は行わないことに注意してください。
3. 最後に、RX FIFO データフェッチ命令を実行し、FIFO からデータを実際に取得します。

**RX FIFO データフェッチ命令:**

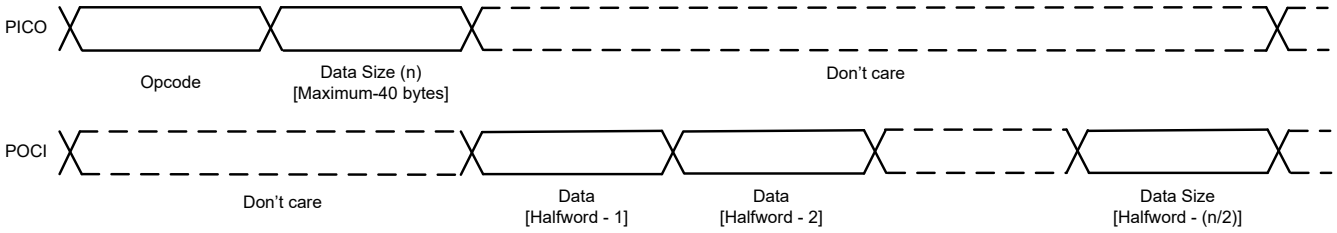



**図 2-15. RXFIFO データフェッチ命令のフォーマット**

Opcode:0b0000000011100000

RX FIFO データフェッチ命令は、RX FIFO 読み取り命令を使用して読み出された RX FIFO データを送信します。

**受信ステータスデータ読み取り命令**

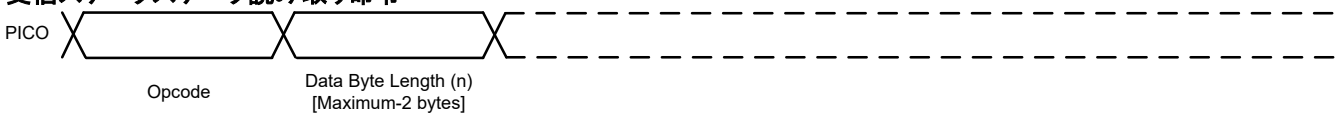



**図 2-16. 受信ステータスデータ読み取り命令のフォーマット**

Opcode:0b0000000011110000

受信ステータスデータ読み取り命令は、SPI から CAN デバイスへの CAN モジュールのステータスビットを取得するために使用されます。この命令に対する応答として受信されたデータビットの意味を図 2-17 に示します。

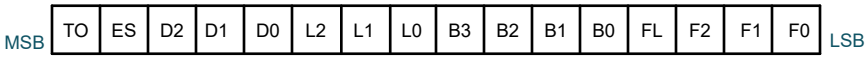



**図 2-17. ステータスビット表示**

- **B0 – B4**:RX は、NDAT レジスタからの新しいデータステータスをバッファします。各ビットは、各 RX バッファのステータスを示します。1:該当する Rx バッファには CAN フレームがあります。0:該当する Rx バッファは空です。
- **FL**:CAN RXFIFO のフルフラグステータスを示します。1:RXFIFO がいっぱいです。0:RXFIFO はいっぱいになっていません。
- **F2、F1、F0**:CAN RXFIFO の充填レベルを示します。1:該当する Rx FIFO には CAN フレームがあります。0:該当する Rx FIFO は空です。
- **L2、L1、L0**:CAN プロトコルの LEC (最終エラーコード) ビットを示します。

**表 2-5. 最終エラーコード (LEC) ビット診断のマッピング**

| LEC ビット [L2、L1、L0] | エラーコード |
|---|---|
| 000 | エラーなし |
| 001 | スタッフエラー |
| 010 | フォームエラー |
| 011 | Ack エラー |
| 100 | Bit1 エラー |
| 101 | Bit0 エラー |


Copyright © 2025 Texas Instruments Incorporated

**表 2-5. 最終エラーコード (LEC) ビット診断のマッピング (続き)**

| LEC ビット [L2、L1、L0] | エラーコード |
|---|---|
| 110 | CRC 誤差 |
| 111 | 変更なし |

- **D2、D1、D0**:CAN プロトコルの DLEC (データフェーズ最終エラーコード) ビットを示します。エラーコードは LEC と同じです。
- **ES**:前の命令のエラーステータスを示します。1:前の命令の受信中にエラーが発生しました。0:前の命令の受信中にはエラーはありませんでした。
- **TO**:前の命令のタイムアウトステータス。1:前の命令でタイムアウトが発生しました。0:前の命令でタイムアウトは発生しませんでした

**受信ステータスデータフェッチ命令**

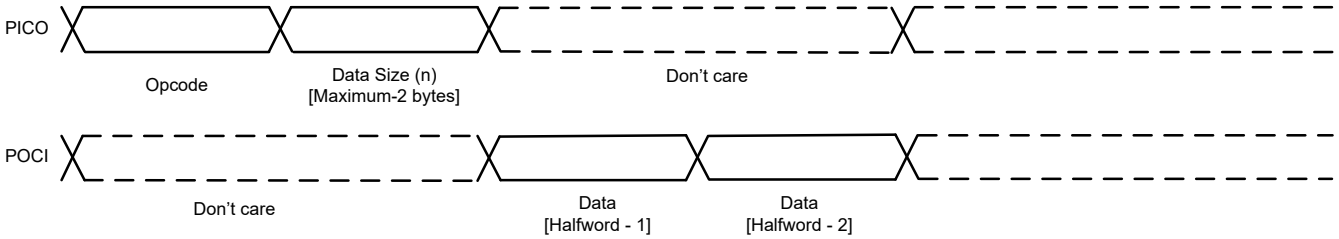



**図 2-18. 受信ステータスデータフェッチ命令のフォーマット**

Opcode:0b0000000000001000

RX ステータスデータフェッチ命令は、RX ステータスデータ読み取り命令によってすでに読み取られ、変数に保存されている RX ステータスデータを受信するために使用されます。

## 2.2 タイムアウト機能

SPI-CAN ブリッジは、以下の動作でタイムアウト保護を実装しています。

- メッセージタイムアウト:
  - 5 秒以内に完全な命令が受信されない場合。
  - ブリッジはタイムアウトをトリガします。
  - 不完全な命令は破棄されます。
- バッファ負荷タイムアウト:
  - TX バッファまたは FIFO ロード中にタイムアウトが発生した場合。
  - 部分的なデータは、メッセージ RAM に書き込むことができます。
  - 同じバッファへの後続の送信命令は、最初から開始されます。
  - 以前の不完全なデータは実質的に破棄されます。

外部 SPI コントローラは、*受信ステータス読み取り命令*を使用してタイムアウトイベントをチェックできます。これにより、コントローラは適切な回復措置を実行できます。

## 2.3 エラー表示

以下の条件が満たされるとエラーとなります。

- Bridge デバイスが受信した命令が無効な場合 (受信した命令コードは命令リストに含まれていない場合)。
- ブリッジによって受信されたオフセットアドレスが有効でない場合 (受信したオフセットアドレスが CAN レジスタアドレスのいずれにも関連していない場合)。

外部 SPI コントローラは、*受信ステータス読み取り命令*を使用して、前の命令のエラー表示を確認できます。

## 2.4 ビジーステータスの表示

SPI-CAN ブリッジは、GPIO ピンを使用してビジーステータスを示します。


English Document: SLAAEQ4
Copyright © 2025 Texas Instruments Incorporated

1. BUSY ピンは、次の場合にアサートされます。
   - opcode が SPI コントローラから受信された。
   - CAN ネットワークアクティビティが進行中。
2. BUSY ピンは、次の場合にクリアされます。
   - 現在の命令の処理が完了した。
   - CAN 転送が終了した。

重要:BUSY ピンがアサートされている間、外部 SPI コントローラは新しいトランザクションを始められません。これにより、競合を防止し、信頼性の高いデータ転送を維持できます。

### 2.5 メッセージ RAM の構成

電源有効化と CAN 構成命令を受信した場合、事前定義された CAN メッセージ RAM 構成がセットアップされます。図 2-19 に、デフォルトのメッセージ RAM の構成を示します。

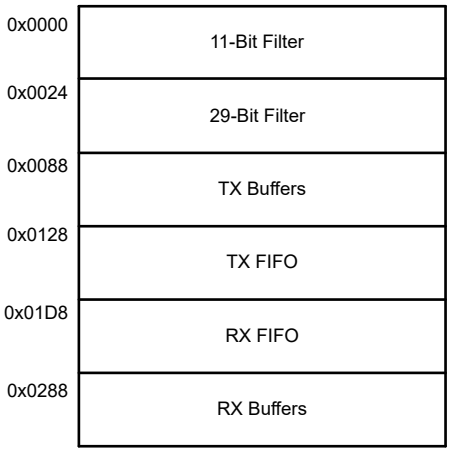



**図 2-19. メッセージ RAM の構成**

セクション 2.1.1 に示すように、メッセージ ID、送信 FIFO、送信バッファは、外部 SPI マスタによって構成できます。


Copyright © 2025 Texas Instruments Incorporated

## 2.6 テスト環境

図 2-20 にテスト環境の詳細を示します。SPI-CAN ブリッジは MSPM0G3507 ローンチパッドに実装され、別の MSPM0G3507 ローンチパッドで外部 SPI コントローラで実現されています。SPI-CAN ブリッジは、CAN-PHY 経由で外部ノードに接続されています。

### ピンの接続性

**表 2-6. ピンの接続性**

| デバイス | ピン機能 | ピン番号 |
|---|---|---|
| MSPM0G3507 – SPI コントローラ | SCLK | PA17 |
| | PICO | PA18 |
| | POCI | PA16 |
| | CS | PB1 |
| | ビジーステータス | PB7 |
| | CAN 新規メッセージ受信通知 | PB8 |
| MSPM0G3507 - ブリッジ | SCLK | PA17 |
| | PICO | PA18 |
| | POCI | PA16 |
| | CS | PB1 |
| | ビジーステータス | PB7 |
| | CAN 新規メッセージ受信通知 | PB8 |
| | CAN_TX | PA12 |
| | CAN_RX | PA13 |
| MSPM0G3507 – CAN ノード | CAN_TX | PA12 |
| | CAN_RX | PA13 |

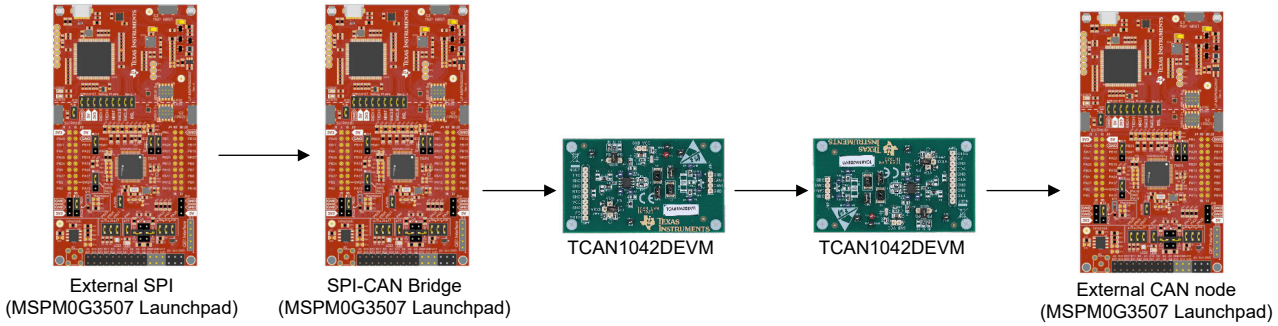



**図 2-20. SPI-CAN ブリッジのテスト設定**

アプリケーションコードは、MSPM0-SDK ソフトウェア開発キット (SDK) | TI.com からダウンロードできます。

# 3 参考資料

- MSPM0G3507 のデータシート、製品情報、およびサポート | TI.com
- LP-MSPM0G3507 評価ボード | TI.com
- TCAN1042DEVM 評価ボード | TI.com
- MSPM0-SDK ソフトウェア開発キット (SDK) | TI.com


Copyright © 2025 Texas Instruments Incorporated

# 重要なお知らせと免責事項

テキサス・インスツルメンツは、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むいかなる保証も、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、テキサス・インスツルメンツ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 テキサス・インスツルメンツ製品の選定、(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 テキサス・インスツルメンツ製品を使用するアプリケーションの開発の目的でのみ、テキサス・インスツルメンツ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。テキサス・インスツルメンツ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、テキサス・インスツルメンツおよびその代理人を完全に補償するものとし、テキサス・インスツルメンツは一切の責任を拒否します。

テキサス・インスツルメンツの製品は、テキサス・インスツルメンツの販売条件、または ti.com やかかる テキサス・インスツルメンツ製品の関連資料などのいずれかを通じて提供する適用可能な条項の下で提供されています。テキサス・インスツルメンツがこれらのリソースを提供することは、適用されるテキサス・インスツルメンツ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。

お客様がいかなる追加条項または代替条項を提案した場合でも、テキサス・インスツルメンツはそれらに異議を唱え、拒否します。

郵送先住所:Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2025, Texas Instruments Incorporated

# 重要なお知らせと免責事項

テキサス・インスツルメンツは、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むいかなる保証も、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、 テキサス・インスツルメンツ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 テキサス・インスツルメンツ製品の選定、(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 テキサス・インスツルメンツ製品を使用するアプリケーションの開発の目的でのみ、 テキサス・インスツルメンツ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。 テキサス・インスツルメンツ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、 テキサス・インスツルメンツおよびその代理人を完全に補償するものとし、 テキサス・インスツルメンツは一切の責任を拒否します。

テキサス・インスツルメンツの製品は、 テキサス・インスツルメンツの販売条件、または ti.com やかかる テキサス・インスツルメンツ製品の関連資料などのいずれかを通じて提供する適用可能な条項の下で提供されています。 テキサス・インスツルメンツがこれらのリソースを提供することは、適用される テキサス・インスツルメンツ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。

お客様がいかなる追加条項または代替条項を提案した場合でも、 テキサス・インスツルメンツはそれらに異議を唱え、拒否します。

郵送先住所：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2025, Texas Instruments Incorporated