



# CY9B560R/460R/360R/160R シリーズ 32 ビット・マイクロコントローラ

## FM4 ファミリ Flash Programming Specifications

Document Number: 002-04907 Rev. \*C

Cypress Semiconductor  
198 Champion Court  
San Jose, CA 95134-1709  
[www.cypress.com](http://www.cypress.com)

© Cypress Semiconductor Corporation, 2013-2019. 本書面は、Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社（以下、「Cypress」という。）に帰属する財産である。本書面（本書面に含まれ又は言及されているあらゆるソフトウェア又はファームウェア（以下、「本ソフトウェア」という。）を含む）は、アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき、Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し、また、本段落で特に記載されているものを除き、Cypress の特許権、著作権、商標権又はその他の知的財産権のライセンスを一切許諾していない。本ソフトウェアにライセンス契約書が伴っておらず、かつ、あなたが Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意をしていない場合、Cypress は、あなたに対して、

(1) 本ソフトウェアの著作権に基づき、(a) ソースコード形式で提供されている本ソフトウェアについて、Cypress ハードウェア製品と共に用いるためののみ、組織内部でのみ、本ソフトウェアの修正及び複製を行うこと、並びに (b) Cypress のハードウェア製品ユニットに用いるためののみ、（直接又は再販売者及び販売代理店を介して間接のいずれかで）エンドユーザーに対して、バイナリーコード形式で本ソフトウェアを外部に配布すること、並びに (2) 本ソフトウェア（Cypress により提供され、修正がなされていないもの）に抵触する Cypress の特許権のクレームに基づき、Cypress ハードウェア製品と共に用いるためののみ、本ソフトウェアの作成、利用、配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス（サブライセンスの権利を除く）を付与する。本ソフトウェアのその他の使用、複製、修正、変換又はコンパイルを禁止する。

**適用される法律により許される範囲内で、Cypress は、本書面又はいかなる本ソフトウェア若しくはこれに伴うハードウェアに関しても、明示又は黙示をとわず、いかなる保証（商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない）も行わない。**いかなるコンピューティングデバイスも絶対に安全ということはない。従って、Cypress のハードウェアまたはソフトウェア製品に講じられたセキュリティ対策にもかかわらず、Cypress は、Cypress 製品への権限のないアクセスまたは使用といったセキュリティ違反から生じる一切の責任を負わない。加えて、本書面に記載された製品には、エラッタと呼ばれる設計上の欠陥またはエラーが含まれている可能性があり、公表された仕様とは異なる動作をする場合がある。適用される法律により許される範囲内で、Cypress は、別途通知することなく、本書面を変更する権利を留保する。Cypress は、本書面に記載のある、いかなる製品若しくは回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報（あらゆるサンプルデザイン情報又はプログラムコードを含む）は、参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計、プログラム、かつテストすることは、本書面のユーザーの責任において行われるものとする。Cypress 製品は、兵器、兵器システム、原子力施設、生命維持装置若しくは生命維持システム、蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム、汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分としての使用、又は装置若しくはシステムの不具合が人身傷害、死亡若しくは物的損害を生じさせるようなその他の使用（以下「本目的外使用」という。）のためには設計、意図又は承認されていない。重要な構成部分とは、その不具合が装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できるような装置若しくはシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ、若しくは本目的外使用に関連するいかなる請求、損害又はその他の責任についても、Cypress はその全部又は一部をとわず一切の責任を負わず、かつ Cypress はそれら一切から本書により免除される。Cypress は Cypress 製品の本来目的外使用から生じ又は本来目的外使用に関連するあらゆる請求、費用、損害及びその他の責任（人身傷害又は死亡に基づく請求を含む）から免責補償される。

Cypress, Cypress のロゴ, Spansion, Spansion のロゴ及びこれらの組み合わせ, WICED, PSoC, CapSense, EZ-USB, F-RAM, 及び Traveo は、米国及びその他の国における Cypress の商標又は登録商標である。Cypress のより完全な商標のリストは、cypress.com を参照すること。その他の名称及びブランドは、それぞれの権利者の財産として権利主張がなされている可能性がある。

Arm and Cortex are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

# はじめに



## 本書の目的と対象読者

本書は、実際に本シリーズを使用して製品を開発される技術者を対象に、フラッシュメモリの機能や動作や、フラッシュメモリのシリアル書込みについて解説しています。

## 本書の全体構成

本書は、以下に示す4つの章から構成されます。

### 第1章 メインフラッシュメモリ

メインフラッシュメモリの概要、構成、動作およびレジスタについて説明します。

### 第2章 ワークフラッシュメモリ

ワークフラッシュメモリの概要、構成、動作およびレジスタについて説明します。

### 第3章 フラッシュセキュリティ

フラッシュセキュリティ機能によりフラッシュメモリの内容を保護できます。  
フラッシュセキュリティの概要、動作について説明します。

### 第4章 シリアル書込み接続

フラッシュメモリのシリアル書込みについて、Cypress 製シリアルプログラマを用いた場合の基本構成を説明します。

## サンプルプログラムおよび開発環境

FM4 ファミリの周辺機能を動作させるためのサンプルプログラムを無償で提供しております。また、本シリーズで使用する開発環境も掲載しています。当社マイコンの動作仕様や使用方法の確認などにお役立てください。

## マイコンサポート情報

<https://community.cypress.com/community/MCU>

\*: サンプルプログラムは、予告なしに変更することがあります。また、サンプルプログラムは標準的な動作や使い方を示したものですので、お客様のシステム上でご使用の際は十分評価された上でご使用ください。  
また、サンプルプログラムの使用に起因し生じた損害については、当社は一切その責任を負いません。

# 本書の使い方



## 機能の探し方

本書では次の方法で、使いたい機能の説明を探すことができます。

### ■ 目次から探す

本書の内容を記載順に示します。

### ■ レジスタから探す

本文中では各レジスタの配置アドレスを記載していません。各レジスタのアドレスを確認するときは『FM4 ファミリーペリフェラルマニュアル』の『Appendixes』の『A. レジスタマップ』を参照してください。

## 用語について

本書で使用している用語について示します。

用語	説明
ワード	32ビット単位でのアクセスを指します。
ハーフワード	16ビット単位でのアクセスを指します。
バイト	8ビット単位でのアクセスを指します。

## 表記について

■ 本書のレジスタ説明中のビット構成図では以下のように表記しています。

- bit : ビット番号
- Field : ビットフィールド名
- 属性 : 各ビットのリード、ライト属性
  - ・ R : リードオンリ
  - ・ W : ライトオンリ
  - ・ RW : リード・ライト可能
  - ・ - : 未定義
- 初期値: リセット直後のレジスタ初期値
  - ・ 0 : 初期値"0"
  - ・ 1 : 初期値"1"
  - ・ X : 初期値不定

■ 本書では、複数のビットを以下のように表記しています。

例: bit7 から bit0 の場合は bit7:0

■ 本書では、アドレスなどの数値を以下のように表記しています。

- 16進数: プレフィックス(接頭辞)として"0x"を付けて表記しています(例: 0xFFFF)。
- 2進数: プレフィックス(接頭辞)として"0b"を付けて表記しています(例: 0b1111)。
- 10進数: 数値だけで表記しています(例: 1000)。

# Contents



<b>CHAPTER 1: メインフラッシュメモリ</b> .....	<b>7</b>
<b>1. 概要</b> .....	<b>8</b>
<b>2. 構成</b> .....	<b>9</b>
<b>3. 動作説明</b> .....	<b>14</b>
3.1 メインフラッシュメモリのアクセスモード .....	15
3.2 自動アルゴリズム .....	16
3.2.1 コマンドシーケンス .....	17
3.2.2 コマンド動作説明 .....	18
3.2.3 自動アルゴリズムの実行状態 .....	19
3.3 メインフラッシュメモリの動作説明 .....	23
3.3.1 読出し/リセット動作 .....	23
3.3.2 書込み動作 .....	23
3.3.3 フラッシュ消去動作 .....	25
3.3.4 セクタ消去動作 .....	25
3.3.5 セクタ消去一時停止動作 .....	28
3.3.6 セクタ消去再開動作 .....	29
3.4 ECC 搭載品のメインフラッシュメモリへの書込みについて .....	30
3.5 メインフラッシュアクセラレータ .....	31
3.6 メインフラッシュメモリの使用上の注意 .....	33
<b>4. レジスタ</b> .....	<b>34</b>
4.1 FASZR (Flash Access Size Register) フラッシュ・アクセスサイズ・レジスタ .....	35
4.2 FRWTR (Flash Read Wait Register) フラッシュ・リードウェイト・レジスタ .....	36
4.3 FSTR (Flash Status Register) フラッシュ・ステータス・レジスタ .....	37
4.4 FSYNDN (Flash Sync Down Register) フラッシュ・シンクダウン・レジスタ .....	38
4.5 FBFCR (Flash Buffer Control Register) フラッシュ・バッファコントロール・レジスタ .....	39
4.6 FICR (Flash Interrupt Control Register) フラッシュ・割込み制御・レジスタ .....	40
4.7 FISR (Flash Interrupt Status Register) フラッシュ・割込みステータス・レジスタ .....	41
4.8 FICLR (Flash Interrupt Clear Register) フラッシュ・割込みクリア・レジスタ .....	42
4.9 CRTRMM (CR Trimming Data Mirror Register) CR トリミングデータ・ミラー・レジスタ .....	43
4.10 FERRAD (Flash ECC ERR Address Capture Register) フラッシュ・ECC ERR アドレス・キャプチャ・レジスタ .....	44

<b>CHAPTER 2: ワークフラッシュメモリ</b> .....	<b>45</b>
<b>1. 概要</b> .....	<b>46</b>
<b>2. 構成</b> .....	<b>47</b>
<b>3. 動作説明</b> .....	<b>48</b>
3.1 ワークフラッシュメモリのアクセスモード .....	49
3.2 自動アルゴリズム .....	50
3.2.1 コマンドシーケンス .....	51
3.2.2 コマンド動作説明 .....	52
3.2.3 自動アルゴリズムの実行状態 .....	53
3.3 ワークフラッシュメモリの動作説明 .....	57
3.3.1 読出し/リセット動作 .....	57
3.3.2 書込み動作 .....	57
3.3.3 フラッシュ消去動作 .....	59
3.3.4 セクタ消去動作 .....	59
3.3.5 セクタ消去一時停止動作 .....	62
3.3.6 セクタ消去再開動作 .....	63
3.4 ワークフラッシュメモリの使用上の注意 .....	64
<b>4. レジスタ</b> .....	<b>65</b>
4.1 WFASZR (WorkFlash Access Size Register) ワークフラッシュ・アクセスサイズ・レジスタ .....	66
4.2 WFRWTR (WorkFlash Read Wait Register) ワークフラッシュ・リードウェイト・レジスタ .....	67
4.3 WFSTR (WorkFlash Status Register) ワークフラッシュ・ステータス・レジスタ .....	68
<b>CHAPTER 3: フラッシュセキュリティ</b> .....	<b>69</b>
<b>1. 概要</b> .....	<b>70</b>
<b>2. 動作説明</b> .....	<b>71</b>
<b>CHAPTER 4: シリアル書込み接続</b> .....	<b>72</b>
<b>1. シリアルプログラマ</b> .....	<b>73</b>
1.1 基本構成 .....	74
1.2 使用端子 .....	81
<b>改訂履歴</b> .....	<b>82</b>
主な変更内容 .....	82
改訂履歴 .....	83

# CHAPTER 1: メインフラッシュメモリ



本シリーズは 512K~1024K バイトのメインフラッシュメモリおよび 32K バイトのワークフラッシュメモリが内蔵されています。

本章ではメインフラッシュメモリの概要, 構成, 動作およびレジスタについて説明します。

ワークフラッシュの詳細については『ワークフラッシュメモリ』の章を参照してください。本シリーズは 512K~1024K バイトの容量で、全セクタの一括データ消去およびセクタ単位でのデータ消去と CPU によるデータ書込みが可能なメインフラッシュメモリを内蔵しています。

本章でフラッシュメモリと記載されている内容はすべてメインフラッシュメモリを指しています。

1. 概要
2. 構成
3. 動作説明
4. レジスタ

## 1. 概要

本シリーズは 512K~1024K バイトのメインフラッシュメモリが内蔵されています。

メインフラッシュメモリは、Cortex-M4F CPU によるセクタ単位でのデータ消去、全セクター括データ消去およびハーフワード (16 ビット) 単位でのデータ書込みが可能です。

また、ECC (Error Correction Code) 機能が搭載されています。

### メインフラッシュメモリの特長

#### ■ 使用可能容量

最小構成 512K バイト

最大構成 1024K バイト

本シリーズは ECC 符号格納のため、上記に加えて 4 バイトにつき 7 ビットのフラッシュメモリが搭載されています。

#### ■ 高速フラッシュ

~72MHz 時 0Wait

~160MHz 時 フラッシュアクセラレータ機能(プリフェッチバッファ/トレースバッファ)を有効にすることによって高速動作周波数での 0Wait を実現

#### ■ 動作モード

##### 1. CPU ROM モード

フラッシュメモリのデータの読出しのみを行うモードです。ワードアクセスできます。ただし、このモードでは、書込みや消去、自動アルゴリズム\*1 の起動は行えません。

##### 2. CPU プログラミングモード

フラッシュメモリの読出し、書込みまたは消去ができるモードです (自動アルゴリズム\*1)。ワードアクセスできないため、このモードで動作しているときはフラッシュメモリ上のプログラムは実行できません。ハーフワードアクセスが可能です。

##### 3. ROM ライタモード

ROM ライタからフラッシュメモリの読出し、書込みおよび消去ができます (自動アルゴリズム\*1)。

#### ■ フラッシュセキュリティ機能搭載

(第三者によるフラッシュメモリ内容読出し阻止)

フラッシュセキュリティ機能については『フラッシュセキュリティ』の章を参照してください。

#### ■ 1 ワード中の 1 ビットまでの誤りを訂正する ECC (Error Correction Code) 機能を内蔵しています (2 ビット誤り検出機能は搭載していません)。誤りは読出し中に自動で訂正されます。

また、ECC 符号はフラッシュメモリへの書込み時に自動で付加されます。誤り訂正による読出しサイクルペナルティはないため、ソフトウェア開発の際に誤り訂正ペナルティを考慮する必要はありません。

### <注意事項>

- 本書では、フラッシュメモリを CPU モードで利用する場合について記載します。  
ROM ライタからフラッシュメモリにアクセスする場合については、ご利用の ROM ライタの取扱い説明書を参照してください。

\*1: 自動アルゴリズム=Embedded Algorithm



## 2. 構成

本シリーズは 512K~1024K バイトのメインフラッシュメモリ領域、セキュリティコード領域、高速 CR トリミングデータ領域で構成されます。

本シリーズに搭載されているメインフラッシュメモリのアドレス、セクタ構成とセキュリティ/CR トリミングデータのアドレスを [Figure 2-1](#)~[Figure 2-4](#) に示します。

セキュリティについての詳細は、『フラッシュセキュリティ』の章を参照してください。

高速 CR トリミングデータの詳細は、「[4.9 CRTRMM \(CR Trimming Data Mirror Register\) CR トリミングデータ・ミラー・レジスタ](#)」と『FM4 ファミリ ペリフェラルマニュアル』の『高速 CR トリミング機能』の章を参照してください。

**Table 2-1 各製品のメインフラッシュメモリ容量**

メモリ容量	512KB	768KB	1024KB
製品名	CY9BF166	CY9BF167	CY9BF168
	CY9BF366	CY9BF367	CY9BF368
	CY9BF466	CY9BF467	CY9BF468
	CY9BF566	CY9BF567	CY9BF568
	MB9BF166	MB9BF167	MB9BF168
	MB9BF366	MB9BF367	MB9BF368
	MB9BF466	MB9BF467	MB9BF468
	MB9BF566	MB9BF567	MB9BF568

Figure 2-1 メインフラッシュメモリ 512KB のメモリマップ

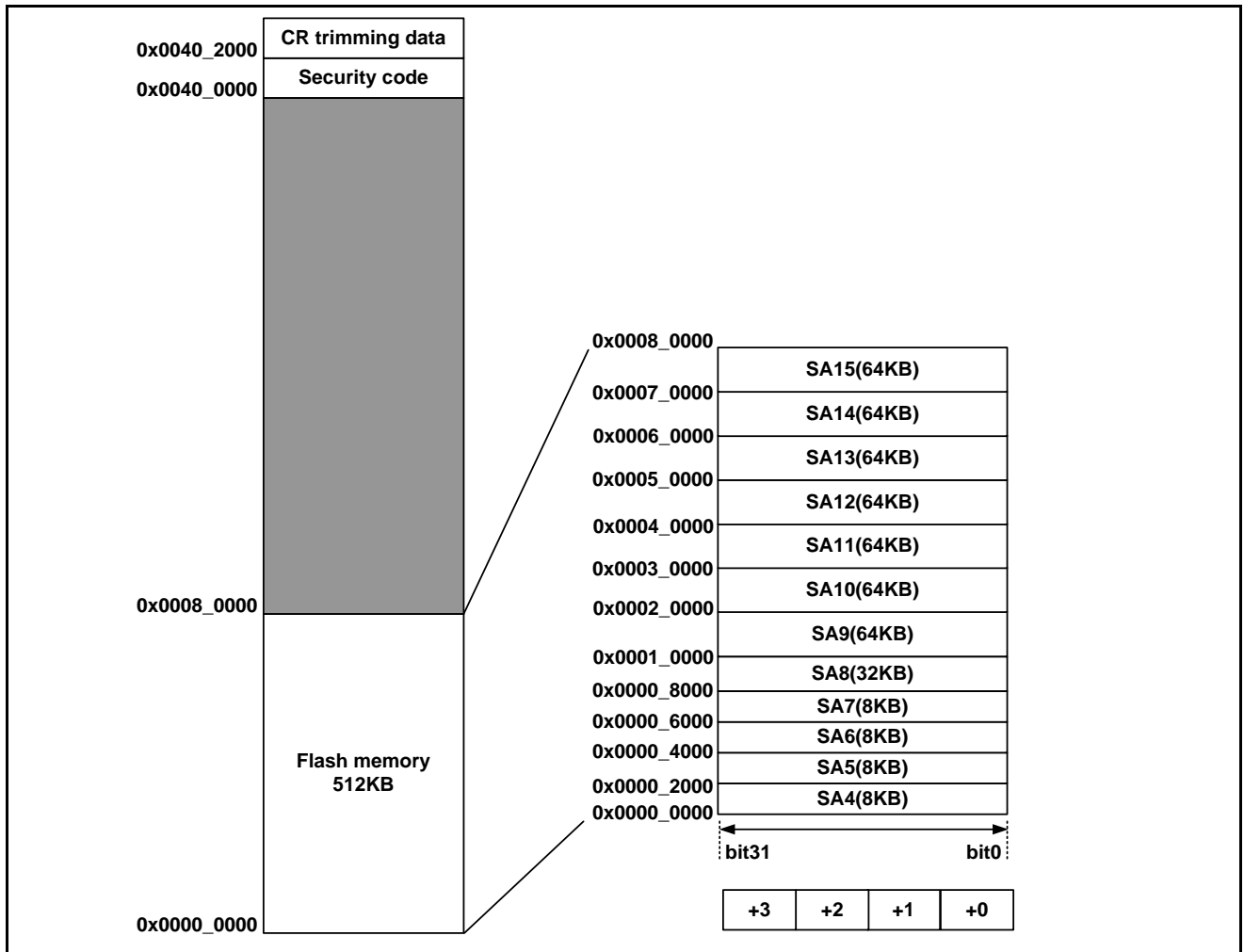


Figure 2-2 メインフラッシュメモリ 768KB のメモリマップ

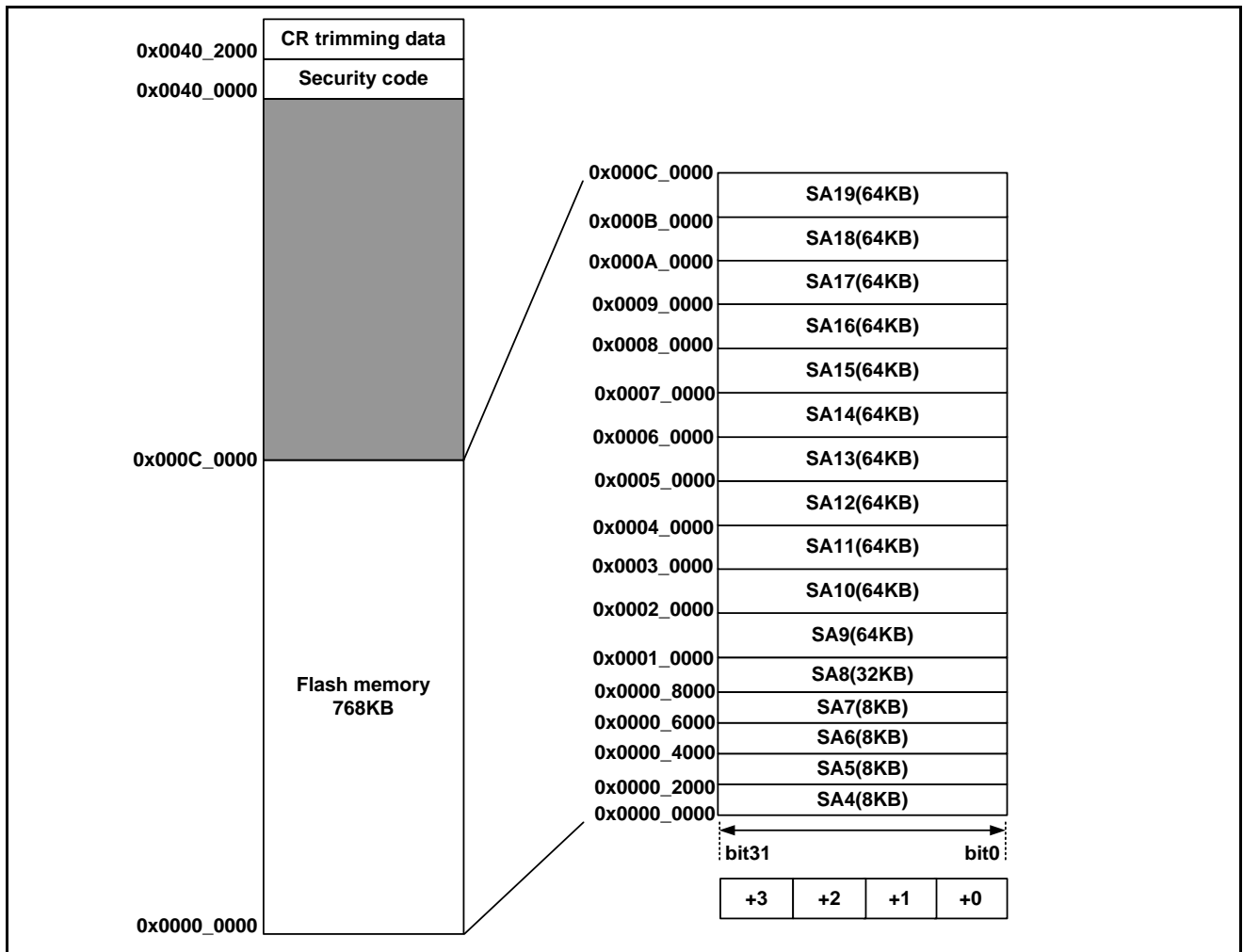


Figure 2-3 メインフラッシュメモリ 1024KB のメモリマップ

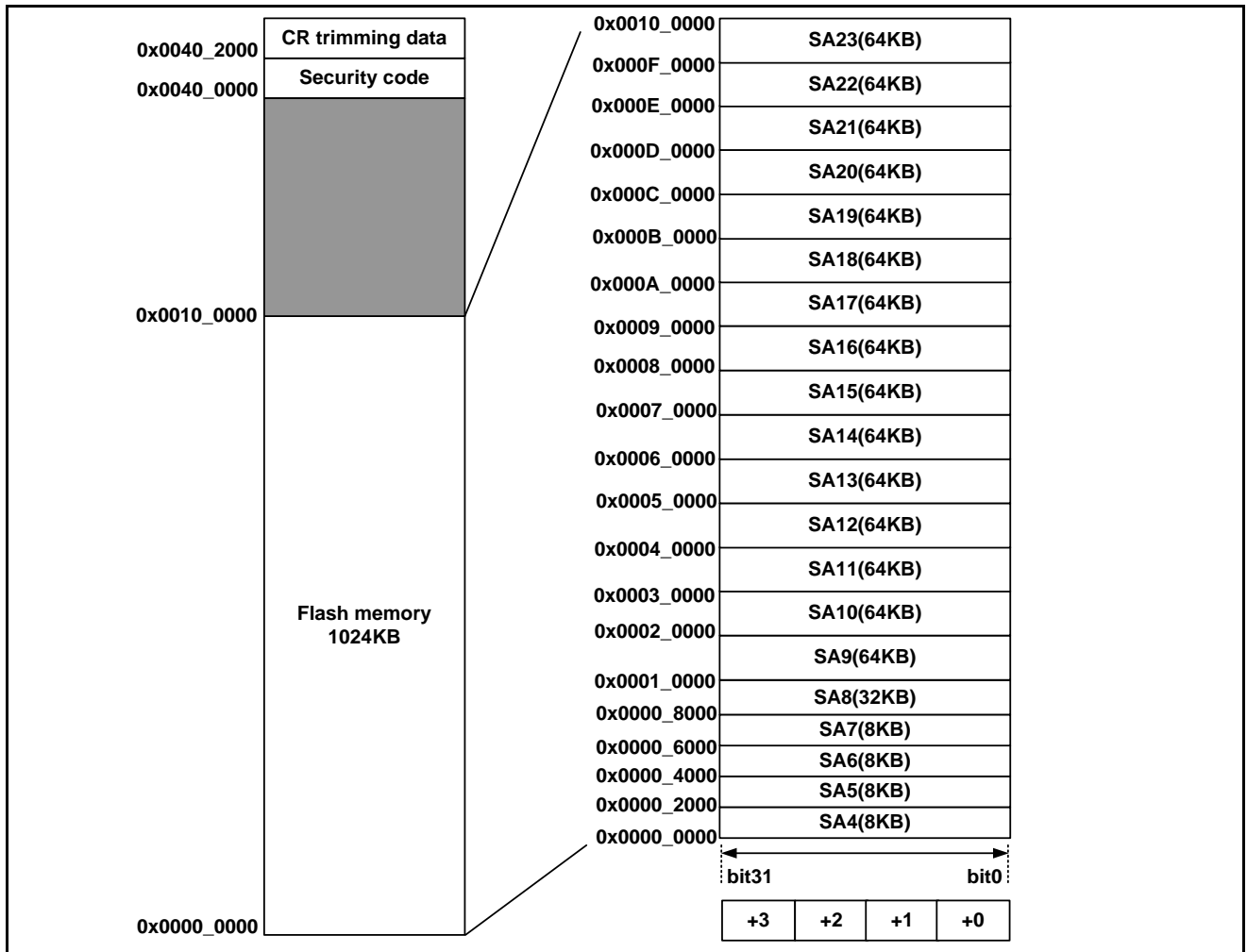


Figure 2-4 セキュリティ/CR トリミングデータのアドレス

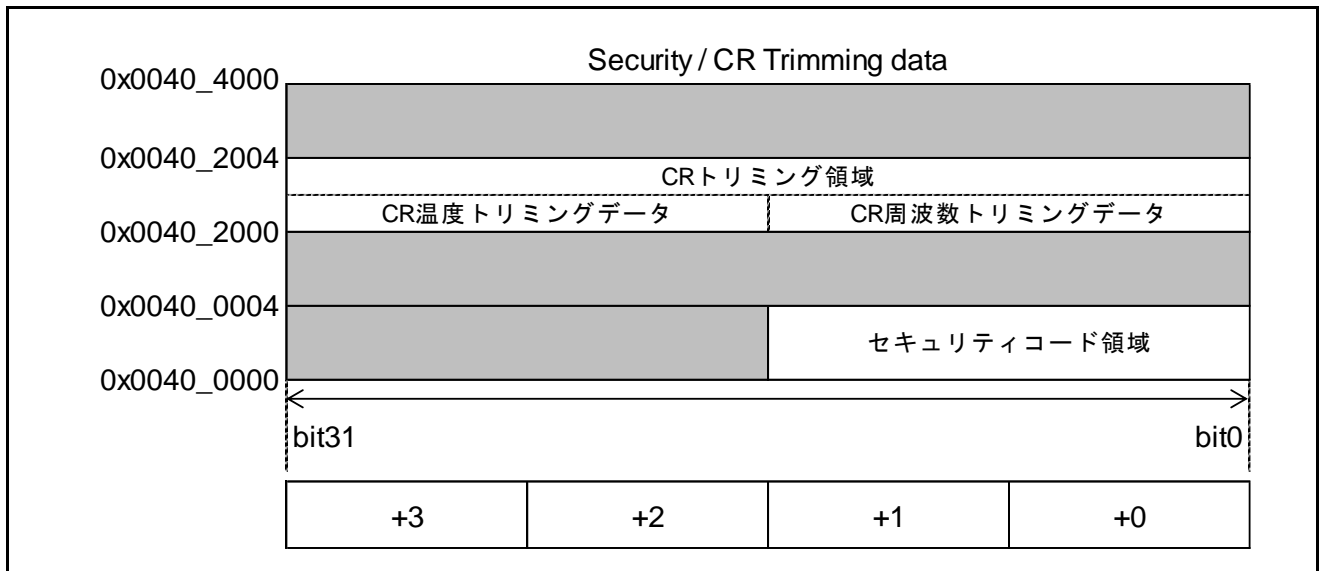
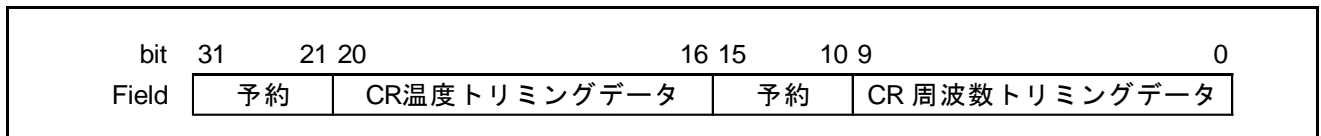


Figure 2-5 CR トリミング領域のビット構成



### 3. 動作説明

動作について説明します。

- 3.1 メインフラッシュメモリのアクセスモード
- 3.2 自動アルゴリズム
- 3.3 メインフラッシュメモリの動作説明
- 3.4 ECC 搭載品のメインフラッシュメモリへの書込みについて
- 3.5 メインフラッシュアクセラレータ
- 3.6 メインフラッシュメモリの使用上の注意

### 3.1 メインフラッシュメモリのアクセスモード

CPU からメインフラッシュメモリにアクセスする場合は次の 2 つのアクセスモードがあります。

- CPU ROM モード
- CPU プログラミングモード

フラッシュ・アクセスサイズ・ビット(FASZR:ASZ)で設定できます。

#### CPU ROM モード

フラッシュメモリのデータの読出しのみを行うモードです。

フラッシュ・アクセスサイズ・ビット(FASZR:ASZ)を"10"(32 ビット リード)に設定した場合に本モードになり、ワードアクセス可能になります。

ただし、このモードでは、コマンドおよびデータ書込みや消去、自動アルゴリズムの起動は行えません。

リセット解除後は必ず本モードになります。

#### CPU プログラミングモード

データの読出し、書込み/消去ができるモードです。

フラッシュ・アクセスサイズ・ビット(FASZR:ASZ)を"01"(16 ビット リード/ライト)に設定した場合に本モードになり、フラッシュプログラミング可能になります。

本モードでは、ワードアクセスできないため、このモードで動作しているときはフラッシュメモリ上のプログラムは実行できません。動作内容は次のとおりです。

- 読出し時

ハーフワードでフラッシュメモリにアクセスして、16 ビットのデータを一括で読み出します。

- コマンド書込み時

自動アルゴリズムを起動し書込みおよび消去を行います。自動アルゴリズムについては、「3.2 自動アルゴリズム」を参照してください。

Table 3-1 フラッシュメモリのアクセスモード

アクセスモード	アクセスサイズ	自動アルゴリズム	フラッシュメモリ上での命令実行
CPU ROM モード	32 ビット	不可能	可能
CPU プログラミングモード	16 ビット	可能	禁止

#### <注意事項>

- リセットが解除されると必ず CPU ROM モードが設定されます。このため CPU プログラミングモード設定後にリセットが発生すると、フラッシュ・アクセスサイズ・ビット(FASZR:ASZ)が"10"に変わり CPU ROM モードに戻ります。

## 3.2 自動アルゴリズム

CPU プログラミングモードを利用する場合、メインフラッシュメモリへの書き込み/消去は自動アルゴリズムを起動して行います。

自動アルゴリズムについて説明します。

### 3.2.1 コマンドシーケンス

### 3.2.2 コマンド動作説明

### 3.2.3 自動アルゴリズムの実行状態



### 3.2.1 コマンドシーケンス

メインフラッシュメモリへ1回~6回連続でハーフワード(16ビット)のデータを書き込むと自動アルゴリズムが起動します。これをコマンドとよびます。コマンドシーケンスを Table 3-2 に示します。

Table 3-2 コマンドシーケンス表

コマンド	書き込み回数	1回目		2回目		3回目		4回目		5回目		6回目	
		アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ
読出し/リセット	1	0xXXX	0xF0	--	--	--	--	--	--	--	--	--	--
書き込み	4	0xAA8	0xAA	0x554	0x55	0xAA8	0xA0	PA	PD	--	--	--	--
フラッシュ消去	6	0xAA8	0xAA	0x554	0x55	0xAA8	0x80	0xAA8	0xAA	0x554	0x55	0xAA8	0x10
セクタ消去	6	0xAA8	0xAA	0x554	0x55	0xAA8	0x80	0xAA8	0xAA	0x554	0x55	SA	0x30
セクタ消去一時停止	1	0xXXX	0xB0	--	--	--	--	--	--	--	--	--	--
セクタ消去再開	1	0xXXX	0x30	--	--	--	--	--	--	--	--	--	--

X: 任意

PA: 書き込みアドレス

SA: セクタアドレス(消去対象となるセクタのアドレス範囲内の任意のアドレスを指定してください)

PD: 書き込みデータ

#### <注意事項>

- Table 3-2 のデータ表記は下位 8 ビット分のみを表記しています。上位 8 ビットは任意です。
- コマンドは必ずハーフワードで書き込んでください。
- Table 3-2 のアドレス表記は下位 12 ビット分のみを表記しています。上位 20 ビット分は、対象となるフラッシュメモリのアドレス範囲内の任意のアドレスを必ず指定してください。フラッシュアドレス範囲外のアドレスを指定した場合、フラッシュメモリはコマンドを認識できないため、コマンドシーケンスは正しく動きません。
- フラッシュセキュリティコードを設定する場合のアドレスには 0x0040\_0000 を指定してください。
- CR トリミングデータを設定または消去する場合のアドレスには 0x0040\_2000 を指定してください。

### 3.2.2 コマンド動作説明

コマンド動作を説明します。

#### 読出し/リセットコマンド

読出し/リセットコマンドを対象セクタに連続して送るとフラッシュメモリを読出し/リセット状態にできます。読出し/リセットコマンドを発行すると、フラッシュメモリは、ほかのコマンドが発行されるまで読出し状態を保ちます。自動アルゴリズムの実行がタイミングリミットを超過した場合は、読出し/リセットコマンドを発行するとフラッシュメモリが読出し/リセット状態へ復帰します。実際の動作については、「[3.3.1 読出し/リセット動作](#)」を参照してください。

#### プログラム(書込み)コマンド

フラッシュメモリにデータを書き込むには、対象セクタに4回連続して書込みコマンドを発行し、自動アルゴリズムを起動してください。データの書込みはどのようなアドレスの順番でも、セクタの境界を越えても行えます。CPUプログラミングモードでは、ハーフワードで書込みを行います。4回目のコマンド発行が終了すると、自動アルゴリズムが起動し、フラッシュメモリへの自動書込みが開始されます。自動書込みアルゴリズムコマンドシーケンス実行後は、外部からフラッシュメモリを制御する必要はありません。実際の動作については、「[3.3.2 書込み動作](#)」を参照してください。

#### <注意事項>

- 4回目の書込みコマンド(書込みデータサイクル)を奇数番地に発行すると、コマンドの認識が正しく行われません。必ず偶数番地に書き込んでください。
- 1回の書込みコマンドシーケンスではハーフワードのデータ1つしか書き込めません。複数のデータを書き込みたい場合は、1データに1度書込みコマンドシーケンスを発行してください。

#### フラッシュ消去コマンド

フラッシュ消去コマンドを対象セクタに6回連続して送るとフラッシュメモリの全セクタのデータを一括で消去できます。6回目の書込みが終了すると、自動アルゴリズムが起動しフラッシュメモリの消去動作が開始されます。

#### セクタ消去コマンド

セクタ消去コマンドを対象セクタに6回連続して送るとフラッシュメモリのセクタのデータを消去できます。6回目の書込みが終了し、35 $\mu$ s経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタのデータを消去したい場合は、35 $\mu$ s(タイムアウト期間)以内に消去するセクタのアドレスへセクタ消去コマンドの6回目の書込みコードであるセクタ消去コード(0x30)を発行してください。タイムアウト期間内にセクタ消去コードを発行しなかった場合、タイムアウト期間を過ぎてから追加したセクタ消去コードが無効になることがあります。

#### セクタ消去一時停止コマンド

セクタ消去中またはコマンドタイムアウト中に、セクタ消去一時停止コマンドを発行することによって、セクタ消去を一時停止できます。セクタ消去一時停止状態では消去対象ではないセクタのメモリセルについて読出し、および書き込み動作が可能です。実際の動作については、「[3.3.5 セクタ消去一時停止動作](#)」を参照してください。

#### <注意事項>

- このコマンドはセクタ消去中のみ有効です。フラッシュ消去中や書込み中にこのコマンドを発行しても無視されません。

#### セクタ消去再開コマンド

セクタ消去一時停止状態から、中断していた消去動作を再開するためにはセクタ消去再開コマンドを発行してください。セクタ消去再開コマンドを発行するとセクタ消去状態に戻り消去動作が再開されます。実際の動作については、「[3.3.6 セクタ消去再開動作](#)」を参照してください。

#### <注意事項>

- このコマンドはセクタ消去一時停止中のみ有効です。セクタ消去中にこのコマンドを発行しても無視されます。

### 3.2.3 自動アルゴリズムの実行状態

メインフラッシュメモリでは、書込みや消去を自動アルゴリズムで行うため、自動アルゴリズムが実行中かどうかをフラッシュ・レディビット(FSTR:RDY)で、動作状態をハードウェアシーケンスフラグで確認できます。

#### ハードウェアシーケンスフラグ

自動アルゴリズムの状態を示すフラグです。フラッシュ・レディビット(FSTR:RDY)が"0"のときに、フラッシュメモリの任意のアドレスを読み出すと動作状態を確認できます。

ハードウェアシーケンスフラグのビット構成を Figure 3-1 に示します。

Figure 3-1 ハードウェアシーケンスフラグのビット構成

ハーフワードアクセスの場合	
bit	15 14 13 12 11 10 9 8
	未定義 未定義 未定義 未定義 未定義 未定義 未定義 未定義
bit	7 6 5 4 3 2 1 0
	DPOL TOGG TLOV 未定義 SETI TOGG2 未定義 未定義
バイトアクセスの場合	
bit	7 6 5 4 3 2 1 0
	DPOL TOGG TLOV 未定義 SETI TOGG2 未定義 未定義

#### <注意事項>

- ワードアクセスでは読み出せません。必ず、CPU プログラミングモードのときにハーフワードかバイトアクセスで読み出してください。
- CPU ROM モードで任意のアドレスを読み出しても、ハードウェアシーケンスフラグを読み出すことはできません。
- コマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後 1 回目のハードウェアシーケンスフラグの読出し値は無視してください。

#### 各ビットとメインフラッシュメモリの状態

ハードウェアシーケンスフラグの各ビットの状態とフラッシュメモリ状態の対応を Table 3-3 に示します。

Table 3-3 ハードウェアシーケンスフラグ状態一覧

状態		DPOL	TOGG	TLOV	SETI	TOGG2	
実行中	自動書込み動作	反転データ (*1)	トグル	0	0	0	
	フラッシュ消去	0	トグル	0	1	トグル	
		タイムアウト期間消去	0	トグル	0	0	トグル
	自動消去動作	セクタ消去	0	0	0	1	トグル
		セクタ消去一時停止	読出し (消去対象セクタ)	データ (*1)	データ (*1)	データ (*1)	データ (*1)
読出し (消去対象外セクタ)			データ (*1)	データ (*1)	データ (*1)	データ (*1)	
	自動書き込み動作 (消去対象外セクタ)	反転データ (*1)	トグル	0	1	0	
タイムリミット超過	自動書込み動作	反転データ (*1)	トグル	1	0	0	
	自動消去動作	0	トグル	1	1	トグル	

\*1: 読み出される値については、「ビット説明」を参照してください。

## ビット説明

### [bit15:8] 未定義ビット

#### [bit7] DPOL : データポーリングフラグビット

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、データポーリング機能で自動アルゴリズムが実行中かどうかをこのビットに示します。

読み出される値は動作状態によって異なります。

##### ■ 書込み時

###### - 書込み中:

書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

###### - 書込み終了後:

ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit7 の値が読み出されます。

##### ■ セクタ消去時

###### - セクタ消去実行中:

フラッシュメモリの全領域から "0" が読み出されます。

###### - セクタ消去後:

必ず "1" が読み出されます。

###### - フラッシュ消去時

- ・ フラッシュ消去実行中 : 必ず "0" が読み出されます。
- ・ フラッシュ消去後 : 必ず "1" が読み出されます。

##### ■ セクタ消去一時停止時

###### - セクタ消去指定したセクタのアドレスを指定して読み出した場合:

"0" が読み出されます。

###### - セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit7 の値が読み出されます。

###### - 書き込み中の場合:

書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

#### <注意事項>

- *自動アルゴリズムを起動しているときは、指定したアドレスのデータを読み出すことはできません。このビットで自動アルゴリズムの動作が終了していることを確認してから、データを読み出してください。*

#### [bit6] TOGG : トグルフラグビット

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムが実行中かどうかをこのビットで示します。

読み出される値は動作状態によって異なります。

##### ■ 書込み/セクタ消去/フラッシュ消去時

###### - 書込み/セクタ消去/フラッシュ消去中:

連続でこのビットを読み出すと、"1" と "0" が交互に読み出されます (トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

###### - 書込み/セクタ消去/フラッシュ消去終了後:

ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit6 の値が読み出されます。

##### ■ セクタ消去一時停止時

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:  
"0"が読み出されます。
- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:  
指定したアドレスの bit6 の値が読み出されます。
- 書き込み中の場合:  
連続でこのビットを読み出すと、"1"と"0"が交互に読み出されます (トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

#### [bit5] TLOV : タイミングリミット超過フラグビット

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムの実行時間がフラッシュメモリ内部で規定している時間 (内部パルスの回数) を超過したかどうかをこのビットで示します。

読み出される値は動作状態によって異なります。

##### ■ 書き込み/セクタ消去/フラッシュ消去時

次の値が読み出されます。

0 : 規定時間内

1 : 規定時間を超えている

このビットが"1"のときに、DPOL ビットや TOGG ビットが自動アルゴリズム実行中であることを示していると、書き込みや消去に失敗したことになります。

例えば、フラッシュメモリでは"0"が書き込まれているデータを"1"に書き換えることができません。このため、"0"が書き込まれているアドレスに"1"を書き込もうとすると、フラッシュメモリがロックされ自動アルゴリズムが終了しません。

この場合は、DPOL ビットの値は無効のままになり、TOGG ビットからは"1"と"0"が交互に読み出され続けます。

この状態のまま規定時間を超えたときに、このビットが"1"に変わります。このビットが"1"になった場合はリセットコマンドを発行してください。

##### ■ セクタ消去一時停止時

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:  
"0"が読み出されます。
- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:  
指定したアドレスの bit5 の値が読み出されます。
- 書き込み中の場合:  
次の値が読み出されます。  
0 : 規定時間内  
1 : 規定時間を超えている  
このビットが"1"のときに、DPOL ビットや TOGG ビットが自動アルゴリズム実行中であることを示していると、書き込みや消去に失敗したことになります。

#### <注意事項>

- このビットが"1"の場合は、フラッシュメモリが正しく使用されなかったことを示しています。フラッシュメモリの不良ではありません。  
リセットコマンドを発行してから適切な処理を行ってください。

#### [bit4] 未定義ビット

**[bit3] SET1 : セクタ消去タイムフラグビット**

セクタ消去時は、セクタ消去コマンドを発行してから実際にセクタ消去が開始されるまでには、35 $\mu$ sのタイムアウト期間が必要です。

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、セクタ消去コマンドのタイムアウト期間中かどうかをこのビットで示します。

読み出される値は動作状態によって異なります。

**■ セクタ消去時:**

セクタ消去時に次のセクタ消去コードを入力する前に、このビットを確認することで次のセクタ消去コードが受け付けられる状態かどうかを確認できます。

ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスせず、次の値が読み出されます。

0 : セクタ消去タイムアウト期間中

次のセクタ消去コード(0x30)を受け付けられます。

1 : セクタ消去タイムアウト期間を超過している

このときに、DPOL ビットや TOGG ビットが自動アルゴリズム実行中であることを示していると、フラッシュメモリ内部の消去が開始されています。この場合、セクタ消去一時停止 (0xB0) 以外のコマンドはフラッシュメモリ内部の消去が完了するまで無視されます。

**■ セクタ消去一時停止時**

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:

“1”が読み出されます。

- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit3 の値が読み出されます。

- 書き込み中の場合:

“1”が読み出されます。

**[bit2] TOGG2 : トグルフラグビット**

セクタ消去一時停止状態において、消去対象以外のセクタからは値を読み出せませんが、消去対象セクタからは読み出せません。TOGG2は、セクタ消去一時停止中、読み出したデータのトグル動作の有無により、該当セクタが消去対象であるかどうかを検出します。

**■ 書き込み時**

“0”が読み出されます。

**■ セクタ消去/フラッシュ消去時**

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます(トグル動作)。

**■ セクタ消去一時停止時**

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます(トグル動作)。

- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit2 の値が読み出されます。

- 書き込み中の場合:

“0”が読み出されます。

**[bit1:0] 未定義ビット**

### 3.3 メインフラッシュメモリの動作説明

メインフラッシュメモリの動作について、コマンドごとに説明します。

#### 3.3.1 読出し/リセット動作

#### 3.3.2 書込み動作

#### 3.3.3 フラッシュ消去動作

#### 3.3.4 セクタ消去動作

#### 3.3.5 セクタ消去一時停止動作

#### 3.3.6 セクタ消去再開動作

#### 3.3.1 読出し/リセット動作

読出し/リセット動作を説明します。

読出し/リセットコマンドを対象セクタに連続して送るとフラッシュメモリを読出し/リセット状態にできます。

この状態は、フラッシュメモリの初期状態のため、電源を投入したときやコマンドが正常終了すると、フラッシュメモリは常に読出し/リセット状態に戻ります。電源投入時はデータ読出しコマンドを発行する必要はありません。また、読出し/リセット状態では通常の読出しアクセスでデータを読み出したり、CPU からプログラムアクセスしたりできます。このため、データを読み出すときに読出し/リセットコマンドを発行する必要はありません。

#### 3.3.2 書込み動作

書込み動作を説明します。

次の順番で書込みを行います。

1. 書込みコマンドを対象セクタに連続して発行する

自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。

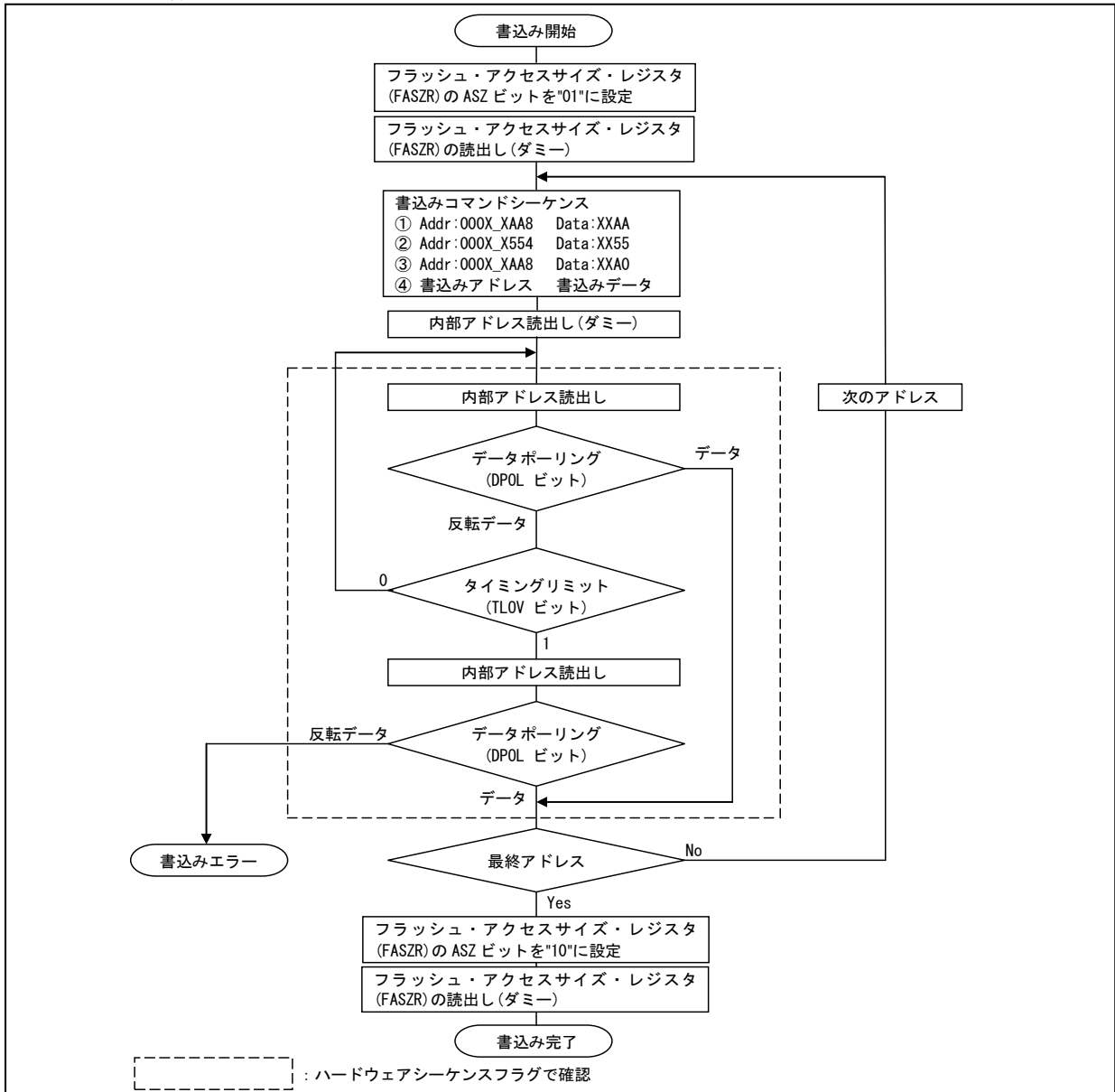
書込みコマンド発行後は外部からフラッシュメモリを制御する必要はありません。

2. 書込みを行ったアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が書き込んだ値と一致していると、フラッシュメモリへの書込みが終了したことになります。書込みが終了していない場合は、書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。

フラッシュメモリへの書込み動作例を [Figure 3-2](#) に示します。

Figure 3-2 書き込み手順例


**<注意事項>**

- 書き込みコマンドについては、「3.2 自動アルゴリズム」を参照してください。
- コマンドシーケンスのアドレス表記は下位 12 ビット分のみ表記しています。上位 20 ビット分は、対象となるフラッシュメモリアドレスのアドレス範囲内の任意のアドレスを必ず指定してください。フラッシュアドレス範囲外のアドレスを指定した場合、フラッシュメモリはコマンドを認識できないため、コマンドシーケンスは正しく動きません。
- ハードウェアシーケンスフラグの DPOL ビットは、TLOV ビットと同時に値が変わるため TLOV ビットが"1"の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG ビット、TLOV ビットが"1"になると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG ビットを再度確認する必要があります。
- フラッシュメモリへは、どのようなアドレスの順番でも、またセクタの境界を越えても書き込めます。しかし 1 回の書き込みコマンドシーケンスではハーファワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、1 データに 1 度書き込みコマンドシーケンスを発行してください。
- 書き込み動作中はフラッシュメモリへ発行されたすべてのコマンドが無視されます。



- 書き込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。
- 本シリーズでは ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビット単位で書き込みを行う必要があります。手順については「3.4 ECC 搭載品のメインフラッシュメモリへの書き込みについて」を参照してください。

### 3.3.3 フラッシュ消去動作

フラッシュ消去動作を説明します。

フラッシュメモリの全セクタのデータを一括して消去できます。これをフラッシュ消去とよびます。

フラッシュ消去コマンドを対象セクタに連続して送ると自動アルゴリズムを起動して、全セクタのデータを一括で消去できます。

フラッシュ消去コマンドについては、「3.2 自動アルゴリズム」を参照してください。

1. フラッシュ消去コマンドを対象セクタに連続して発行する

自動アルゴリズムが起動されフラッシュメモリのフラッシュ消去動作が開始されます。

2. 任意のアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が“1”だと、フラッシュ消去が終了したことになります。

フラッシュ消去に必要な時間は「セクタ消去時間×全セクタ数+フラッシュ書き込み時間 (プリプログラム)」となります。フラッシュ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

### 3.3.4 セクタ消去動作

セクタ消去動作を説明します。

フラッシュメモリ内のセクタを選択して、選択したセクタのデータのみを消去できます。複数のセクタも同時に指定できます。

次の順番でセクタ消去を行います。

1. セクタ消去コマンドを対象セクタに連続して発行する

35  $\mu$ s 経過 (タイムアウト期間) すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。

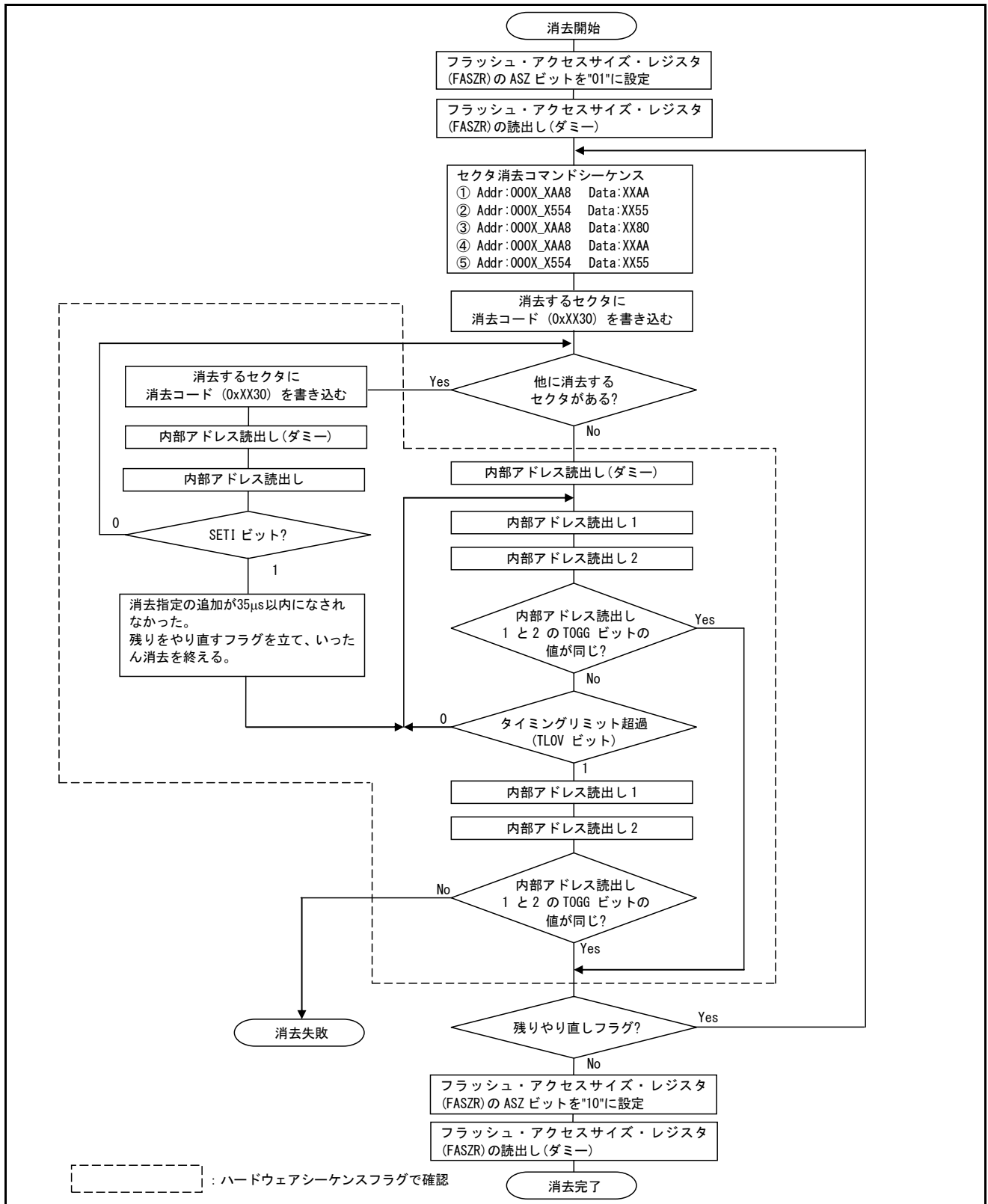
複数のセクタを消去したい場合は、35  $\mu$ s (タイムアウト期間) 以内に消去するセクタのアドレスに消去コード (0x30) を発行してください。タイムアウト期間経過後に発行しても、追加したセクタ消去コードが無効になることがあります。

2. 任意のアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が“1”だと、セクタ消去が終了したことになります。

また、TOGG ビットを利用してセクタ消去が完了したかどうか確認できます。確認動作に TOGG ビットを使用した場合を例にとって、セクタ消去手順例を Figure 3-3 に示します。

Figure 3-3 セクタ消去手順例



セクタ消去に必要な時間は「(セクタ消去時間+セクタ書込み時間 (プリプログラム))×セクタ数」となります。  
セクタ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

**<注意事項>**

- セクタ消去コマンドについては、「3.2 自動アルゴリズム」を参照してください。
- コマンドシーケンスのアドレス表記は下位 12 ビット分のみ表記しています。上位 20 ビット分は、対象となるフラッシュメモリアドレスのアドレス範囲内の任意のアドレスを必ず指定してください。フラッシュアドレス範囲外のアドレスを指定した場合、フラッシュメモリはコマンドを認識できないため、コマンドシーケンスは正しく動きません。
- ハードウェアシーケンスフラグの DPOL ビットは、TLOV ビットと同時に値が変わるため TLOV ビットが"1"の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG ビットは、TLOV ビットが"1"になると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG ビットを再度確認する必要があります。
- タイムアウト期間を含むセクタ消去中に、セクタ消去コマンド/消去一時停止コマンド以外のコマンドを発行すると、無視されます。

### 3.3.5 セクタ消去一時停止動作

セクタ消去一時停止動作を説明します。

セクタ消去中またはコマンドタイムアウト状態からセクタ消去一時停止コマンドを送付すると、セクタ消去一時停止状態に遷移し、消去動作を一時的に中断します。

消去再開コマンドを送付することによってセクタ消去状態に戻り、中断していた消去動作を再開できます。ただし、コマンドタイムアウト状態からセクタ消去一時停止状態に遷移した場合であっても、消去再開コマンドが正常に書き込まれた場合は、コマンドタイムアウト状態には遷移せず、セクタ消去状態に遷移して直ちにセクタ消去動作を再開します。

#### セクタ消去一時停止動作

次の順番でセクタ消去が一時停止されます。

1. コマンドタイムアウト期間～セクタ消去中の間に、セクタ消去一時停止コマンドをフラッシュメモリのアドレス範囲内の任意のアドレスへライトします。
2. コマンドタイムアウト期間中にセクタ消去一時停止コマンドを発行した場合は、直ちにタイムアウトを終了し、消去動作を一時停止します。セクタ消去中にセクタ消去一時停止コマンドを発行した場合、実際に消去を停止するまでに最大で 35 $\mu$ s の時間がかかります。

#### <注意事項>

- セクタ消去一時停止コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。
- セクタ消去を一時停止できるのは、セクタ消去のコマンドタイムアウト期間～セクタ消去中のみです。フラッシュ消去は一時停止できません。また、セクタ消去一時停止中に再度セクタ消去一時停止コマンドを発行しても無視されます。

#### セクタ消去一時停止後の状態

セクタ消去一時停止後に消去対象のセクタを読み出した場合、ハードウェアシーケンスフラグが読み出されます。また、消去対象でないセクタを読み出した場合は、メモリセルのデータが読み出されます。

#### <注意事項>

- セクタ消去一時停止状態で、新たな消去コマンドは無視されます。

### 3.3.6 セクタ消去再開動作

セクタ消去一時停止中にセクタ消去を再開する動作を説明します。

セクタ消去を一時停止中に、任意のアドレスにセクタ消去再開コマンドを発行するとセクタ消去を再開できます。

セクタ消去再開コマンドが発行されると、セクタ消去一時停止中のセクタ消去動作が再開されます。

セクタ消去再開コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。

#### <注意事項>

- セクタ消去再開コマンドは、セクタ消去一時停止中のみ有効です。セクタ消去中にセクタ消去再開コマンドを発行しても無視されます。
- セクタ消去再開コマンド発行後、セクタ消去動作が再開されるまで2ms以上かかります。そのため、この時間に満たない間隔で消去再開と消去停止を繰り返すと、消去動作が進まないままタイミングリミット超過になります。セクタ消去再開コマンド発行後に再度セクタ消去一時停止コマンドを発行する場合は、セクタ消去再開コマンドの発行後から2ms以上の間隔を空けてください。

### 3.4 ECC 搭載品のメインフラッシュメモリへの書込みについて

ECC 搭載品のメインフラッシュメモリへの書込みについて説明します。

本シリーズには 1 ワードごとの ECC(Error Correction Code)が付加されますので、1 ワードごとの書込みを行う必要があります。以下の手順で 1 ハーフワードを 2 連続に、1 ワードごとに書込みを行ってください。この手順に従わない場合、ECC 計算がされずにフラッシュメモリに書き込まれるため、書込み値を正しく読み出せません。

1. フラッシュアクセスサイズ設定を 16 ビットに設定してください。(FASZR:ASZ="01")  
設定後、必ず FASZR レジスタのダミーリードを行ってください。
2. 書込みコマンドを発行してください。書込みアドレス=PA 書込みデータ=PD[15:0]  
書込みコマンドについては「[3.3.2 書込み動作](#)」を参照してください。
3. ハードウェアシーケンスフラグを 1 回読み出してください。コマンド発行直後は正しい値を読み出せないことがあるため、この読出し値は無視してください。
4. 書込みが終了するまで、ハードウェアシーケンスフラグを読み出してください。  
ハードウェアシーケンスフラグ読出しについては「[3.2.3 自動アルゴリズムの実行状態](#)」を参照してください。
5. 書込みコマンドを発行してください。書込みアドレス=PA+2 書込みデータ=PD[31:16]  
この際、ハードウェアが自動で 2.の PD[15:0]とあわせて ECC 符号の計算を行い、ECC 符号の書込みも自動で同時に行われます。
6. ハードウェアシーケンスフラグを 1 回読み出してください。コマンド発行直後は正しい値を読み出せないことがあるため、この読出し値は無視してください。
7. 書込みが終了するまで、ハードウェアシーケンスフラグを読み出してください。
8. 書込みデータがまだある場合は 2.に戻ってください。すべて書込み完了した場合は 9.へ進んでください。
9. CPU ROM モードに設定してください。フラッシュアクセスサイズ設定を 32 ビットに設定してください。  
(FASZR:ASZ="10")設定後、必ず FASZR レジスタのダミーリードを行ってください。
10. 書込みした値を読み出して、正しい値を読み出せるか確認してください。また、正しい値が読み出せた場合でもフラッシュ ECC エラービット(FSTR : ERR)を確認して ECC 訂正がされていないか確認してください。ECC 訂正が発生していた場合はフラッシュメモリ消去を行い、最初からやり直してください。

PA :	書込み対象アドレス(ワードアラインド)
PD[31:0] :	書込みデータ
PD[31:16] :	書込みデータ上位 16 ビット分
PD[15:0] :	書込みデータ下位 16 ビット分

### 3.5 メインフラッシュアクセラレータ

メインフラッシュアクセラレータについて説明します。

本シリーズは高速(最大 160MHz)で 0Wait の CPU 動作を実現するために命令コード用のフラッシュアクセラレータを搭載しています。

フラッシュアクセラレータには以下の機能があります。

#### 1. プリフェッチバッファ機能

アドレスを先読みし、命令をプリフェッチバッファに格納します。プリフェッチバッファは 128 ビット×2 個用意され、このバッファにアドレスがヒットすると 0Wait で値を出力します。

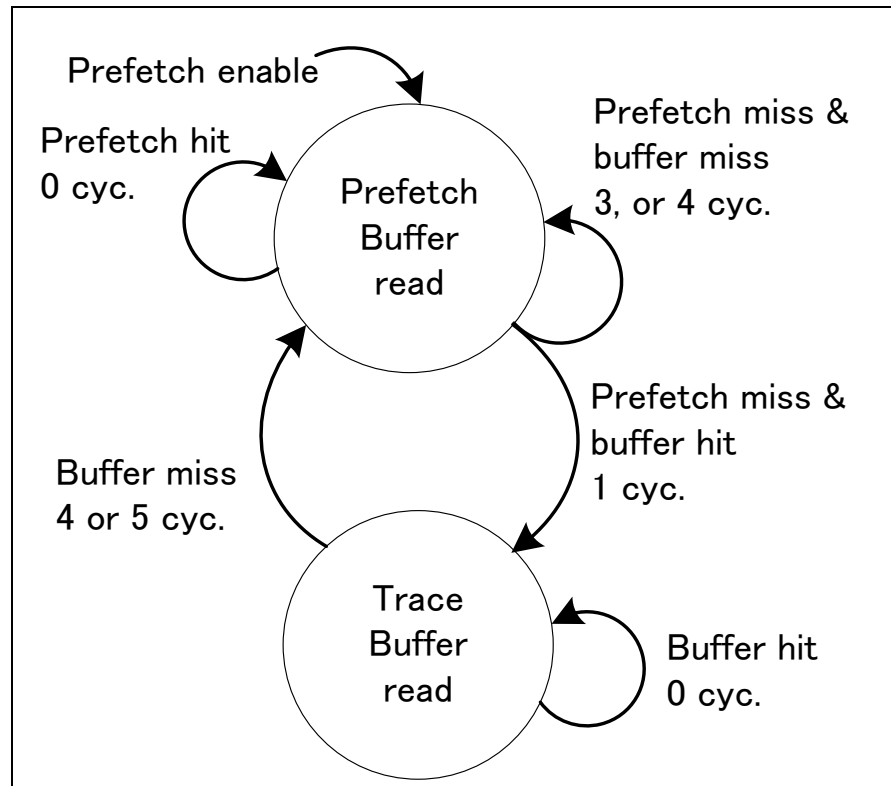
#### 2. トレースバッファ機能

トレースバッファとして 16K バイトの RAM を搭載しています。フラッシュメモリから読み出された値は常にこのバッファに格納されます。以後の命令フェッチでこのバッファにアドレスがヒットした場合はバッファヒットとなり、0Wait で値を出力します。

フラッシュアクセラレータ動作フローと Wait 数を図 3-4 に示します。

- 初期状態ではプリフェッチバッファにアクセスします。プリフェッチバッファにアドレスがヒットしない場合はプリフェッチミスとなり、1 サイクルの Wait 後にトレースバッファへのアクセスに切り換わります。しかし、その後はトレースバッファ側でアドレスがヒットした場合はバッファヒットとなり、0Wait でトレースバッファに格納された値を出力します。
- トレースバッファ側でアドレスがヒットしない場合はバッファミスとなり、再度プリフェッチバッファへのアクセスに切り換わります。その際はフラッシュメモリへのアクセスが発生し、4 サイクルまたは 5 サイクルの Wait サイクルが発生します。
- プリフェッチバッファおよびトレースバッファのいずれにもアドレスがヒットしない場合は、フラッシュメモリへのアクセスが発生し、3 サイクルまたは 4 サイクルの Wait サイクルが発生します。
- トレースバッファ機能をレジスタ設定により停止した場合、トレースバッファへのアクセス切換えは発生しません。(詳細は「[4.5 FBFCR \(Flash Buffer Control Register\) フラッシュ・バッファコントロール・レジスタ](#)」を参照してください。)プリフェッチミス時はフラッシュメモリへのアクセスが発生し、3 サイクルまたは 4 サイクルの Wait サイクルが発生します。

Figure 3-4 フラッシュアクセラレータ動作フロー



リセット後、FRWTR レジスタの RWT ビットは"10"となり、フラッシュアクセラレータのプリフェッチバッファ機能は動作していますが、トレースバッファ機能は停止しています。本機能を動作させるためには FBFCR (フラッシュ・バッファコントロール・レジスタ)の BE ビットに"1"を書き込む必要があります。詳しくは「[4.5 FBFCR \(Flash Buffer Control Register\) フラッシュ・バッファコントロール・レジスタ](#)」を参照してください。



### 3.6 メインフラッシュメモリの使用上の注意

メインフラッシュメモリの使用上の注意について説明します。

- 書き込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。また、書き込み・消去中には、ウォッチドッグタイマなどの予期しないリセットがかからないようにする必要があります。
- フラッシュ・アクセスサイズ・レジスタ(FASZR)の ASZ[1:0]ビットで CPU プログラミングモードを設定(ASZ=01)したときは、フラッシュメモリ上のプログラムを実行しないでください。正常な値を取り出せずにプログラムが暴走します。
- フラッシュ・アクセスサイズ・レジスタ(FASZR)の ASZ[1:0]ビットで CPU プログラミングモードを設定(ASZ=01)し、フラッシュメモリ上に割込みベクタテーブルがある場合は、割込み要求を発生させないでください。正常な値を取り出せずにプログラムが暴走します。
- フラッシュ・アクセスサイズ・レジスタ(FASZR)の ASZ[1:0]ビットで CPU プログラミングモードを設定(ASZ=01)した場合、低消費電力モードには遷移しないでください。
- フラッシュ・アクセスサイズ・レジスタ(FASZR)の ASZ[1:0]ビットで CPU ROM モードを設定(ASZ=10)した場合は、フラッシュメモリに書き込みを行わないでください。
- フラッシュ・アクセスサイズ・レジスタ(FASZR)の ASZ[1:0]ビットで CPU プログラミングモードを設定(ASZ=01)した場合、フラッシュメモリへの書き込みは必ずハーフワードで行ってください。バイト書き込みはしないでください。
- フラッシュメモリに対して自動アルゴリズムのコマンド発行直後は、必ずダミーの読出しをしてから実際に読み出したいデータを読み出してください。自動アルゴリズムコマンド発行直後にデータを読み出しても、読出し値は保証できません。
- 本デバイスを低消費電力モードへ遷移させる場合は、フラッシュメモリ自動アルゴリズムの動作が終了していることを必ず確認してください。  
低消費電力モードについては、『FM4 ファミリーペリフェラルマニュアル』の『低消費電力モード』の章を参照してください。
- 本シリーズでは ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビット単位で書き込みを行う必要があります。手順については「[3.4 ECC 搭載品のメインフラッシュメモリへの書き込みについて](#)」を参照してください。

## 4. レジスタ

レジスタについて説明します。

### レジスタ一覧

レジスタ略称	レジスタ名	参照先
FASZR	フラッシュ・アクセスサイズ・レジスタ	4.1
FRWTR	フラッシュ・リードウェイト・レジスタ	4.2
FSTR	フラッシュ・ステータス・レジスタ	4.3
FSYNDN	フラッシュ・シンクダウン・レジスタ	4.4
FBFCR	フラッシュ・バッファコントロール・レジスタ	4.5
FICR	フラッシュ・割込み制御・レジスタ	4.6
FISR	フラッシュ・割込みステータス・レジスタ	4.7
FICLR	フラッシュ・割込みクリア・レジスタ	4.8
CRTRMM	CR トリミングデータ・ミラー・レジスタ	4.9
FERRAD	フラッシュ・ECC ERR アドレス・キャプチャ・レジスタ	4.10

## 4.1 FASZR (Flash Access Size Register) フラッシュ・アクセスサイズ・レジスタ

FASZR について説明します。

フラッシュメモリに対するアクセスサイズを設定します。リセット解除後は ASZ="10"(32 ビットリード)になり、CPU ROM モードに設定されます。CPU プログラミングモードに設定する場合には、ASZ="01"に設定してください。

bit	7	6	5	4	3	2	1	0
Field	予約						ASZ	
属性							RW	RW
初期値							1	0

### [bit7:2] 予約ビット

読出し値は不定です。書き込み時は無視されます。

### [bit1:0] ASZ: Access Size

フラッシュメモリのアクセスサイズを指定します。

名称	bit	説明
ASZ	1:0	Flash Access Size 00: 設定禁止 01: 16 ビットリード/ライト(CPU プログラミングモード) 10: 32 ビットリード (CPU ROM モード: 初期値) 11: 設定禁止

### <注意事項>

- ASZ="01"の設定時、フラッシュに対しての書き込みは必ずハーフワードアクセス(16 ビットアクセス)を行ってください。
- 本レジスタを変更する場合にはフラッシュメモリに書き込まれた命令によって行わないでください。フラッシュ領域以外のプログラムにて書き換えてください。
- 本レジスタを変更後、必ずレジスタに対してダミーリードを行ってください。
- ASZ="01"の設定時、FBFCR レジスタの BS ビット, BE ビットはともに"0"にクリアされ、トレースバッファ機能はオフになります。

## 4.2 FRWTR (Flash Read Wait Register) フラッシュ・リードウェイト・レジスタ

FRWTR について説明します。

ASZ="10"(32ビットリードモード)のときに有効なレジスタです。  
フラッシュメモリに対するアクセス方法を設定します。

bit	7	6	5	4	3	2	1	0
Field	予約						RWT	
属性							RW	R
初期値							1	0

### [bit7:2] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit1:0] RWT: Read Wait Cycle

フラッシュメモリのアクセス方法を指定します。

名称	bit	説明
RWT	1:0	Read Wait Cycle 00: 0サイクルウェイトモード(0レイテンシ) HCLK が 72MHz 以下の場合に設定可能です。 01: 設定禁止 10: フラッシュアクセラレータモード(初期値) HCLK が 72MHz を超える場合には必ず設定してください。 11: 設定禁止

フラッシュアクセラレータモード(RWT="10")の場合、フラッシュアクセラレータのプリフェッチバッファ機能が動作し、高速周波数(最大 160MHz)の 0Wait 動作を可能にします。

フラッシュアクセラレータのプリフェッチバッファ機能を有効(RWT="10")にした後、トレースバッファ機能(「[4.5 FBFCR \(Flash Buffer Control Register\) フラッシュ・バッファコントロール](#)

ロール・レジスタ」を参照)を動作させることによりさらにパフォーマンスが向上します。

72MHz 以下の場合にはレイテンシの発生しない 0 サイクルウェイトモード(RWT="00") を推奨します。

### <注意事項>

- HCLK が 72MHz を超える場合は RWT=00 (0 サイクルウェイトモード)での使用は禁止です。RWT=00 の間、HCLK が一瞬でも 72MHz を超えないようにしてください。
- 本レジスタを変更後、必ずレジスタに対してダミーリードを行ってください。

### 4.3 FSTR (Flash Status Register) フラッシュ・ステータス・レジスタ

FSTR について説明します。

フラッシュメモリのステータスを示すレジスタです。

bit	7	6	5	4	3	2	1	0
Field	Reserved					ERR	HNG	RDY
属性						RW	R	R
初期値						0	0	X

#### [bit7:3] 予約ビット

読出し値は不定です。書込み時は無視されます。

#### [bit2] ERR : Flash ECC Error

ECC エラー訂正が発生すると本ビットは"1"になります。

名称	bit	説明
ERR	2	Flash ECC Error リード時 0: ECC エラーによる訂正は発生していません。 1: ECC エラーによる訂正が発生しています。 ライト時 0: 本ビットをクリアします。 1: 無視されます。

#### [bit1] HNG : Flash Hang

フラッシュメモリの HANG 状態を示します。タイミング超過(「[bit5] TLOV : タイミングリミット超過フラグビット」参照)すると HANG 状態になります。本ビットが"1"になった場合はリセットコマンド(「3.2.1 コマンドシーケンス」参照)を発行してください。

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後 1 回目の本ビット読出し値は無視してください。

名称	bit	説明
HNG	1	Flash Hang 0: フラッシュメモリの HANG 状態を検出していません。 1: フラッシュメモリの HANG 状態を検出しました。

#### [bit0] RDY : Flash Rdy

自動アルゴリズムでフラッシュメモリの書込み/消去動作が実行中か完了しているかを示します。動作中の場合、フラッシュメモリヘータの書込みや消去はできません。

名称	bit	説明
RDY	0	Flash Rdy 0: 動作中(書込み/消去不可) 1: 動作完了(書込み/消去可能)

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後 1 回目の本ビット読出し値は無視してください。

## 4.4 FSYNDN (Flash Sync Down Register) フラッシュ・シンクダウン・レジスタ

FSYNDN について説明します。

CPU ROM モード時、フラッシュメモリへのリードアクセスに対し、ウェイトサイクルを挿入します。これによりフラッシュへのアクセスを低速化することで、消費電流を減らすことができます。

bit	7	6	5	4	3	2	1	0
Field	Reserved					SD		
属性						RW	RW	RW
初期値						0	0	0

### [bit7:3] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit2:0] SD: Sync Down

フラッシュメモリのリードアクセスに対してウェイトサイクルを挿入します。

名称	bit	説明
SD	2:0	000: 0(初期値) 001: +1 ウェイト 010: 設定禁止 011: +3 ウェイト 100: 設定禁止 101: +5 ウェイト 110: 設定禁止 111: +7 ウェイト

本ビットで設定したウェイト数はフラッシュ・リードウェイト・レジスタ(FRWTR)の RWT ビットに対して追加されます。

例)

RWT=00(0 サイクルウェイト), SD=011 の場合、 0+3=3 ウェイト

### <注意事項>

- 本レジスタは FRWTR レジスタの RWT が"00"の場合のみ有効です。フラッシュアクセラレータモード時(RWT=10)は本レジスタの値は無視されます。
- 本レジスタを変更後、必ずレジスタに対してダミーリードを行ってください。

## 4.5 BFCR (Flash Buffer Control Register) フラッシュ・バッファコントロール・レジスタ

BFCR について説明します。

フラッシュアクセラレータモード時(FRWTR レジスタの RWT=10 設定時)、本レジスタによりフラッシュアクセラレータのトレースバッファ機能を動作許可することで、パフォーマンスがさらに向上します。

bit	7	6	5	4	3	2	1	0
Field	Reserved						BS	BE
属性							R	RW
初期値							0	0

### [bit7:2] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit1] BS: Buffer Status

名称	bit	説明
BS	1	Buffer Status 0: トレースバッファ機能は停止、または初期化中です。(初期値) 1: トレースバッファ機能は動作状態です。

### [bit0] BE: Buffer Enable

名称	bit	説明
BE	0	Buffer Enable 0: トレースバッファ機能を停止します。(初期値) 1: トレースバッファ機能を動作許可します。

トレースバッファ機能を動作許可にした後(BE ビットに"1"を書込み後)、トレースバッファの初期化が開始されます。HCLK×1025 サイクル後、初期化が完了し、トレースバッファは動作状態になります。このとき BS ビットは"1"に設定されます。

トレースバッファ初期化中(BE=1 かつ BS=0)もプリフェッチバッファは機能しているため、フラッシュメモリへのアクセスは行えます。BS=1 に変化したトレースバッファが動作状態になると自動的にトレースバッファはトレースを開始します。

## 4.6 FICR (Flash Interrupt Control Register) フラッシュ・割込み制御・レジスタ

FICR について説明します。

フラッシュメモリの割込み許可の設定を行うレジスタです。

bit	7	6	5	4	3	2	1	0
Field	予約					ERRIE	HNGIE	RDYIE
属性						RW	RW	RW
初期値						0	0	0

### [bit7:3] 予約ビット

読出し値は不定です。書き込み時は無視されます。

### [bit2] ERRIE : Flash ECC Error Interrupt Enable

ECC エラー訂正割込みを許可するビットです。FISR レジスタの ERRIF ビットが"1"で本ビットが"1"の時、CPU に対して割込みを発生します。

名称	bit	説明
ERRIE	2	Flash ECC Error Interrupt Enable 0: ECC エラー訂正割込み禁止。(初期値) 1: ECC エラー訂正割込み許可。

### [bit1] HNGIE : Flash HANG Interrupt Enable

フラッシュ HANG 状態割込みを許可するビットです。FISR レジスタの HNGIF ビットが"1"で本ビットが"1"の時、CPU に対して割込みを発生します。

名称	bit	説明
HNGIE	1	Flash HANG Interrupt Enable 0: フラッシュ HANG 割込み禁止。(初期値) 1: フラッシュ HANG 割込み許可。

### [bit0] RDYIE : Flash RDY Interrupt Enable

フラッシュ RDY 状態割込みを許可するビットです。FISR レジスタの RDYIF ビットが"1"で本ビットが"1"の時、CPU に対して割込みを発生します。

名称	bit	説明
RDYIE	0	Flash RDY Interrupt Enable 0: フラッシュ RDY 割込み禁止。(初期値) 1: フラッシュ RDY 割込み許可。



## 4.7 FISR (Flash Interrupt Status Register) フラッシュ・割込みステータス・レジスタ

FISR について説明します。

フラッシュメモリの割込み状態を示すレジスタです。

bit	7	6	5	4	3	2	1	0
Field	予約					ERRIF	HNGIF	RDYIF
属性						R	R	R
初期値						0	0	0

### [bit7:3] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit2] ERRIF : Flash ECC Error Interrupt Flag

フラッシュリードデータの ECC エラー訂正の発生を検出した場合"1"にセットされます。ERR 信号の立上りエッジでセットされます。FICLR レジスタの ERRIC ビットへの"1"書込みでクリアされます。

名称	bit	説明
ERRIF	2	Flash ECC Error Interrupt Flag 0: ECC エラー訂正の発生を検出していない。(初期値) 1: ECC エラー訂正の発生を検出。

### [bit1] HNGIF : Flash HANG Interrupt Flag

フラッシュ HANG 状態を検出した場合"1"にセットされます。HNG 信号の立上りエッジでセットされます。FICLR レジスタの HNGIC ビットへの"1"書込みでクリアされます。

名称	bit	説明
HNGIF	1	Flash HANG Interrupt Flag 0: フラッシュ HANG 状態を検出していない。(初期値) 1: フラッシュ HANG 状態を検出。

### [bit0] RDYIF : Flash RDY Interrupt Flag

フラッシュ RDY 状態を検出した場合"1"にセットされます。RDY 信号の立上りエッジでセットされます。FICLR レジスタの RDYIC ビットへの"1"書込みでクリアされます。

名称	bit	説明
RDYIF	0	Flash RDY Interrupt Flag 0: フラッシュ RDY 状態を検出していない。(初期値) 1: フラッシュ RDY 状態を検出。

## 4.8 FICLR (Flash Interrupt Clear Register) フラッシュ・割込みクリア・レジスタ

FICLR について説明します。

フラッシュメモリの割込み状態をクリアするレジスタです。

bit	7	6	5	4	3	2	1	0
Field	予約					ERRIC	HNGIC	RDYIC
属性						RW	RW	RW
初期値						0	0	0

### [bit7:3] 予約ビット

読出し値は不定です。書き込み時は無視されます。

### [bit2] ERRIC : Flash ECC Error Interrupt Clear

ERR 割込みフラグのクリアビットです。本ビットへ"1"を書き込むことで FISR レジスタの ERRIF ビットを"0"にクリアします。

名称	bit	説明
ERRIC	2	Flash ECC Error Interrupt Clear ライト時 0: ECC エラー訂正割込みフラグ(ERRIF)は変化しない。 1: ECC エラー訂正割込みフラグ(ERRIF)をクリア。 リード時 "0"が読み出されます。

### [bit1] HNGIC : Flash HANG Interrupt Clear

HNG 割込みフラグのクリアビットです。本ビットへ"1"を書き込むことで FISR レジスタの HNGIF ビットを"0"にクリアします。

名称	bit	説明
HNGIC	1	Flash HANG Interrupt Clear ライト時 0: フラッシュ HANG 割込みフラグ(HNGIF)は変化しない。 1: フラッシュ HANG 割込みフラグ(HNGIF)をクリア。 リード時 "0"が読み出されます。

### [bit0] RDYIC : Flash RDY Interrupt Clear

RDY 割込みフラグのクリアビットです。本ビットへ"1"を書き込むことで FISR レジスタの RDYIF ビットを"0"にクリアします。

名称	bit	説明
RDYIC	0	Flash RDY Interrupt Clear ライト時 0: フラッシュ RDY 割込みフラグ(RDYIF)は変化しない。 1: フラッシュ RDY 割込みフラグ(RDYIF)をクリア。 リード時 "0"が読み出されます。

## 4.9 CRTRMM (CR Trimming Data Mirror Register) CR トリミングデータ・ミラー・レジスタ

CRTRMM について説明します。

このレジスタは CR トリミングデータのミラーレジスタです。  
ユーザモード、シリアルライターモードで本レジスタの値を使用できます。

bit	31	20	16	9	0
Field	予約		TTRMM	予約	TRMM
属性			R		R
初期値			*		*

### [bit31:21] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit20:16] TTRMM : Temperature CR Trimming Data Mirror

リセット解除後、フラッシュメモリ領域のアドレス 0x0040\_2002 番地の bit[4:0] (CR 温度トリミングデータ) が本ビットに格納されます。

CR 温度トリミングデータの詳細は、『FM4 ファミリ パリフェラルマニュアル』の『高速 CR トリミング機能』の章を参照してください。

名称	bit	説明
TTRMM	20:16	*:アドレス 0x0040_2002 番地の bit[4:0]が読み出されます。

### [bit15:10] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit9:0] TRMM : CR Trimming Data Mirror

リセット解除後、フラッシュメモリ領域のアドレス 0x0040\_2000 番地の bit[9:0](CR 周波数トリミングデータ)が本レジスタに格納されます。

CR 周波数トリミングデータの詳細は、『FM4 ファミリ パリフェラルマニュアル』の『高速 CR トリミング機能』の章を参照してください。

名称	bit	説明
TRMM	9:0	*:アドレス 0x0040_2000 番地の bit[9:0]が読み出されます。

#### <注意事項>

- フラッシュメモリのデータを消去した後、チップ内部にリセットが発行されると本レジスタはクリアされます。このとき、格納されている CR トリミングデータは消失します。そのため、本レジスタがクリアされる前にレジスタに格納されている CR トリミングデータを RAM などに退避してください。

## 4.10 FERRAD (Flash ECC ERR Address Capture Register) フラッシュ・ECC ERR アドレス・キャプチャ・レジスタ

FERRAD について説明します。

フラッシュメモリのリードデータの ECC エラー訂正発生時のアドレスを格納するレジスタです。

bit	31		22	0
Field	予約		ERRAD	
属性			R	
初期値			0	

### [bit31:23] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit22:0] ERRAD : Flash ECC Error Address Capture

フラッシュメモリのリードデータの ECC エラー訂正発生時のアドレスが格納されます。

名称	bit	説明
ERRAD	22:0	ECC エラー訂正発生時のアドレスが格納されます。

### <注意事項>

- 一度格納したアドレスは次に FSTR レジスタの ERR ビットが再度"1"にセットされるまで保持されます。すなわち、FSTR.ERR ビットをクリアしない限り、連続で ERR が発生しても最初に取り込んだアドレスを保持します。

# CHAPTER 2: ワークフラッシュメモリ



本シリーズは 512K~1024K バイトのメインフラッシュメモリおよび 32K バイトのワークフラッシュメモリが内蔵されています。

本章ではワークフラッシュメモリの概要, 構成, 動作およびレジスタについて説明します。

メインフラッシュの詳細については『メインフラッシュメモリ』の章を参照してください。本シリーズは 32K バイトの容量で、全セクタの一括データ消去およびセクタ単位でのデータ消去と CPU によるデータ書込みが可能なワークフラッシュメモリを内蔵しています。

本章でフラッシュメモリと記載されている内容はすべてワークフラッシュメモリを指しています。

1. 概要
2. 構成
3. 動作説明
4. レジスタ

## 1. 概要

本シリーズは 32K バイトのワークフラッシュメモリが内蔵されています。

ワークフラッシュメモリは、Cortex-M4F CPU によるセクタ単位でのデータ消去、全セクター一括データ消去およびハーフワード(16 ビット)単位でのデータ書込みが可能です。

### ワークフラッシュメモリの特長

#### ■ 使用可能容量

32K バイト (8K バイト×4 セクタ)

#### ■ 高速フラッシュ

～40MHz 時 0Wait で読出し可能

～72MHz 時 2Wait で読出し可能

～120MHz 時 4Wait で読出し可能

～160MHz 時 6Wait で読出し可能

#### ■ 動作モード

##### 1. CPU ROM モード

フラッシュメモリのデータの読出しのみを行うモードです。ワードアクセスできます。ただし、このモードでは、書込みや消去、自動アルゴリズム\*1 の起動は行えません。

##### 2. CPU プログラミングモード

フラッシュメモリの読出し、書込みまたは消去ができるモードです (自動アルゴリズム\*1)。ワードアクセスできないため、このモードで動作しているときはフラッシュメモリ上のプログラムは実行できません。ハーフワードアクセスが可能です。

##### 3. ROM ライタモード

ROM ライタからフラッシュメモリの読出し、書込みおよび消去ができます (自動アルゴリズム\*1)。

#### ■ フラッシュセキュリティ機能搭載

(第三者によるフラッシュメモリ内容読出し阻止)

フラッシュセキュリティ機能については『フラッシュセキュリティ』の章を参照してください。

### <注意事項>

- 本書では、フラッシュメモリを CPU モードで利用する場合について記載します。  
ROM ライタからフラッシュメモリにアクセスする場合には、ご利用の ROM ライタの取扱い説明書を参照してください。

\*1: 自動アルゴリズム=Embedded Algorithm

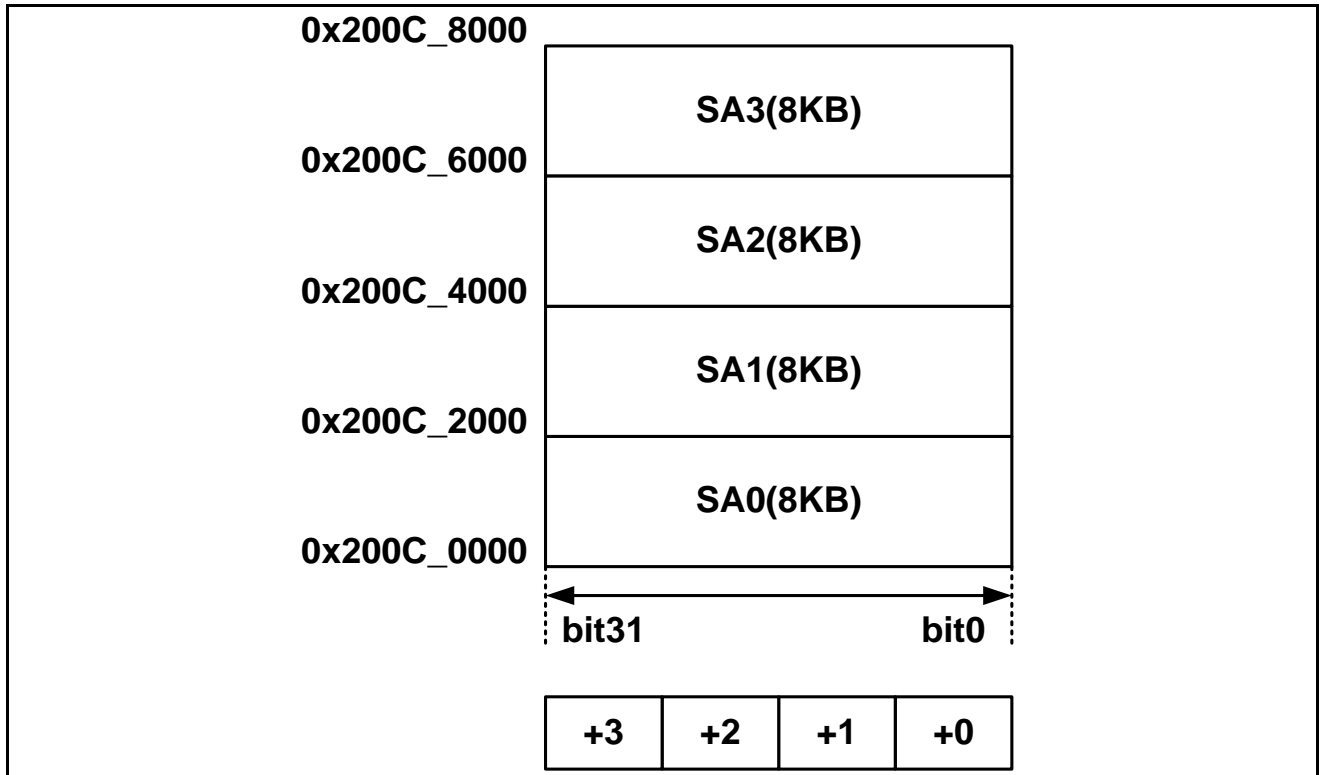
## 2. 構成

本シリーズは 32K バイトのワークフラッシュメモリ領域で構成されます。

本シリーズに搭載されているワークフラッシュメモリのアドレス、セクタ構成を [Figure 2-1](#) に示します。

セキュリティについての詳細は、『フラッシュセキュリティ』の章を参照してください。

**Figure 2-1** ワークフラッシュメモリのメモリマップ



### 3. 動作説明

動作について説明します。

3.1 ワークフラッシュメモリのアクセスモード

3.2 自動アルゴリズム

3.3 ワークフラッシュメモリの動作説明

3.4 ワークフラッシュメモリの使用上の注意



### 3.1 ワークフラッシュメモリのアクセスモード

CPUからワークフラッシュメモリにアクセスする場合は次の2つのアクセスモードがあります。

- CPU ROM モード
- CPU プログラミングモード

ワークフラッシュ・アクセスサイズ・ビット(WFASZR : ASZ)で設定できます。

#### CPU ROM モード

フラッシュメモリのデータの読出しのみを行うモードです。

ワークフラッシュ・アクセスサイズ・ビット(WFASZR : ASZ)を"1"(32ビットリード)に設定した場合に本モードになり、ワードアクセス可能になります。

ただし、このモードでは、コマンドおよびデータ書込みや消去、自動アルゴリズムの起動は行えません。

リセット解除後は必ず本モードになります。

#### CPU プログラミングモード

データの読出し、書込み/消去ができるモードです。

ワークフラッシュ・アクセスサイズ・ビット(WFASZR : ASZ)を"0"(16ビットリード/ライト)に設定した場合に本モードになり、フラッシュプログラミング可能になります。

本モードでは、ワードアクセスできないため、このモードで動作しているときはフラッシュメモリ上のプログラムは実行できません。動作内容は次のとおりです。

- 読出し時
  - ハーフワードでフラッシュメモリにアクセスして、16ビットのデータを一括で読み出します。
- コマンド書込み時
  - 自動アルゴリズムを起動し書込みおよび消去を行います。自動アルゴリズムについては、「3.2 自動アルゴリズム」を参照してください。

Table 3-1 フラッシュメモリのアクセスモード

アクセスモード	アクセスサイズ	自動アルゴリズム	フラッシュメモリ上での命令実行
CPU ROM モード	32 ビット	不可能	可能
CPU プログラミングモード	16 ビット	可能	禁止

#### <注意事項>

- リセットが解除されると必ずCPU ROM モードが設定されます。このためCPU プログラミングモード設定後にリセットが発生すると、ワークフラッシュ・アクセスサイズ・ビット(WFASZR : ASZ)が"1"に変わりCPU ROM モードに戻ります。

## 3.2 自動アルゴリズム

CPU プログラミングモードを利用する場合、ワークフラッシュメモリへの書き込み/消去は自動アルゴリズムを起動して行います。

自動アルゴリズムについて説明します。

### 3.2.1 コマンドシーケンス

### 3.2.2 コマンド動作説明

### 3.2.3 自動アルゴリズムの実行状態

### 3.2.1 コマンドシーケンス

ワークフラッシュメモリへ1回~6回連続でハーフワード(16ビット)のデータを書き込むと自動アルゴリズムが起動します。これをコマンドとよびます。コマンドシーケンスを Table 3-2 に示します。

Table 3-2 コマンドシーケンス表

コマンド	書き込み回数	1回目		2回目		3回目		4回目		5回目		6回目	
		アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ
読出し/ リセット	1	0xXXX	0xF0	--	--	--	--	--	--	--	--	--	--
書き込み	4	0xAA8	0xAA	0x554	0x55	0xAA8	0xA0	PA	PD	--	--	--	--
フラッシュ 消去	6	0xAA8	0xAA	0x554	0x55	0xAA8	0x80	0xAA8	0xAA	0x554	0x55	0xAA8	0x10
セクタ 消去	6	0xAA8	0xAA	0x554	0x55	0xAA8	0x80	0xAA8	0xAA	0x554	0x55	SA	0x30
セクタ 消去 一時停止	1	0xXXX	0xB0	--	--	--	--	--	--	--	--	--	--
セクタ 消去再開	1	0xXXX	0x30	--	--	--	--	--	--	--	--	--	--

X: 任意

PA: 書き込みアドレス

SA: セクタアドレス(消去対象となるセクタのアドレス範囲内の任意のアドレスを指定してください)

PD: 書き込みデータ

#### <注意事項>

- Table 3-2 のデータ表記は下位 8 ビット分のみを表記しています。上位 8 ビットは任意です。
- コマンドは必ずハーフワードで書き込んでください。
- Table 3-2 のアドレス表記は下位 12 ビット分のみを表記しています。上位 20 ビット分は、対象となるフラッシュメモリのアドレス範囲内の任意のアドレスを必ず指定してください。フラッシュアドレス範囲外のアドレスを指定した場合、フラッシュメモリはコマンドを認識できないため、コマンドシーケンスは正しく動きません。

## 3.2.2 コマンド動作説明

コマンド動作を説明します。

### 読出し/リセットコマンド

読出し/リセットコマンドを対象セクタに連続して送るとフラッシュメモリを読出し/リセット状態にできます。読出し/リセットコマンドを発行すると、フラッシュメモリは、ほかのコマンドが発行されるまで読出し状態を保ちます。自動アルゴリズムの実行がタイミングリミットを超過した場合は、読出し/リセットコマンドを発行するとフラッシュメモリが読出し/リセット状態へ復帰します。実際の動作については、「[3.3.1 読出し/リセット動作](#)」を参照してください。

### プログラム(書込み)コマンド

フラッシュメモリにデータを書き込むには、対象セクタに4回連続して書込みコマンドを発行し、自動アルゴリズムを起動してください。データの書込みはどのようなアドレスの順番でも、セクタの境界を越えても行えます。CPUプログラミングモードでは、ハーフワードで書込みを行います。4回目のコマンド発行が終了すると、自動アルゴリズムが起動し、フラッシュメモリへの自動書込みが開始されます。自動書込みアルゴリズムコマンドシーケンス実行後は、外部からフラッシュメモリを制御する必要はありません。実際の動作については、「[3.3.2 書込み動作](#)」を参照してください。

#### <注意事項>

- 4回目の書込みコマンド(書込みデータサイクル)を奇数番地に発行すると、コマンドの認識が正しく行われません。必ず偶数番地に書き込んでください。
- 1回の書込みコマンドシーケンスではハーフワードのデータ1つしか書き込めません。複数のデータを書き込みたい場合は、1データに1度書込みコマンドシーケンスを発行してください。

### フラッシュ消去コマンド

フラッシュ消去コマンドを対象セクタに6回連続して送るとフラッシュメモリ全体のデータを一括で消去できます。6回目の書込みが終了すると、自動アルゴリズムが起動しフラッシュメモリの消去動作が開始されます。

### セクタ消去コマンド

セクタ消去コマンドを対象セクタに6回連続して送るとフラッシュメモリのセクタのデータを消去できます。6回目の書込みが終了し、35 $\mu$ s経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタのデータを消去したい場合は、35 $\mu$ s(タイムアウト期間)以内に消去するセクタのアドレスへセクタ消去コマンドの6回目の書込みコードであるセクタ消去コード(0x30)を発行してください。タイムアウト期間内にセクタ消去コードを発行しなかった場合、タイムアウト期間を過ぎてから追加したセクタ消去コードが無効になることがあります。

### セクタ消去一時停止コマンド

セクタ消去中またはコマンドタイムアウト中に、セクタ消去一時停止コマンドを発行することによって、セクタ消去を一時停止できます。セクタ消去一時停止状態では消去対象ではないセクタのメモリセルについて読出しおよび書込み動作が可能です。実際の動作については、「[3.3.5 セクタ消去一時停止動作](#)」を参照してください。

#### <注意事項>

- このコマンドはセクタ消去中のみ有効です。フラッシュ消去中や書込み中にこのコマンドを発行しても無視されません。

### セクタ消去再開コマンド

セクタ消去一時停止状態から、中断していた消去動作を再開するためにはセクタ消去再開コマンドを発行してください。セクタ消去再開コマンドを発行するとセクタ消去状態に戻り消去動作が再開されます。実際の動作については、「[3.3.6 セクタ消去再開動作](#)」を参照してください。

#### <注意事項>

- このコマンドはセクタ消去一時停止中のみ有効です。セクタ消去中にこのコマンドを発行しても無視されません。

### 3.2.3 自動アルゴリズムの実行状態

ワークフラッシュメモリでは、書込みや消去を自動アルゴリズムで行うため、自動アルゴリズムが実行中かどうかをワークフラッシュ・レディビット(WFSTR:RDY)で、動作状態をハードウェアシーケンスフラグで確認できます。

#### ハードウェアシーケンスフラグ

自動アルゴリズムの状態を示すフラグです。ワークフラッシュ・レディビット(WFSTR:RDY)が"0"のときに、フラッシュメモリの任意のアドレスを読み出すと動作状態を確認できます。

ハードウェアシーケンスフラグのビット構成を [Figure 3-1](#) に示します。

Figure 3-1 ハードウェアシーケンスフラグのビット構成

ハーフワードアクセスの場合								
bit	15	14	13	12	11	10	9	8
	未定義	未定義	未定義	未定義	未定義	未定義	未定義	未定義
bit	7	6	5	4	3	2	1	0
	DPOL	TOGG	TLOV	未定義	SETI	TOGG2	未定義	未定義
バイトアクセスの場合								
bit	7	6	5	4	3	2	1	0
	DPOL	TOGG	TLOV	未定義	SETI	TOGG2	未定義	未定義

#### <注意事項>

- ワードアクセスでは読み出せません。必ず、CPU プログラミングモードのときにハーフワードかバイトアクセスで読み出してください。
- CPU ROM モードで任意のアドレスを読み出しても、ハードウェアシーケンスフラグを読み出すことはできません。
- コマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後1回目のハードウェアシーケンスフラグの読出し値は無視してください。

#### 各ビットとワークフラッシュメモリの状態

ハードウェアシーケンスフラグの各ビットの状態とフラッシュメモリ状態の対応を [Table 3-3](#) に示します。

Table 3-3 ハードウェアシーケンスフラグ状態一覧

		状態	DPOL	TOGG	TLOV	SETI	TOGG2	
実行中	自動書込み動作		反転データ (*1)	トグル	0	0	0	
	自動消去動作	フラッシュ消去	0	トグル	0	1	トグル	
		セクタ消去	タイムアウト期間	0	トグル	0	0	トグル
			消去	0	トグル	0	1	トグル
		セクタ消去一時停止	読出し (消去対象セクタ)	0	0	0	1	トグル
	読出し (消去対象外セクタ)		データ (*1)	データ (*1)	データ (*1)	データ (*1)	データ (*1)	
		自動書き込み動作 (消去対象外セクタ)	反転データ (*1)	トグル	0	1	0	
タイムリミット超過	自動書込み動作		反転データ (*1)	トグル	1	0	0	
	自動消去動作		0	トグル	1	1	トグル	

\*1: 読み出される値については、「ビット説明」を参照してください。

## ビット説明

### [bit15:8] 未定義ビット

#### [bit7] DPOL : データポーリングフラグビット

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、データポーリング機能で自動アルゴリズムが実行中かどうかをこのビットに示します。

読み出される値は動作状態によって異なります。

##### ■ 書込み時

###### - 書込み中:

書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

###### - 書込み終了後:

ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit7 の値が読み出されます。

##### ■ セクタ消去時

###### - セクタ消去実行中:

フラッシュメモリの全領域から "0" が読み出されます。

###### - セクタ消去後:

必ず "1" が読み出されます。

###### - フラッシュ消去時

- ・ フラッシュ消去実行中 : 必ず "0" が読み出されます。
- ・ フラッシュ消去後 : 必ず "1" が読み出されます。

###### - セクタ消去一時停止時

- ・ セクタ消去指定したセクタのアドレスを指定して読み出した場合: "0" が読み出されます。
- ・ セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合: 指定したアドレスの bit7 の値が読み出されません。

###### - 書込み中の場合:

書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。

ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

#### <注意事項>

- 自動アルゴリズムを起動しているときは、指定したアドレスのデータを読み出すことはできません。このビットで自動アルゴリズムの動作が終了していることを確認してから、データを読み出してください。

**[bit6] TOGG : トグルフラグビット**

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムが実行中かどうかをこのビットで示します。読み出される値は動作状態によって異なります。

**■ 書込み/セクタ消去/フラッシュ消去時**

## - 書込み/セクタ消去/フラッシュ消去中:

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます（トグル動作）。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

## - 書込み/セクタ消去/フラッシュ消去終了後:

ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit6 の値が読み出されます。

**■ セクタ消去一時停止時**

## - セクタ消去指定したセクタのアドレスを指定して読み出した場合:

“0”が読み出されます。

## - セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit6 の値が読み出されます。

## - 書込み中の場合:

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます（トグル動作）。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。

**[bit5] TLOV : タイミングリミット超過フラグビット**

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムの実行時間がフラッシュメモリ内部で規定している時間 (内部パルスの回数) を超過したかどうかをこのビットで示します。読み出される値は動作状態によって異なります。

**■ 書込み/セクタ消去/フラッシュ消去時**

次の値が読み出されます。

0 : 規定時間内

1 : 規定時間を超過している

このビットが“1”のときに、DPOL ビットや TOGG ビットが自動アルゴリズム実行中であることを示していると、書込みや消去に失敗したことになります。

例えば、フラッシュメモリでは“0”が書き込まれているデータを“1”に書き換えることができません。このため、“0”が書き込まれているアドレスに“1”を書き込もうとすると、フラッシュメモリがロックされ自動アルゴリズムが終了しません。この場合は、DPOL ビットの値は無効のままになり、TOGG ビットからは“1”と“0”が交互に読み出され続けます。

この状態のまま規定時間を越えたときに、このビットが“1”に変わります。このビットが“1”になった場合はリセットコマンドを発行してください。

**■ セクタ消去一時停止時**

## - セクタ消去指定したセクタのアドレスを指定して読み出した場合:

“0”が読み出されます。

## - セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit5 の値が読み出されます。

## - 書込み中の場合:

次の値が読み出されます。

0 : 規定時間内

1 : 規定時間を超過している

このビットが“1”のときに、DPOL ビットや TOGG ビットが自動アルゴリズム実行中であることを示していると、書込みや消去に失敗したことになります。

**<注意事項>**

- このビットが“1”の場合は、フラッシュメモリが正しく使用されなかったことを示しています。フラッシュメモリの不良ではありません。リセットコマンドを発行してから適切な処理を行ってください。

**[bit4] 未定義ビット**

**[bit3] SET1 : セクタ消去タイムフラグビット**

セクタ消去時は、セクタ消去コマンドを発行してから実際にセクタ消去が開始されるまでには、35 $\mu$ sのタイムアウト期間が必要です。

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、セクタ消去コマンドのタイムアウト期間中かどうかをこのビットで示します。

読み出される値は動作状態によって異なります。

**■ セクタ消去時:**

セクタ消去時に次のセクタ消去コードを入力する前に、このビットを確認することで次のセクタ消去コードが受け付けられる状態かどうかを確認できます。

ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスせず、次の値が読み出されます。

0 : セクタ消去タイムアウト期間中

次のセクタ消去コード(0x30)を受け付けられます。

1 : セクタ消去タイムアウト期間を超過している

このときに、DPOL ビットやTOGG ビットが自動アルゴリズム実行中であることを示している、フラッシュメモリ内部の消去が開始されています。この場合、セクタ消去一時停止 (0xB0)以外のコマンドはフラッシュメモリ内部の消去が完了するまで無視されます。

**■ セクタ消去一時停止時**

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:

“1”が読み出されます。

- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit3 の値が読み出されます。

- 書込み中の場合:

“1”が読み出されます。

**[bit2] TOGG2 : トグルフラグビット**

セクタ消去一時停止状態において、消去対象以外のセクタからは値を読み出せますが、消去対象セクタからは読み出せません。TOGG2は、セクタ消去一時停止中、読み出したデータのトグル動作の有無により、該当セクタが消去対象であるかどうかを検出します。

**■ 書込み時**

“0”が読み出されます。

**■ セクタ消去/フラッシュ消去時**

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます(トグル動作)。

**■ セクタ消去一時停止時**

- セクタ消去指定したセクタのアドレスを指定して読み出した場合:

連続でこのビットを読み出すと、“1”と“0”が交互に読み出されます(トグル動作)。

- セクタ消去指定したセクタ以外のアドレスを指定して読み出した場合:

指定したアドレスの bit2 の値が読み出されます。

**[bit1:0] 未定義ビット**



### 3.3 ワークフラッシュメモリの動作説明

ワークフラッシュメモリの動作について、コマンドごとに説明します。

#### 3.3.1 読出し/リセット動作

#### 3.3.2 書込み動作

#### 3.3.3 フラッシュ消去動作

#### 3.3.4 セクタ消去動作

#### 3.3.5 セクタ消去一時停止動作

#### 3.3.6 セクタ消去再開動作

#### 3.3.1 読出し/リセット動作

読出し/リセット動作を説明します。

読出し/リセットコマンドを対象セクタに連続して送るとフラッシュメモリを読出し/リセット状態にできます。

この状態は、フラッシュメモリの初期状態のため、電源を投入したときやコマンドが正常終了すると、フラッシュメモリは常に読出し/リセット状態に戻ります。電源投入時はデータ読出しコマンドを発行する必要はありません。また、読出し/リセット状態では通常の読出しアクセスでデータを読み出したり、CPU からプログラムアクセスしたりできます。このため、データを読み出すときに読出し/リセットコマンドを発行する必要はありません。

#### 3.3.2 書込み動作

書込み動作を説明します。

次の順番で書込みを行います。

##### 1. 書込みコマンドを対象セクタに連続して発行する

自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。

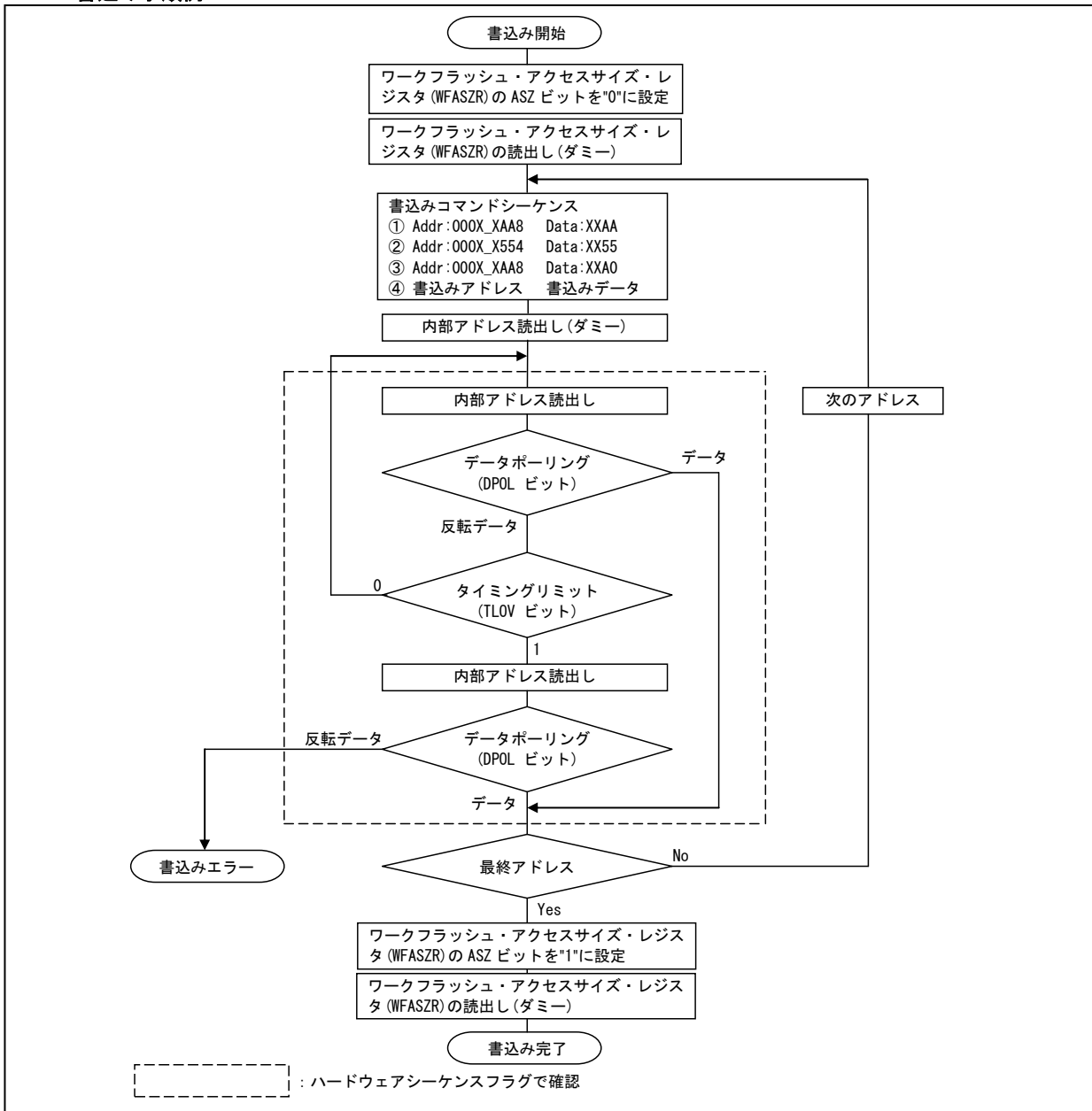
書込みコマンド発行後は外部からフラッシュメモリを制御する必要はありません。

##### 2. 書込みを行ったアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が書き込んだ値と一致していると、フラッシュメモリへの書込みが終了したことになります。書込みが終了していない場合は、書込みコマンドシーケンスで最後に書き込まれたデータ (PD) の bit7 の値と逆の値 (反転データ) が読み出されます。

フラッシュメモリへの書込み動作例を [Figure 3-2](#) に示します。

Figure 3-2 書き込み手順例


**<注意事項>**

- 書き込みコマンドについては、「3.2 自動アルゴリズム」を参照してください。
- ハードウェアシーケンスフラグの DPOL ビットは、TLOV ビットと同時に値が変わるため TLOV ビットが"1"の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG ビット、TLOV ビットが"1"になると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG ビットを再度確認する必要があります。
- フラッシュメモリへは、どのようなアドレスの順番でも、またセクタの境界を越えても書き込めます。しかし 1 回の書き込みコマンドシーケンスではハーフワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、1 データに 1 度書き込みコマンドシーケンスを発行してください。
- 書き込み動作中はフラッシュメモリへ発行されたすべてのコマンドが無視されます。
- 書き込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。

### 3.3.3 フラッシュ消去動作

フラッシュ消去動作を説明します。

フラッシュメモリの全セクタのデータを一括して消去できます。これをフラッシュ消去とよびます。

フラッシュ消去コマンドを対象セクタに連続して送ると自動アルゴリズムを起動して、全セクタのデータを一括で消去できます。

フラッシュ消去コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。

1. フラッシュ消去コマンドを対象セクタに連続して発行する

自動アルゴリズムが起動されフラッシュメモリのフラッシュ消去動作が開始されます。

2. 任意のアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が "1"だと、フラッシュ消去が終了したことになります。

フラッシュ消去に必要な時間は「セクタ消去時間×全セクタ数+フラッシュ書込み時間 (プリプログラム)」となります。フラッシュ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

### 3.3.4 セクタ消去動作

セクタ消去動作を説明します。

フラッシュメモリ内のセクタを選択して、選択したセクタのデータのみを消去できます。複数のセクタも同時に指定できます。

次の順番でセクタ消去を行います。

1. セクタ消去コマンドを対象セクタに連続して発行する

35  $\mu$ s 経過 (タイムアウト期間) すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。

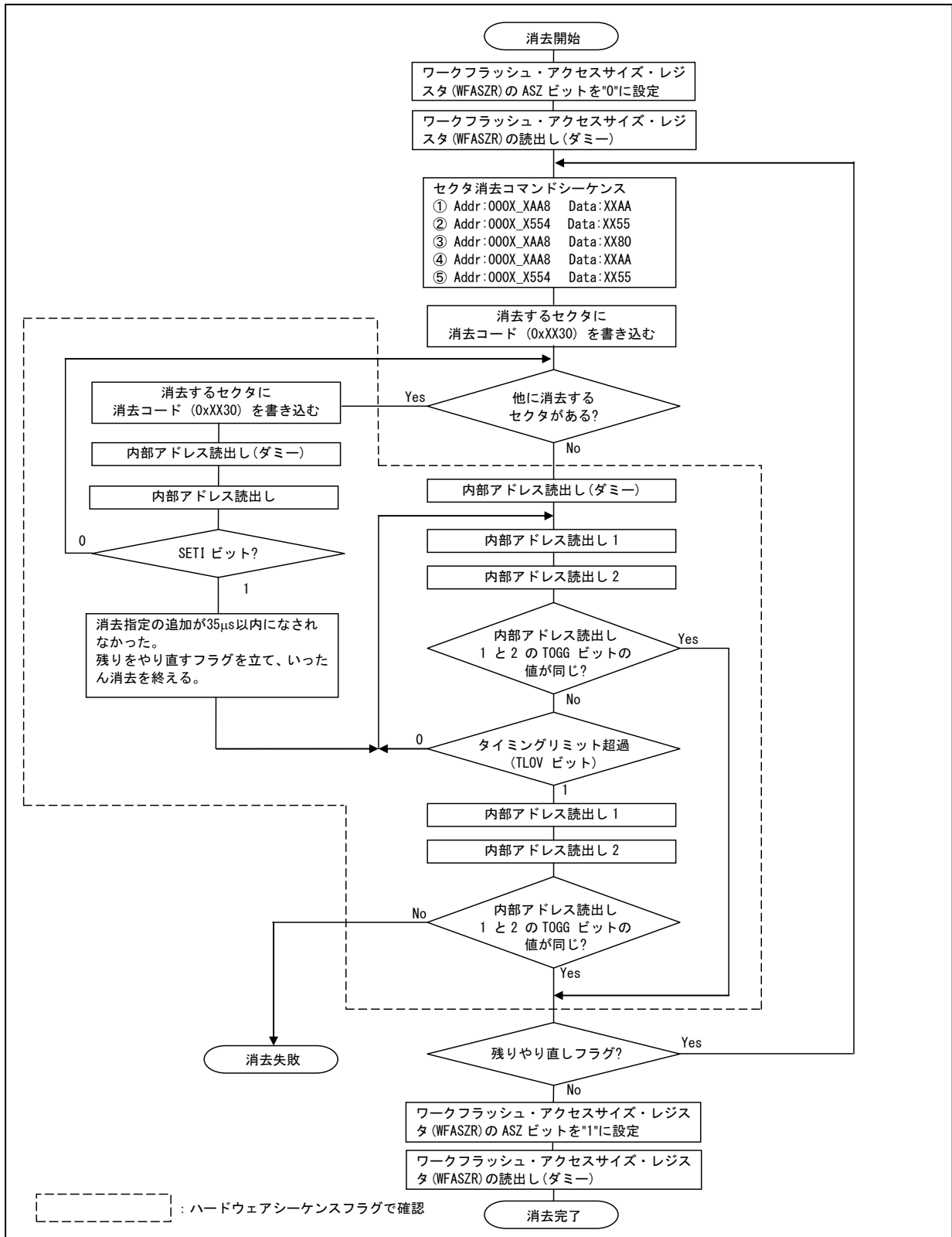
複数のセクタを消去したい場合は、35  $\mu$ s (タイムアウト期間) 以内に消去するセクタのアドレスに消去コード (0x30) を発行してください。タイムアウト期間経過後に発行しても、追加したセクタ消去コードが無効になることがあります。

2. 任意のアドレスにリードアクセスする

読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOL ビット) が "1"だと、セクタ消去が終了したことになります。

また、TOGG ビットを利用してセクタ消去が完了したかどうか確認できます。確認動作に TOGG ビットを使用した場合を例にとって、セクタ消去手順例を [Figure 3-3](#) に示します。

Figure 3-3 セクタ消去手順例



セクタ消去に必要な時間は「(セクタ消去時間+セクタ書込み時間 (プリプログラム))×セクタ数」となります。  
セクタ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

**<注意事項>**

- セクタ消去コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。
- ハードウェアシーケンスフラグの DPOL ビットは、TLOV ビットと同時に値が変わるため TLOV ビットが"1"の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG ビットは、TLOV ビットが"1"になると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG ビットを再度確認する必要があります。
- タイムアウト期間を含むセクタ消去中に、セクタ消去コマンド/消去一時停止コマンド以外のコマンドを発行すると、無視されます。

### 3.3.5 セクタ消去一時停止動作

セクタ消去一時停止動作を説明します。

セクタ消去中またはコマンドタイムアウト状態からセクタ消去一時停止コマンドを送付すると、セクタ消去一時停止状態に遷移し、消去動作を一時的に中断します。

消去再開コマンドを送付することによってセクタ消去状態に戻り、中断していた消去動作を再開できます。ただし、コマンドタイムアウト状態からセクタ消去一時停止状態に遷移した場合であっても、消去再開コマンドが正常に書き込まれた場合は、コマンドタイムアウト状態には遷移せず、セクタ消去状態に遷移して直ちにセクタ消去動作を再開します。

#### セクタ消去一時停止動作

次の順番でセクタ消去が一時停止されます。

1. コマンドタイムアウト期間～セクタ消去中に、セクタ消去一時停止コマンドをフラッシュメモリのアドレス範囲内の任意のアドレスへライトします。
2. コマンドタイムアウト期間中にセクタ消去一時停止コマンドを発行した場合は、直ちにタイムアウトを終了し、消去動作を一時停止します。セクタ消去中にセクタ消去一時停止コマンドを発行した場合、実際に消去を停止するまでに最大で 35 $\mu$ s の時間がかかります。

#### <注意事項>

- セクタ消去一時停止コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。
- セクタ消去を一時停止できるのは、セクタ消去のコマンドタイムアウト期間～セクタ消去中のみです。フラッシュ消去は一時停止できません。また、セクタ消去一時停止中に再度セクタ消去一時停止コマンドを発行しても無視されます。

#### セクタ消去一時停止後の状態

セクタ消去一時停止後に消去対象のセクタを読み出した場合、ハードウェアシーケンスフラグが読み出されます。また、消去対象でないセクタを読み出した場合は、メモリセルのデータが読み出されます。

#### <注意事項>

- セクタ消去一時停止状態で、新たな消去コマンドは無視されます。

### 3.3.6 セクタ消去再開動作

セクタ消去一時停止中にセクタ消去を再開する動作を説明します。

セクタ消去を一時停止中に、任意のアドレスにセクタ消去再開コマンドを発行するとセクタ消去を再開できます。

セクタ消去再開コマンドが発行されると、セクタ消去一時停止中のセクタ消去動作が再開されます。

セクタ消去再開コマンドについては、「[3.2 自動アルゴリズム](#)」を参照してください。

#### <注意事項>

- セクタ消去再開コマンドは、セクタ消去一時停止中のみ有効です。セクタ消去中にセクタ消去再開コマンドを発行しても無視されます。
- セクタ消去再開コマンド発行後、セクタ消去動作が再開されるまで2ms 以上かかります。そのため、この時間に満たない間隔で消去再開と消去停止を繰り返すと、消去動作が進まないままタイミングリミット超過になります。セクタ消去再開コマンド発行後に再度セクタ消去一時停止コマンドを発行する場合は、セクタ消去再開コマンドの発行後から2ms 以上の間隔を空けてください。

### 3.4 ワークフラッシュメモリの使用上の注意

ワークフラッシュメモリの使用上の注意について説明します。

- 書き込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。  
また、書き込み・消去中には、ウォッチドッグタイマなどの予期しないリセットがかからないようにする必要があります。
- ワークフラッシュ・アクセスサイズ・レジスタ(WFASZR)の ASZ ビットで CPU プログラミングモードを設定(ASZ=0)したときは、フラッシュメモリ上のプログラムを実行しないでください。正常な値を取り出せずにプログラムが暴走します。
- ワークフラッシュ・アクセスサイズ・レジスタ(WFASZR)の ASZ ビットで CPU プログラミングモードを設定(ASZ=0)し、フラッシュメモリ上に割込みベクタテーブルがある場合は、割込み要求を発生させないでください。正常な値を取り出せずにプログラムが暴走します。
- ワークフラッシュ・アクセスサイズ・レジスタ(WFASZR)の ASZ ビットで CPU プログラミングモードを設定(ASZ=0)した場合、低消費電力モードには遷移しないでください。
- ワークフラッシュ・アクセスサイズ・レジスタ(WFASZR)の ASZ ビットで CPU ROM モードを設定(ASZ=1)した場合は、フラッシュメモリに書き込みを行わないでください。
- ワークフラッシュ・アクセスサイズ・レジスタ(WFASZR)の ASZ ビットで CPU プログラミングモードを設定(ASZ=0)した場合、フラッシュメモリへの書き込みは必ずハーフワードで行ってください。バイト書き込みはしないでください。
- フラッシュメモリに対して自動アルゴリズムのコマンド発行直後は、必ずダミーの読出しをしてから実際に読み出したいデータを読み出してください。自動アルゴリズムコマンド発行直後にデータを読み出しても、読出し値は保証できません。
- 本デバイスを低消費電力モードへ遷移させる場合は、フラッシュメモリ自動アルゴリズムの動作が終了していることを必ず確認してください。  
低消費電力モードについては、『FM4 ファミリー ペリフェラルマニュアル』の『低消費電力モード』の章を参照してください。



## 4. レジスタ

レジスタについて説明します。

### レジスタ一覧

レジスタ略称	レジスタ名	参照先
WFASZR	ワークフラッシュ・アクセスサイズ・レジスタ	4.1
WFRWTR	ワークフラッシュ・リードウェイト・レジスタ	4.2
WFSTR	ワークフラッシュ・ステータス・レジスタ	4.3

## 4.1 WFASZR (WorkFlash Access Size Register) ワークフラッシュ・アクセスサイズ・レジスタ

WFASZR について説明します。

フラッシュメモリに対するアクセスサイズを設定します。リセット解除後は ASZ="1"(32 ビットリード)になり、CPU ROM モードに設定されます。CPU プログラミングモードに設定する場合には、ASZ="0"に設定してください。

bit	7	6	5	4	3	2	1	0
Field	予約							ASZ
属性								RW
初期値								1

### [bit7:2] 予約ビット

読出し値は不定です。書き込み時は無視されます。

### [bit1:0] ASZ: Access Size

フラッシュメモリのアクセスサイズを指定します。

名称	bit	説明
ASZ	0	WorkFlash Access Size 0: 16 ビットリード/ライト(CPU プログラミングモード) 1: 32 ビットリード (CPU ROM モード: 初期値)

#### <注意事項>

- ASZ="0"の設定時、フラッシュに対しての書き込みは必ずハーフワードアクセス(16 ビットアクセス)を行ってください。
- 本レジスタを変更する場合にはフラッシュメモリに書き込まれた命令によって行わないでください。フラッシュ領域以外のプログラムにて書き換えてください。
- 本レジスタを変更後、必ずレジスタに対してダミーリードを行ってください。

## 4.2 WFRWTR (WorkFlash Read Wait Register) ワークフラッシュ・リード ウェイト・レジスタ

WFRWTR について説明します。

ASZ="1"(32 ビットリードモード)のときに有効なレジスタです。  
フラッシュメモリに対するアクセス方法を設定します。

bit	7	6	5	4	3	2	1	0
Field	予約					RWT		
属性						RW	RW	RW
初期値						1	X	X

### [bit7:3] 予約ビット

読出し値は不定です。書込み時は無視されます。

### [bit2:0] RWT: Read Wait Cycle

フラッシュメモリのアクセス方法を指定します。

名称	bit	説明
RWT	2:0	Read Wait Cycle 000: 0 サイクルウェイトモード HCLK が 40MHz 以下の場合に設定可能です。 001: 2 サイクルウェイトモード HCLK が 40MHz を超え、72MHz 以下の場合に設定可能です。 01x: 4 サイクルウェイトモード HCLK が 72MHz を超え、120MHz 以下の場合に設定可能です。 1xx: 6 サイクルウェイトモード(初期値) HCLK が 120MHz を超える場合には必ず設定してください。

#### <注意事項>

- HCLK が 40MHz を超える場合は RWT=000 (0 サイクルウェイトモード)での使用は禁止です。RWT=000 の間、HCLK が一瞬でも 40MHz を超えないようにしてください。
- HCLK が 72MHz を超える場合は RWT=000 (0 サイクルウェイトモード)、および RWT=001 (2 サイクルウェイトモード)での使用は禁止です。RWT=000、または RWT=001 の間、HCLK が一瞬でも 72MHz を超えないようにしてください。
- HCLK が 120MHz を超える場合は RWT=000 (0 サイクルウェイトモード)、RWT=001 (2 サイクルウェイトモード)、および RWT=01x (4 サイクルウェイトモード)での使用は禁止です。RWT=000、RWT=001、または RWT=01x の間、HCLK が一瞬でも 120MHz を超えないようにしてください。
- 本レジスタを変更後、必ずレジスタに対してダミーリードを行ってください。

### 4.3 WFSTR (WorkFlash Status Register) ワークフラッシュ・ステータス・レジスタ

WFSTR について説明します。

フラッシュメモリのステータスを示すレジスタです。

bit	7	6	5	4	3	2	1	0
Field	予約						HNG	RDY
属性							R	R
初期値							0	X

#### [bit7:2] 予約ビット

読出し値は不定です。書込み時は無視されます。

#### [bit1] HNG : WorkFlash Hang

フラッシュメモリの HANG 状態を示します。タイミング超過(「[bit5] TLOV : タイミングリミット超過フラグビット」参照)すると HANG 状態になります。本ビットが"1"になった場合はリセットコマンド(「3.2.1 コマンドシーケンス」参照)を発行してください。

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後 1 回目の本ビット読出し値は無視してください。

名称	bit	説明
HNG	1	WorkFlash Hang 0: フラッシュメモリの HANG 状態を検出していません。 1: フラッシュメモリの HANG 状態を検出しました。

#### [bit0] RDY : WorkFlash Rdy

自動アルゴリズムでフラッシュメモリの書込み/消去動作が実行中か完了しているかを示します。動作中の場合、フラッシュメモリへデータを書き込んだりデータを消去したりできません。

名称	bit	説明
RDY	0	WorkFlash Rdy 0: 動作中(書込み/消去不可) 1: 動作完了(書込み/消去可能)

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるため、コマンド発行後 1 回目の本ビット読出し値は無視してください。

## CHAPTER 3: フラッシュセキュリティ



フラッシュセキュリティ機能によりメインフラッシュメモリとワークフラッシュメモリのデータを保護できます。フラッシュセキュリティの概要、動作について説明します。

1. 概要
2. 動作説明

## 1. 概要

フラッシュセキュリティの概要を説明します。

メインフラッシュメモリのセキュリティコード領域に保護コード"0x0001"を書き込むと、メインフラッシュメモリとワークフラッシュメモリへのアクセスを制限できます。2つのフラッシュメモリを一度保護すると、両方のフラッシュメモリの消去を行うまで保護された状態は解除できません。保護された状態を解除しない限り、外部端子からメインフラッシュメモリ、およびワークフラッシュメモリのデータを読み出し/書込みをすることはできません。

この機能はフラッシュメモリに格納される自己完結型プログラムやデータのセキュリティを必要とするアプリケーションに適しています。

Table 1-1 にセキュリティコードのアドレスと保護コードを示します。

Table 1-1 セキュリティコードのアドレスと保護コード

アドレス	保護コード
0x0040_0000	0x0001

### <注意事項>

- メインフラッシュメモリのセキュリティコード領域に保護コードを書き込むと、メインフラッシュメモリとワークフラッシュメモリの両方に対して同時にフラッシュセキュリティが有効になります。

## 2. 動作説明

フラッシュセキュリティの動作を説明します。

### セキュリティのかけ方

セキュリティコードのアドレスに保護コード"0x0001"を書き込みます。すべてのリセット要因または電源再投入後に有効となりセキュリティがかかります。

### セキュリティの解除

1. ワークフラッシュに対してフラッシュ消去コマンドを発行します。
2. ワークフラッシュのフラッシュ消去の完了を確認します。
3. セキュリティコードが格納されているメインフラッシュに対してフラッシュ消去コマンドを発行します。
4. フラッシュ消去実行後のすべてのリセット要因または電源再投入で、セキュリティは解除されます。

### セキュリティ有効時の動作

セキュリティ有効時の動作は、各モードにより動作が異なります。

Table 2-1 に各モードにおけるセキュリティ動作について示します。

Table 2-1 セキュリティ有効時のフラッシュ動作

モード	モード端子 MD[1:0]	フラッシュへのアクセス			JTAG 端子からの アクセス
		フラッシュ 消去	その他の コマンド	読出し	
ユーザモード	"00"	可能	可能	有効データ	不可能
シリアルライターモード	"01"	可能	不可能	無効データ	不可能

#### <注意事項>

- 保護コードは、フラッシュプログラミングの最後に書き込むことを推奨します。これは、プログラミング中、不意に保護されてしまうことを防ぐためです。
- ユーザモード時は、セキュリティ有効時でもフラッシュメモリに対して制限はありません。しかし、セキュリティ有効時はJTAG端子から内部へアクセスできないようにJTAG端子は固定されます。セキュリティ解除はJTAG端子を通して行えないため、解除を行う場合はシリアルライターを使用してフラッシュ消去を行ってください。
- セキュリティをかけた状態でのフラッシュメモリ障害解析は一切行えません。
- セキュリティを解除する時は、必ずワークフラッシュのデータから消去してください。ワークフラッシュの消去完了前にメインフラッシュを消去することはできません。

# CHAPTER 4: シリアル書込み接続



本シリーズは、フラッシュメモリのシリアルオンボード書込み(Cypress 標準)に対応しています。本章では、フラッシュメモリのシリアル書込みについて、Cypress 製シリアルプログラマを用いた場合の基本構成を説明します。

## 1. シリアルプログラマ



## 1. シリアルプログラマ

Cypress 製シリアルプログラマ(ソフトウェア)は、フラッシュメモリ内蔵マイコン全製品を対象にしたオンボードプログラミングツールです。

本シリアルプログラマは、PC との I/F(RS-232C, USB)により、2 種類のツールを用意していますので、環境に合わせてご利用いただけます。

なお、USB ファンクションを搭載している品種は USB-シリアル変換をすることなくパソコンとマイコンをダイレクトに接続してオンボード書込みができます。

### 1.1 基本構成

### 1.2 使用端子

## 1.1 基本構成

基本構成について説明します。

### FLASH MCU Programmer(クロック非同期シリアル書き込み)の基本構成

FLASH MCU Programmer は、パソコンとユーザシステムを RS-232C ケーブルで接続することで、クロック非同期シリアル通信によりユーザシステムに実装されているマイコンに内蔵されているフラッシュメモリへデータを書き込みます。本シリーズでは水晶振動子、または内蔵高速 CR 発振器を使用してシリアル書き込み(UART 通信モード)することが可能です。

なお、外部に水晶振動子を接続せずに内蔵 CR で動作している状態でフラッシュセキュリティ機能が有効になっているフラッシュメモリの消去を行うと、消去後に通信ができなくなり内蔵 CR トリミングデータが消失します。フラッシュセキュリティ機能が有効になっている場合には、外部に水晶振動子を接続してください。

FLASH MCU Programmer の基本構成を Figure 1-1、システム構成を Table 1-1 に示します。

Figure 1-1 FLASH MCU Programmer の基本構成

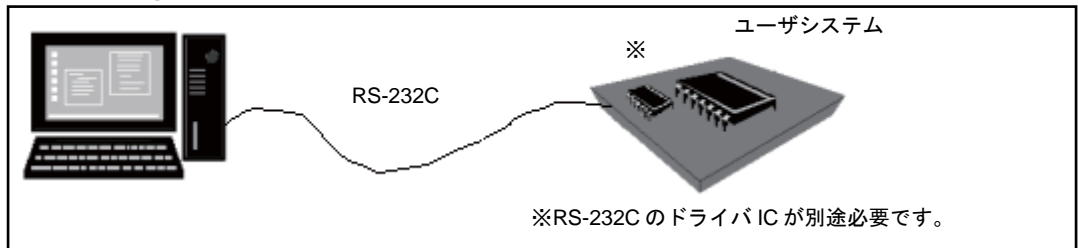


Table 1-1 FLASH MCU Programmer のシステム構成

名称	仕様
FLASH MCU Programmer	ソフトウェア (入手方法は弊社営業部門までお問い合わせください。)
RS-232C ケーブル	市販品

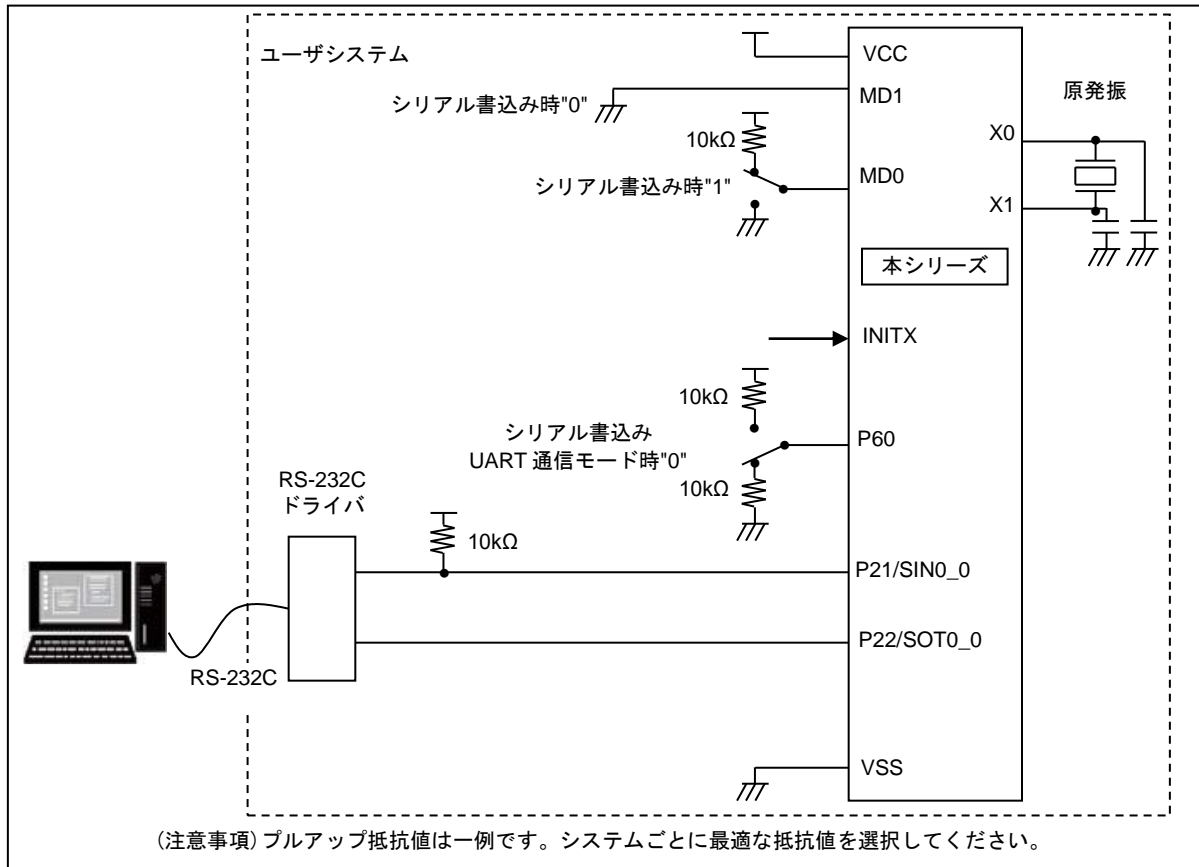
## RS-232C I/F の接続例

RS-232C I/F の接続例について説明します。

### ■ 原発振に水晶振動子を使用する場合

原発振に水晶振動子を使用する場合の RS-232C I/F の接続例を Figure 1-2 に示します。水晶振動子を使用した場合の通信ボーレートは 115200[bps]になります。入力可能な周波数と対応する通信開始時のボーレートを Table 1-2 に示します。

Figure 1-2 水晶振動子を使用する場合の接続例



■ 原発振に外部クロックを使用する場合

原発振に外部クロックを使用する場合の RS-232C I/F の接続例を Figure 1-3 に示します。外部クロックを使用した場合の通信ボーレートは 115200[bps]になります。入力可能な周波数と対応する通信開始時のボーレートを Table 1-2 に示します。

外部クロックを使用する場合、以下の制限があります。

- ユーザシステム上 X1 端子 GPIO として使用するため開放できない場合、発振が不安定となる可能性があります。X1 端子を開放できない場合は、内蔵高速 CR 発振器を使用して通信を行ってください。

Figure 1-3 外部クロックを使用する場合の接続例

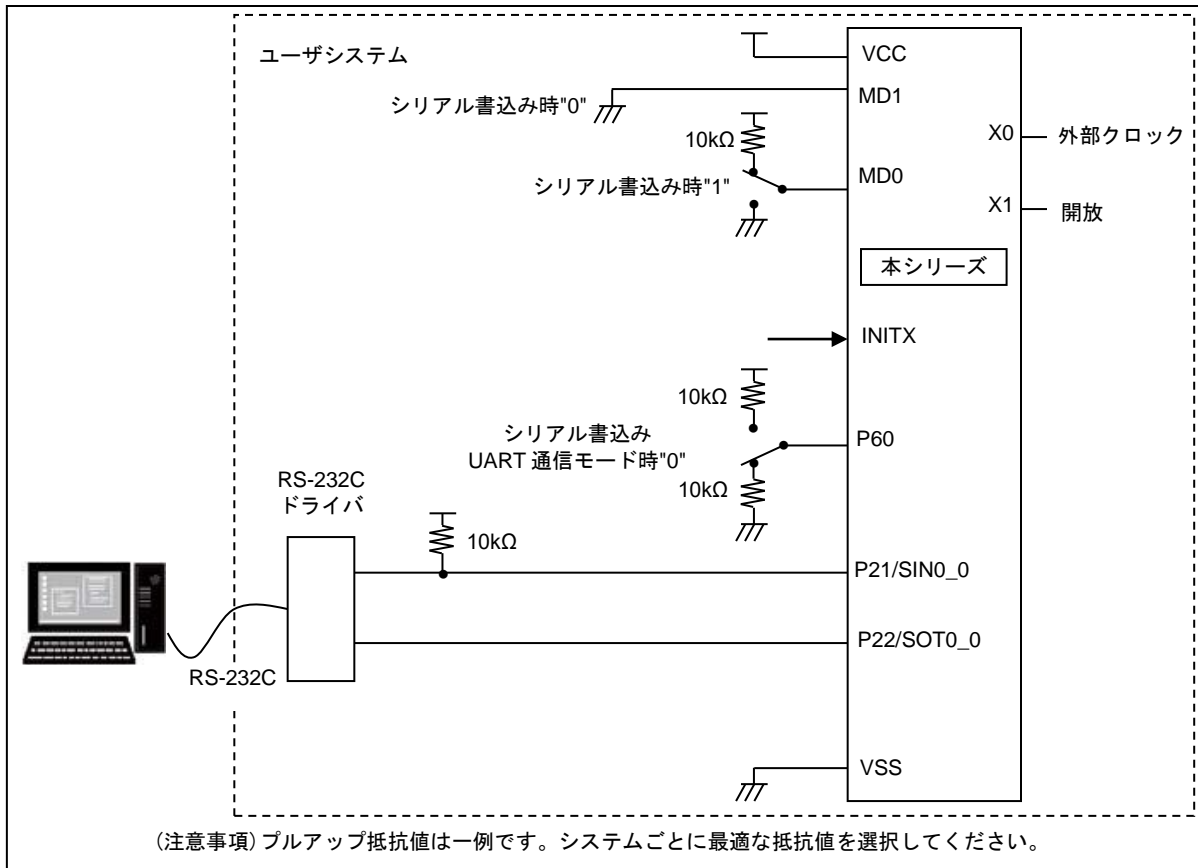


Table 1-2 クロック非同期シリアル通信時の入力可能な発振周波数と通信ボーレート

原発振周波数	通信ボーレート
4MHz	9600bps
8MHz	19200bps
16MHz	38400bps
24MHz	57600bps
48MHz	115200bps

■ 原発振に内蔵高速 CR 発振器を使用する場合

原発振に内蔵高速 CR 発振器を使用する場合の RS-232C I/F の接続例を Figure 1-4 に示します。

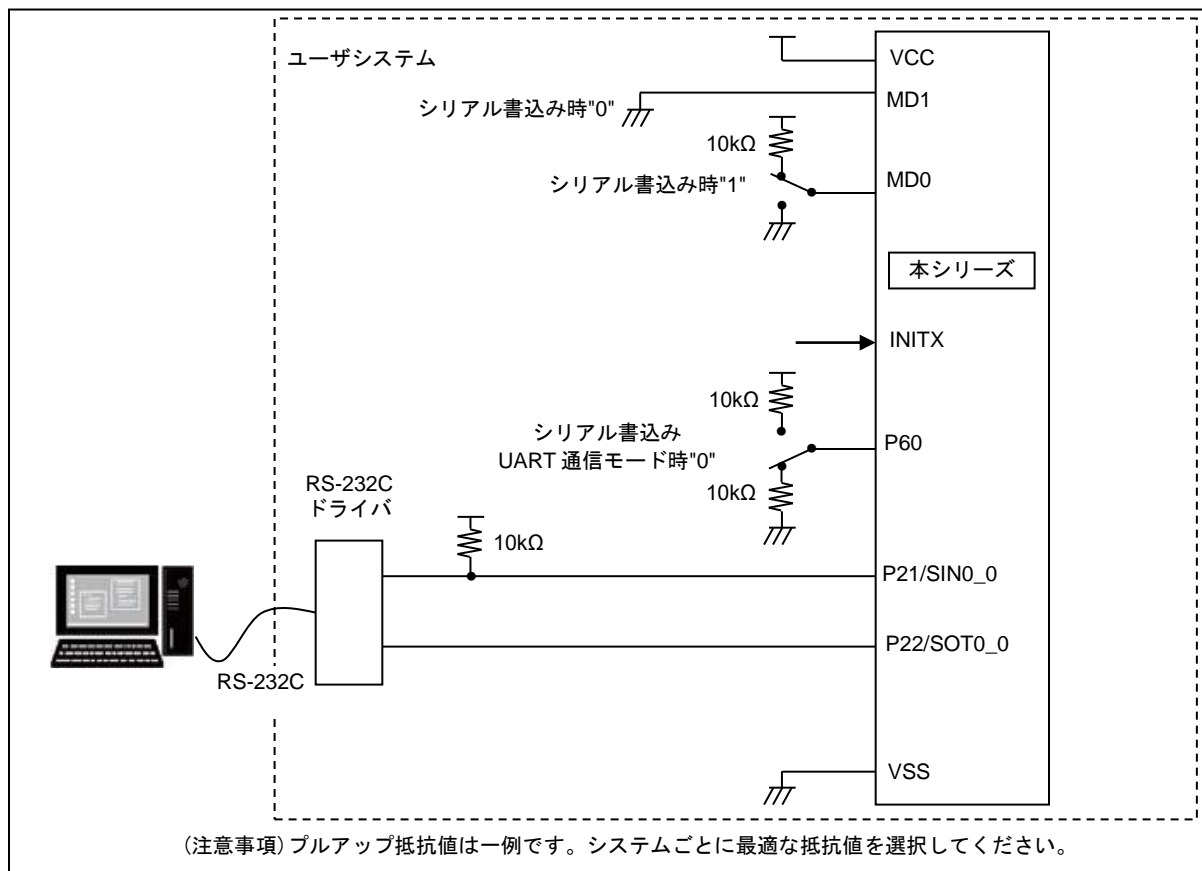
X0/X1 端子に水晶振動子、または外部クロックを接続しない場合に内蔵高速 CR 発振器を使用して通信を行います。

内蔵高速 CR 発振器を使用した場合の通信ボーレートは 115200[bps]、通信開始時のボーレートは 9600[bps]になります。

また、内蔵高速 CR 発振器を使用する場合、以下の制限があります。

- 内蔵高速 CR 発振器の発振周波数は温度/電圧により変動するため、許容ボーレート誤差範囲を超える場合があります。
- 内蔵高速 CR 発振器を使用する場合は、ご使用する製品の『データシート』の『内蔵 CR 発振規格』を確認してください。
- 外部に水晶振動子を接続せずに内蔵 CR で動作している状態でフラッシュセキュリティ機能が有効になっているフラッシュメモリの消去を行うと、消去後に通信ができなくなり内蔵 CR トリミングデータが消失します。フラッシュセキュリティ機能が有効になっている場合には、外部に水晶振動子を接続してください。

Figure 1-4 内蔵高速 CR 発振器を使用する場合の接続例



### FLASH USB DIRECT Programmer(USB シリアル書き込み)の基本構成

FLASH USB DIRECT Programmer は、パソコンとユーザシステムを USB ケーブルで接続することで、USB 通信モードによりユーザシステムに実装されているマイコンに内蔵されているフラッシュメモリへデータを書き込みます。

FLASH USB DIRECT Programmer の基本構成を Figure 1-5、システム構成を Table 1-3 に示します。

Figure 1-5 FLASH USB DIRECT Programmer の基本構成

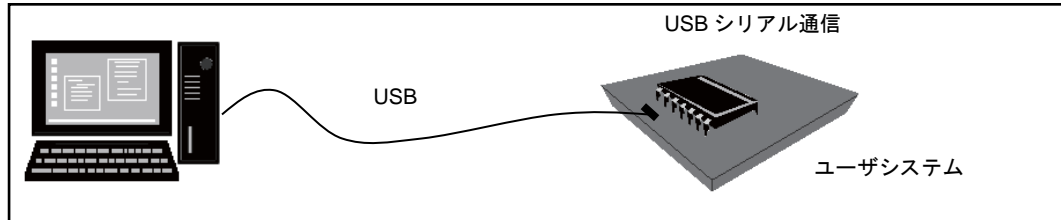


Table 1-3 FLASH USB DIRECT Programmer のシステム構成

名称	仕様
FLASH USB DIRECT Programmer	ソフトウェア (入手方法は弊社営業部門までお問い合わせください。)
USB ケーブル	市販品

接続例については FLASH USB DIRECT Programmer のマニュアル(HELP)を確認してください。

Figure 1-6 USB I/F の接続例(自己電源使用時)

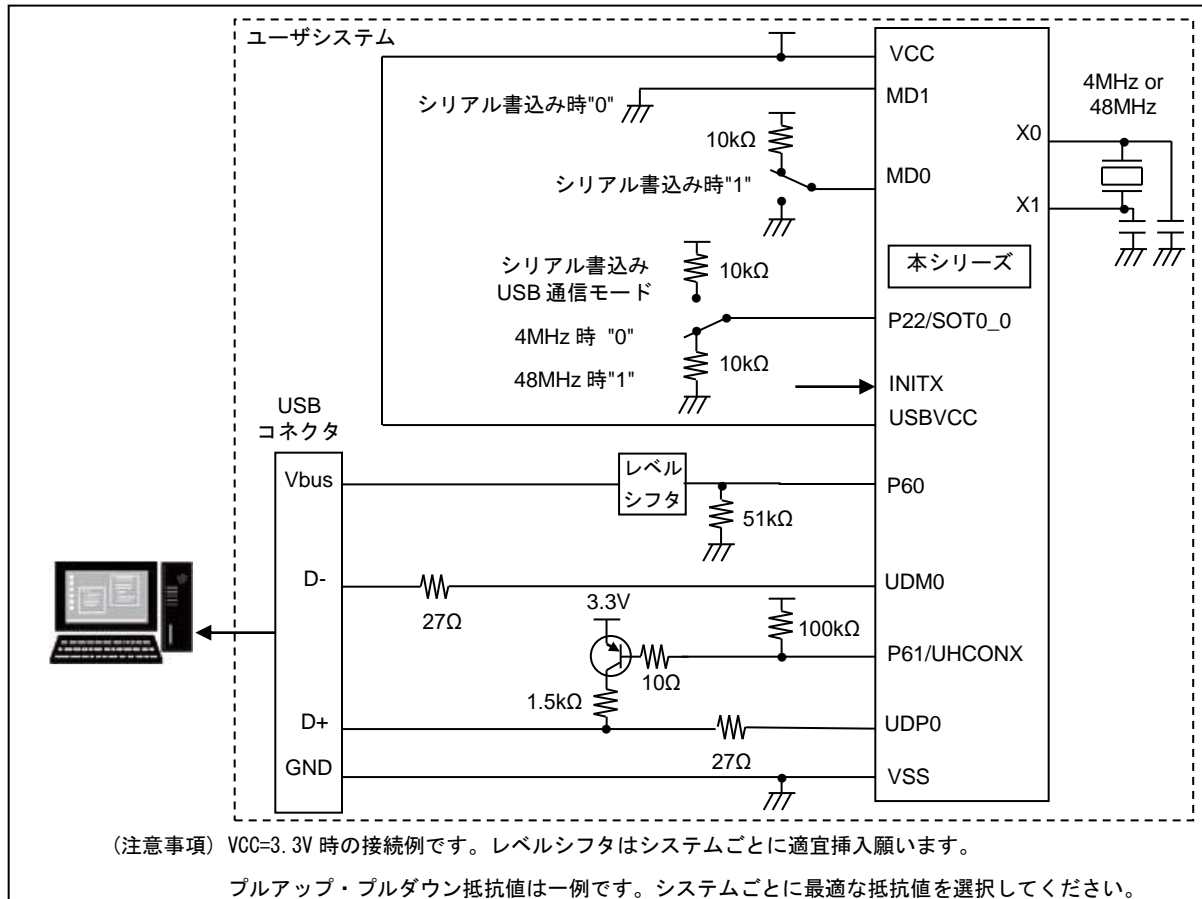
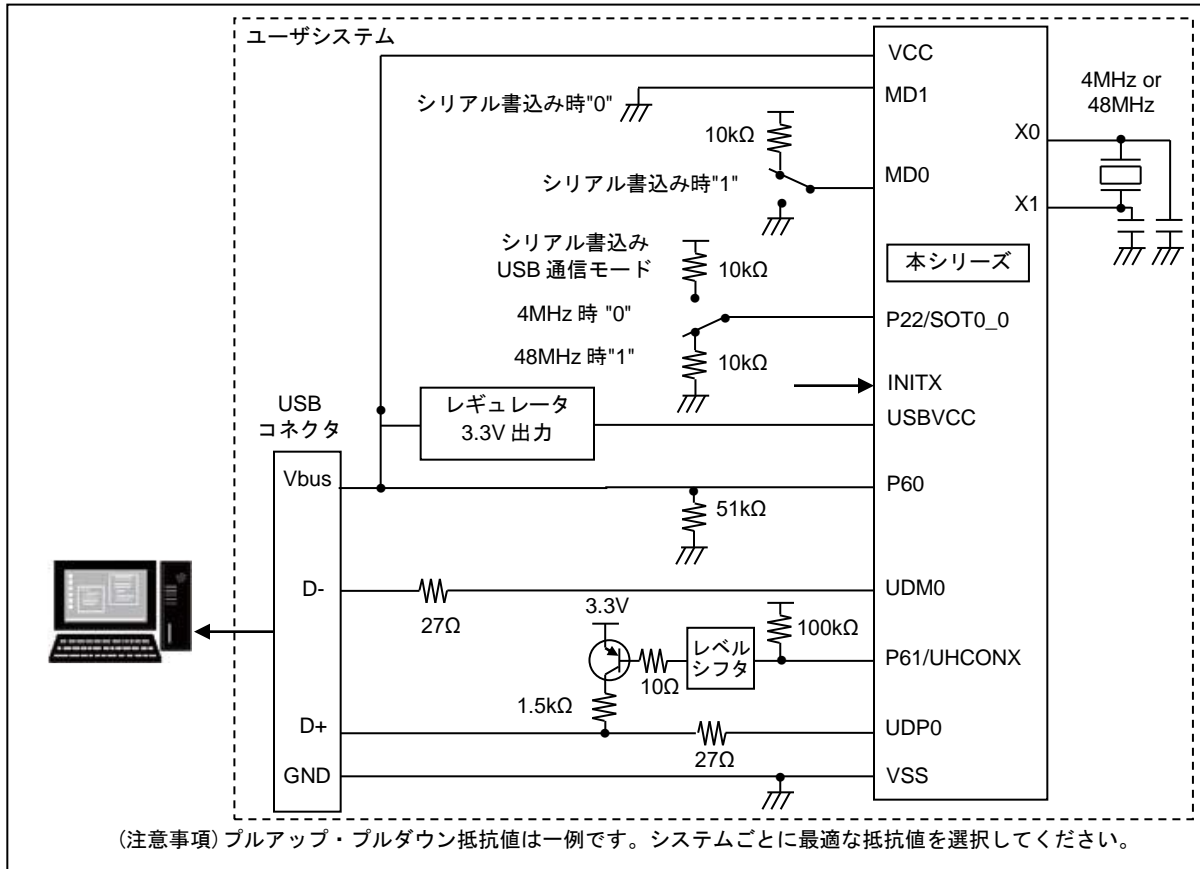


Figure 1-7 USB I/F の接続例(バス電源使用時)





## 1.2 使用端子

使用端子について説明します。

Table 1-4 シリアル書き込みに使用する端子

端子	機能	補足説明
MD0, MD1	モード端子	MD0=H, MD1=L に設定した状態で外部リセットまたは電源投入するとシリアル書き込みモードとなります。 なお、プルアップ/プルダウン抵抗を付ける場合は、配線を引き回さないでください。
X0, X1	発振用端子	シリアル書き込みモード時に使用可能な原発振クロック(メインクロック)周波数は対応する「データシート」を参照してください。 (クロック非同期通信の場合は制限があります。詳細は Table 1-2 を確認してください。)
P22/SOT0_0	UART シリアルデータ出力端子/ USB 原発振周波数選択端子	通信モードを UART とした場合、シリアル書き込みモードが起動し、通信が開始された時点でシリアルデータ出力端子となります。 通信モードを USB とした場合、原発振クロックの周波数を設定します。 P22=L: 原発振周波数 4MHz P22=H: 原発振周波数 48MHz
P21/SIN0_0	クロック同期・非同期選択端子/ UART シリアルデータ入力端子	通信を開始するまでの本端子の入力レベルを"H"にするとクロック非同期通信モードとなり、"L"にするとクロック同期通信モードとなります。通信モードを UART とした場合、シリアル書き込みモードが起動し、通信が開始された時点で UART シリアルデータ入力端子として使用できます。
P60	通信モード選択端子	シリアル書き込みモードに移行するリセット時の本端子の入力レベルにより通信モードを決定します。H のときは USB 通信モードとなり、L の時は UART 通信モードとなります。
P61/UHCONX	UDP0 用プルアップ制御端子	通信モードを USB とした場合、D+側のプルアップを制御します。 UHCONX=L: プルアップ抵抗接続 UHCONX=H: プルアップ抵抗切断
UDP0	USB 入出力端子	通信モードを USB とした場合、D+側の入出力端子となります。
UDM0	USB 入出力端子	通信モードを USB とした場合、D-側の入出力端子となります。
INITX	リセット端子	-
VCC	電源供給端子	書き込み時、マイコンの電源はユーザシステムから供給してください。
USBVCC	USB I/O 用電源供給端子	-
VSS	GND 端子	-

### <注意事項>

- シリアルライターモードで使用しないピンの初期状態は、ユーザモードの初期状態と同じです。ご使用する製品の『データシート』の『各 CPU ステートにおける端子状態』および『FM4 ファミリペリフェラルマニュアル』の関連する章を確認してください。

# 改訂履歴



## 主な変更内容

Spancion Publication Number: MN709-00005

ページ	場所	変更箇所
Revision 1.0		
-	-	Initial release
Revision 2.0		
9	CHAPTER 1:メインフラッシュメモリ 2.構成	図 2-4 を修正 図 2-5 を追加
41	CHAPTER 1:メインフラッシュメモリ 4.レジスタ 4.9 CRTRMM (CR Trimming Data Mirror Register) CR トリミングデー タ・ミラー・レジスタ	TTRMM と TRMM の記述を変更
74~75	CHAPTER 4:シリアル書き込み接続 1.1 基本構成	項目追加 FUJITSU SEMICONDUCTOR MCU Programmer の接続例 - 原発振に水晶振動子を使用する場合 - 原発振に内蔵高速 CR 発振器を使用する場合 図 1-3 を追加 図 1-6 を修正
Revision 2.1		
-	-	社名変更および記述フォーマットの変換

注意事項: 以降の変更点に関しては、「改訂履歴」を参照してください。

## 改訂履歴

文書名: CY9B560R/460R/360R/160R シリーズ 32 ビット・マイクロコントローラ FM4 ファミリ Flash Programming Specifications 文書番号: 002-04907			
版	ECN 番号	変更者	変更内容
**	-	AKIH	<p>サイプレスとしてドキュメントコード 002-04907 に登録しました。</p> <p>本版の内容およびフォーマットに変更はありません。</p> <p>(これは英語版 002-04905 Rev.** を翻訳した日本語版です。)</p>
*A	5303322	AKIH	<p>これは英語版 002-04905 Rev.*A の日本語版です。</p> <p>社名変更と記述フォーマットの変換</p>
*B	6058709	NOSU	<p>これは英語版 002-04905 Rev.*C の日本語版です。</p> <p>Cypress の新ロゴを適用</p> <p>P.3: マイコンサポート情報を追加</p> <p>P.8: ■動作モード 2.CPU プログラミングモードの説明を修正</p> <p>P.24, 27: コマンドシーケンスのアドレス表記についての注意書きを追加</p> <p>P.75: セキュリティが有効なフラッシュの消去を行うとトリミングデータが消失する旨の注意書きを追加</p> <p>P.76: 通信ボーレートについての情報を追加</p> <p>P.77, 78: FLASH MCU Programmer の接続例に外部クロックを使用する場合を追加</p> <p>P.80, 81: Vbus の結線を修正</p> <p>P.82: シリアルライターモードで使用しないピンの初期状態についての注意書きを追加</p>
*C	6466399	KTOM	<p>これは英語版の 002-04905 Rev *D を翻訳した日本語版です。</p> <p>シリーズ名</p> <p>MB9B560R/460R/360R/160R -&gt; CY9B560R/460R/360R/160R</p> <p>タイトルとカテゴリを変更</p>