



AN-1453 アプリケーション・ノート

PMBus を介した **ADM1266** のファームウェアと設定のプログラミング

著者 : Hossain Opal and Navdeep Dhanjal

はじめに

このアプリケーション・ノートでは、大規模なプログラミング環境またはシステム・アプリケーションで **ADM1266** のファームウェアと設定をプログラムするために必要なステップを詳しく説明します。ファームウェアと設定のアップデートは、条件に基づいて個別に行うことができます。

アナログ・デバイセズ社は、提供する情報が正確で信頼できるものであることを期していますが、その情報の利用に関して、あるいは利用によって生じる第三者の特許やその他の権利の侵害に関して一切の責任を負いません。また、アナログ・デバイセズ社の特許または特許の権利の使用を明示的または暗示的に許諾するものでもありません。仕様は、予告なく変更される場合があります。本紙記載の商標および登録商標は、それぞれの所有者の財産です。※日本語版資料は REVISION が古い場合があります。最新の内容については、英語版をご参照ください。

Rev. 0

©2018 Analog Devices, Inc. All rights reserved.

アナログ・デバイセズ株式会社

本社	〒105-6891	東京都港区海岸 1-16-1	ニューピア竹芝サウスタワービル 10F
			電話 03 (5402) 8200
大阪営業所	〒532-0003	大阪府大阪市淀川区宮原 3-5-36	新大阪トラストタワー 10F
			電話 06 (6350) 6868
名古屋営業所	〒451-6038	愛知県名古屋市中区牛島町 6-1	名古屋ルーセントタワー 40F
			電話 052 (569) 6300

目次

はじめに.....	1	設定のプログラミング	7
改訂履歴.....	2	シーケンスを停止してから設定データをプログラミング	7
ハードウェア構成.....	3	メモリ・ポインタをメイン・メモリに設定.....	8
メモリ・パーティション	4	データ書込みの時間間隔	8
.Hex ファイルの読出し.....	4	アップデートされたメイン・メモリ設定データの確認	9
ファームウェアのプログラミング.....	5	アップデート後のシーケンスの再開.....	9
ステップ 1 : シーケンスを停止してからファームウェアをア ップグレード.....	5	メモリ・リフレッシュのトリガ	9
ステップ 2 : ADM1266 のロック解除	5	ファームウェアと設定データの確認.....	9
ステップ 3 : ADM1266 をブートローダ・モードにする	5	ADM1266 のロック	10
ステップ 4 : ファームウェア・データの書込みの時間間隔.....	6	パケット・エラー・チェック (PEC) コントロール (オ プション)	10
ステップ 5 : ファームウェアのプログラミング後におけるリ セット・コマンドの送信または電源の入れ直し.....	6		

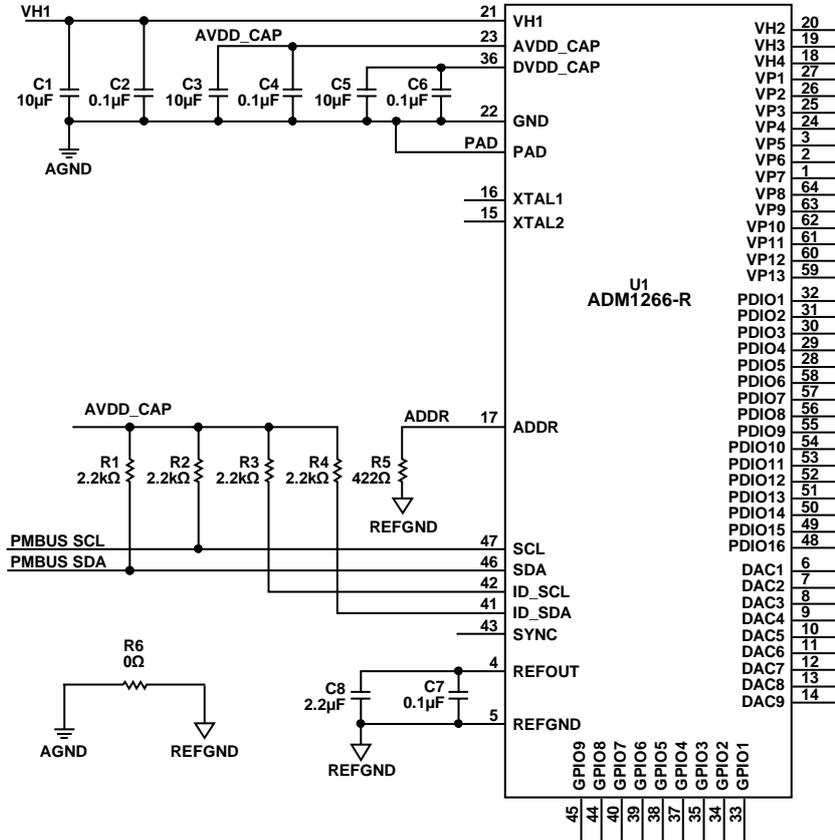
改訂履歴

7/2018—Revision 0: Initial Version

ハードウェア構成

図 1 は、ADM1266 について大規模なプログラミング環境に限定した場合に推奨されるピン設定を示しています。ハードウェア構成に使用するすべての部品は、表面実装部品にすることを推奨します。

ADM1266 の PMBus アドレスは、ADDR ピンと GND の間に外付け抵抗 (R5) を接続して設定します。表 1 に、推奨する抵抗値とそれに対応する PMBus アドレスを示します。



- NOTES
 1. FLOATING PINS ARE NOT CONNECTED (NC).
 2. VH1 MUST BE CONNECTED TO 12V.

図 1. 大規模なプログラミング環境における推奨のピン設定

表 1. PMBus アドレスの設定

PMBus Address	1% Resistor (kΩ), E96 Series
0x40	0.422
0x41	1.5
0x42	2.67
0x43	4.12
0x44	5.36
0x45	7.15
0x46	8.87
0x47	10.7
0x48	12.7
0x49	14.7
0x4A	16.9
0x4B	19.1
0x4C	21.5
0x4D	24.3
0x4E	27.4
0x4F	31.6

メモリ・パーティション

図 2 に、このアプリケーション・ノートに適用可能なメモリ・パーティションの概要を示します。

ADM1266 には、ミニ・ブートローダやブートローダだけが予めプログラムされています。ミニ・ブートローダやブートローダは、ユーザがアップグレードすることはできません。

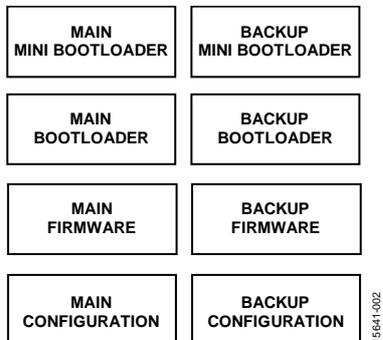


図 2. ADM1266 のメモリ・パーティション

ADM1266 にブートローダだけが予めプログラムされている場合、最初にファームウェアをプログラムし、続いて設定データをプログラムする必要があります。

ADM1266 用のファームウェアと設定は、ADI Power Studio®グラフィカル・ユーザ・インターフェース (GUI) ソフトウェアによって生成されます。このソフトウェアは、[EVAL-ADM1266](#) 評価用ボードの製品ページからダウンロードできます。

すべてのデバイスで同じファームウェアが使用され、またデバイスごとに異なる設定が使用されます。

.HEX ファイルの読出し

ファームウェアと設定用に生成される .hex ファイルは、Intel®16 進数フォーマットに従います。

.hex ファイルの各行には、16 進数が含まれています。この 16 進数は、データを表す複数の 2 進数と、データを書き込む必要のあるレジスタ・アドレスがエンコードされたものです。

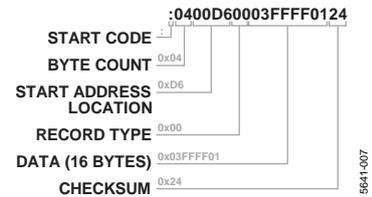


図 3. ADM1266 .Hex ファイルにおける代表的な記録内容

.hex ファイルの最後は :00000001FF で表されます。

PMBus の書き込みタイプはバイト・カウントに基づいており、例えば、1 バイトはバイト書き込み、2 バイトはワード書き込み、3 バイト以上はブロック書き込みになります。

ファームウェアのプログラミング

ADM1266 は、ファームウェアがない場合にはブートローダ・モードで起動し、ブートローダ・モードを維持します。図 4 に示すように、ファームウェアをロードするにはステップ 4 とステップ 5 だけがが必要です。

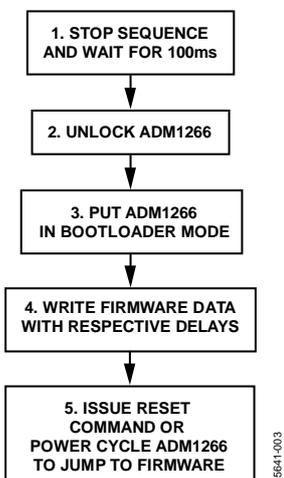


図 4. ファームウェアをプログラミングするためのフローチャート

ステップ 1: シーケンスを停止してからファームウェアをアップグレード

PMBus を介してファームウェアをアップグレードする前に、現在のシーケンスを停止する必要があります。シーケンスの停止は、GO_COMMAND コマンド (レジスタ 0xD8) に 0x0003 を書き込むことによって行います。図 5 に示すように、グループ・コマンドを發して、同じ PMBus にある複数の ADM1266 デバイスを同時に停止することができます。

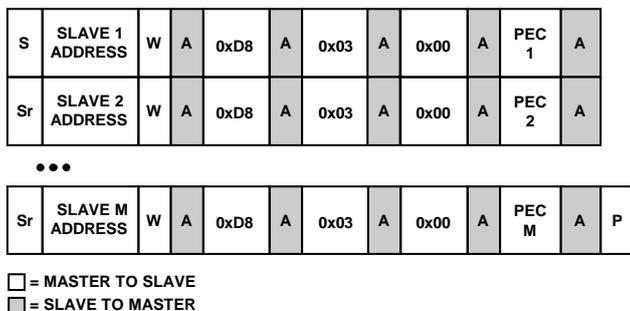


図 5. シーケンスを停止するためのグループ・コマンド

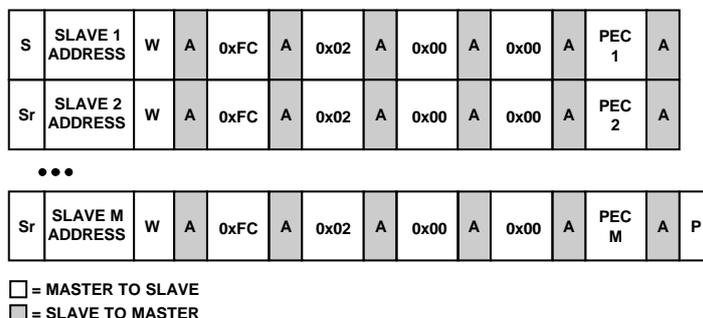


図 7. ADM1266 をブートローダ・モードにするグループ・コマンド

グループ・コマンドを送信できない場合は、ADM1266 デバイスごとに停止することができます。

停止コマンドを送信後、100ms 待機してから次の PMBus コマンドを送信する必要があります。

ステップ 2: ADM1266 のロック解除

ADM1266 は、ファームウェアがアップデート可能になる前にロックを解除する必要があります。

ADM1266 のロック解除は、FW_PASSWORD コマンド (レジスタ 0xFD) に正しいパスワードを 2 回連続して書き込むことによって実行できます。ADM1266 のロックを解除するブロック書き込みコマンドを図 6 に示します。デフォルトのパスワードは最初の 16 バイトが 0xFF で、17 番目のバイト (パスワード・コマンド) が 0x02 に設定されています。

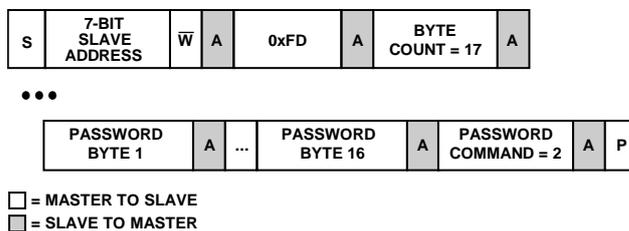


図 6. ADM1266 のロックを解除するブロック書き込みコマンド

ロック解除のステータスは、STATUS_MFR_SPECIFIC コマンド (レジスタ 0x80) の PART_LOCKED ビット (ビット 2) によって確認できます。このビットは、ADM1266 がロック解除された場合、0 に設定されています。詳細については表 5 を参照してください。

ステップ 3: ADM1266 をブートローダ・モードにする

ADM1266 は、ファームウェアがプログラム可能になる前にブートローダ・モードにする必要があります。図 7 に示すように、ADM1266 はグループ・コマンドとして、UPDATE_FW コマンド (レジスタ 0xFC) に 0x02、0x00、および 0x00 を書き込むことによって、ブートローダ・モードにできます。

このコマンドを送信できない場合は、ADM1266 デバイスごとにブートローダ・モードにできます。

ステップ 4：ファームウェア・データの書込みの時間間隔

ファームウェアが PMBus を介して ADM1266 にプログラムされる時は、EEPROM を消去する必要はありません。ブートローダがファームウェア・データの最初のストリームを受信しているときに、ブートローダはファームウェア用に予約されている EEPROM のページを自動的に消去します。表 2 に、それぞれのデータ書込みの時間間隔を示します。

ステップ 5：ファームウェアのプログラミング後におけるリセット・コマンドの送信または電源の入れ直し

ファームウェア・データが ADM1266 に正常に書き込まれた後、設定データをプログラミングする前にリセット・コマンドの送信または電源の入れ直しを行う必要があります。

ADM1266 のリセットは GO_COMMAND コマンド（レジスタ 0xD8）に 0x0004 を書き込むことによって行うことができます。

表 2. UPDATE_FW コマンドへの書込みの時間間隔

Command	Wait Time Required Before Writing Next Block of Data
First Write to 0xFC (UPDATE_FW)	2 sec
Subsequent Writes to 0xFC	40 ms

設定のプログラミング

設定をプログラミングするためのフローチャートについては、図 8 を参照してください。

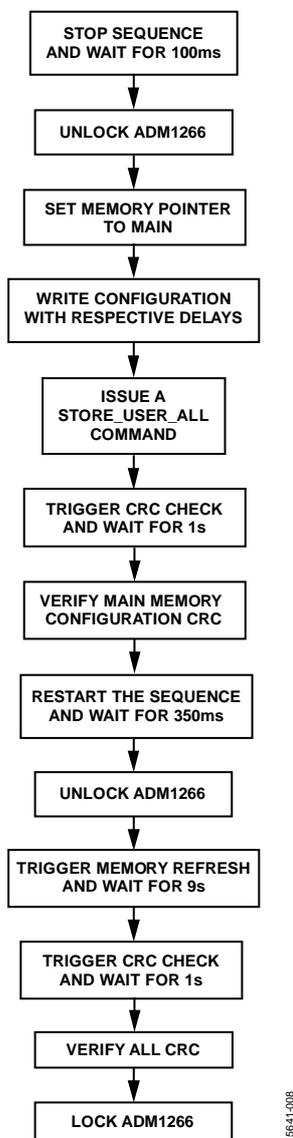


図 8. 設定をプログラミングするためのフローチャート

ステップ 2 : ADM1266 のロック解除のセクションと、.Hex ファイルの読出しのセクションに詳述されている手順に従ってください。

シーケンスを停止してから設定データをプログラミング

設定を PMBus を介してプログラムする前に、現在のシーケンスを停止する必要があります。シーケンスを停止するには、シームレス・アップデート・リセット・ストップとソフト・リセット・ストップという 2 つのオプションがあります。設定をアップデートするときは、シームレス・アップデート・リセット・ストップを使用することを推奨します。このオプションを使用すると、シーケンスが再開したとき、シーケンスが新しい設定のパワーグッド状態に直接ジャンプします。

図 9 に示すように、シームレス・アップデート・リセット・ストップでは、同じ PMBus 上の複数の ADM1266 デバイスを同時に停止するグループ・コマンドとして、GO_COMMAND コマンド (レジスタ 0xD8) に 0x0011 を書き込むことによってシーケンスが停止します。

S	SLAVE 1 ADDRESS	W	A	0xD8	A	0x11	A	0x00	A	PEC 1	A
Sr	SLAVE 2 ADDRESS	W	A	0xD8	A	0x11	A	0x00	A	PEC 2	A
...											
Sr	SLAVE M ADDRESS	W	A	0xD8	A	0x11	A	0x00	A	PEC M	A P

= MASTER TO SLAVE
 = SLAVE TO MASTER

図 9. シームレス・アップデートでのシーケンスを停止するためのグループ・コマンド

シームレス・アップデート・リセット・ストップには、10 分間のデフォルト・タイムアウトがあります。ユーザがシーケンスを再開しない場合、シーケンスはタイムアウトが満了した後に自動的に再開します。タイムアウト値は、シーケンスを停止する前に、HITLESS_TIMEOUT コマンド (レジスタ 0xF6) を使用することによってアップデートできます。図 10 に、リセット時間を 60 秒に設定する例を示します。

大規模なプログラミング環境や、設定が予めプログラムされていないときは、ソフト・リセット・ストップの使用のみを推奨します。

図 9 に示すように、ソフト・リセット・ストップでは、同じ PMBus 上の複数の ADM1266 デバイスを同時に停止するグループ・コマンドとして、GO_COMMAND コマンド (レジスタ 0xD8) に 0x0003 を書き込むことによってシーケンスが停止します。

グループ・コマンドを送信できない場合は、ADM1266 デバイスごとに停止することができます。

停止コマンドを送信後、100ms 待機してから次の PMBus コマンドを送信する必要があります。

Sr	SLAVE ADDRESS	W	A	0xF6	A	0x02	A	0x3C	A	0x00	A	PEC M	A	P
----	---------------	---	---	------	---	------	---	------	---	------	---	-------	---	---

= MASTER TO SLAVE
 = SLAVE TO MASTER

図 10. ヒットレス・タイムアウト・コマンド

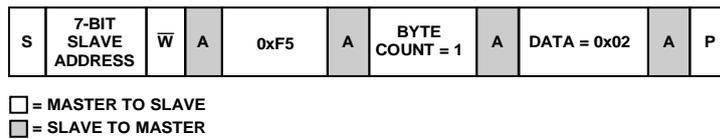


図 13. メモリ・リフレッシュのトリガ

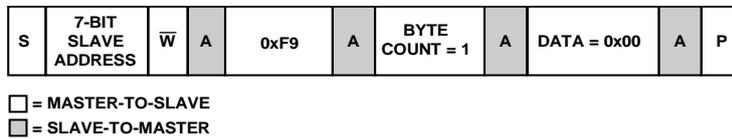


図 14. CRC 再計算のトリガ

アップデートされたメイン・メモリ設定データの確認

設定をアップデートしたときにのみ確認が必要です。ADM1266 に設定が予めプログラムされていない場合は、アップデートされたメイン・メモリ設定データの確認は任意です。

図 14 に示すように、設定データを書き込んだ後、すべてのメモリ・セクションの巡回冗長検査 (CRC) の計算は、MEMORY_RECALCULATE_CRC コマンド (レジスタ 0xF9) に書き込みを行うことによってトリガできます。

1 秒間待機してからメモリ CRC のステータスを読み出します。このステータスは STATUS_MFR_SPECIFIC_2 コマンド (レジスタ 0xED) のワード読出しによって確認できます。このレジスタのビット・マッピングを表 4 に列挙します。MAIN_PROJECT_CRC_FAULT ビットと MAIN_ABCONFIG_CRC_FAULT ビットが 0 に設定されていることを確認します。これらのビットが 0 に設定されている場合、メイン・メモリ設定は正常に書き込まれたこととなります。

アップデート後のシーケンスの再開

新しい設定データが ADM1266 に書き込まれた場合、複数の ADM1266 デバイスのシーケンスを同時に再開するグループ・コマンドとして、GO_COMMAND コマンド (レジスタ 0xD8) に 0x0000 を書き込むことによって、新しいシーケンスを実行できます。グループ・コマンドのプロトコルを図 5 に示します。

グループ・コマンドを送信できない場合は、ADM1266 デバイスごとに再開することができます。

シームレス・アップデート・リセット・ストップを使用してシーケンスが停止した場合、再開時にはシーケンスは設定で定義されたパワーグッド状態に直接ジャンプします。パワーグッド状態が定義されていない場合は、シーケンスは最初のユーザ状態から開始します。

ソフト・リセット・ストップを使用してシーケンスが停止した場合は、再開時にボードの電源を入れ直すことで、シーケンスは最初のユーザ状態から開始します。

シーケンスが再開後、350ms 待機してから次の PMBus コマンドを送信します。

メモリ・リフレッシュのトリガ

ADM1266 には、ミニ・ブートローダ、ブートローダ、ファームウェア、および設定用にバックアップ・メモリ・セクションがあります。ファームウェアと設定が正常にプログラムされた後、メモリ・リフレッシュをトリガしてください。メモリ・リフレッシュにより、メイン・メモリの内容がバックアップ・メモリにコピーされ、バックアップ・メモリの内容がメイン・メモリにコピーされます。

必ず、ADM1266 をロック解除してからメモリ・リフレッシュをトリガするようにしてください。ステップ 2: ADM1266 のロック解除のセクションを参照してください。

図 13 に示すように、REFRESH_FLASH コマンド (レジスタ 0xF5) にブロック書き込みを行うことによって、すべてのメモリ・セクションに対するメモリ・リフレッシュをトリガできます。

メモリ・リフレッシュは、終了まで約 9 秒かかります。STATUS_S_MFR_SPECIFIC コマンド (レジスタ 0x80) の RUNNING_REFRESH ビットを読み出すことにより、メモリ・リフレッシュのステータスを確認してください。RUNNING_REFRESH ビットが 1 に設定されている場合は、メモリ・リフレッシュが依然として動作しています。このビットが 0 に設定されるまで待つから次に進みます。

ファームウェアと設定データの確認

ファームウェアと設定データを書き込み、メモリ・リフレッシュをトリガした後、図 14 に示すように、すべてのメモリ・セクションの CRC 計算が、MEMORY_RECALCULATE_CRC コマンド (レジスタ 0xF9) に書き込みを行うことによってトリガできます。

1 秒間待ってからメモリ CRC のステータスを読み出します。メモリ CRC のステータスを確認するには、STATUS_MFR_SPECIFIC_2 コマンド (レジスタ 0xED) のワード読出しを行います。このレジスタのビット・マッピングを表 4 に列挙します。

表 4. STATUS_MFR_SPECIFIC_2 コマンドのビット・マッピング

Bit	Bit Name
15	BKUP_PASSWORD_CRC_FAULT
14	BKUP_FIRMWARE_CRC_FAULT
13	BKUP_PROJECT_CRC_FAULT
12	BKUP_ABCONFIG_CRC_FAULT
11	MAIN_PASSWORD_CRC_FAULT
10	MAIN_FIRMWARE_CRC_FAULT
9	MAIN_PROJECT_CRC_FAULT
8	MAIN_ABCONFIG_CRC_FAULT
7	BKUP_IAP_CRC_FAULT
6	BKUP_MINI_IAP_CRC_FAULT
5	MAIN_IAP_CRC_FAULT
4	MAIN_MINI_IAP_CRC_FAULT

ALL_CRC_FAULT は単一のビットで、表 4 に示しているすべての CRC ビットの OR です。このビットは、STATUS_MFR_SPECIFIC コマンド（レジスタ 0x80）から読み出すことができます。

STATUS_MFR_SPECIFIC コマンド（レジスタ 0x80）をリードバックすることによって、設定互換性（ビット 1）とシリコン互換性（ビット 0）のステータスを確認します。

表 5. STATUS_MFR_SPECIFIC コマンドのビット・マッピング

Bits	Bit Name
5	ALL_CRC_FAULT
4	TESTMODE_ENABLED
3	RUNNING_REFRESH
2	PART_LOCKED
1	PART_DATA_COMPATIBLE_FAULT
0	SILICON_COMPATIBLE_FAULT

CRC と互換性が合格の場合、それらが合格したビットには 0 が設定されます。CRC と互換性が不合格の場合、それらが不合格したビットには 1 が設定されます。

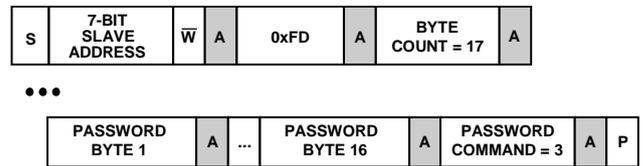
ファームウェアのバージョン番号は、IC_DEVICE_REV コマンド（レジスタ 0xAE）からリードバックできます。ブロック読み出しを使用して、ファームウェアが正しいリビジョンにアップデートされていることを確認します。コマンドのバイト・マッピングを表 6 に示します。

表 6. IC_DEVICE_REV コマンドのマッピング

Byte Name	Byte	Example	Notes
Data Length	Byte 0	0x08	Fixed to 0x08
Firmware Revision	Byte 1	0x01	Indicates Version 1.8.7
	Byte 2	0x08	
	Byte 3	0x07	
Bootloader Revision	Byte 4	0x00	Indicates Version 0.0.7
	Byte 5	0x00	
	Byte 6	0x07	
Chip Revision	Byte 7	0x42	Indicates Version B0
	Byte 8	0x30	

ADM1266 のロック

ADM1266 は、電源を入れ直すと自動的にロックされます。また、ADM1266 は、任意の 16 バイトのデータと、0x03 に設定された 17 番目のバイト（パスワード・コマンド）を FW_PASSWORD コマンド（レジスタ 0xFD）に 1 回書き込むことによってロックできます。ADM1266 をロックするブロック書き込みコマンドを図 15 に示します。



- = MASTER TO SLAVE
- = SLAVE TO MASTER

図 15. ADM1266 をロックするブロック書き込みコマンド

ロックのステータスは、STATUS_MFR_SPECIFIC コマンド（レジスタ 0x80）の PART_LOCKED ビット（ビット 2）によって確認できます。このビットは、デバイスが正常にロックされた場合、1 に設定されています。

パケット・エラー・チェックング (PEC) コントロール (オプション)

PC 通信においてアクノレッジ・ビットとノー・アクノレッジ・ビットを認識できる場合、正常なデータ転送を確保するため、すべての書き込みで PEC バイトを使用することを推奨します。

PMBus コントローラは、PEC を実行して信頼性と通信の堅牢性を向上させます。PEC は、転送メッセージの最後に PEC バイトを付加することにより実行されます。PEC バイトは、開始ビットから停止ビットまでの間にあるすべてのアドレス、コマンド、およびデータ・バイトに CRC-8 アルゴリズムを使用して計算されます。この処理では、アクノレッジ、ノー・アクノレッジ、開始、再開、および停止の各ビットは除外されます。PEC バイトは、最後のデータ・バイトを供給したデバイスによってメッセージの最後に付加されます。PEC バイトを受信したデバイスは、内部パケット・エラー・コードを計算して、このコードと受信した PEC バイトを比較します。

PMBus のスレーブ・デバイスは、PEC をサポートしているマスタの PMBus デバイスと同様に、PEC をサポートしていないマスタ・デバイスとも通信できます。PEC バイトが利用可能な場合、PMBus は PEC バイトをチェックして、PEC バイトが正しい場合にはアクノレッジを返します。PEC バイトの比較に失敗した場合は、PMBus デバイスはこの PEC バイトにアクノレッジを返さず、また、マスタから送信されたコマンドも処理しません。

PMBus は内蔵ハードウェアを使用して、CRC-8 多項式、 $C(x) = x^8 + x^2 + x^1 + 1$ により PEC コードを計算します。PEC コードは、受信した順に 1 回に 1 バイトずつ計算されます。読み出しトランザクションでは、PMBus は最後のデータ・バイトの次に PEC バイトを付加します。書き込みトランザクションでは、PMBus は受信した PEC バイトを内部で計算した PEC コードと比較します。