
32 ビット RISC マイクロコントローラー

TXZ ファミリー

リファレンスマニュアル

フラッシュメモリー

(コードフラッシュ : 256KB/128KB/96KB/64KB)
(FLASH256-B)

Revision 1.3

2022-07

東芝デバイス&ストレージ株式会社

目次

序章	8
関連するリファレンスマニュアル	8
表記規約	9
用語・略語	11
1. 概要	12
1.1. メモリーマップ	14
2. 構成	15
2.1. ブロック図	15
2.2. コードフラッシュの構成	16
2.2.1. コードフラッシュの構成単位	16
2.2.2. コードフラッシュのブロック構成	16
2.2.3. コードフラッシュのページ構成	19
2.2.4. コードフラッシュのユーザーインフォメーションエリアの構成	21
2.2.5. コードフラッシュの書き込み、消去時間	21
2.2.6. メモリー容量と構成	21
3. 機能説明・動作説明	22
3.1. コードフラッシュ	23
3.1.1. コードフラッシュのコマンドシーケンス	23
3.1.1.1. コードフラッシュのコマンドシーケンス一覧	23
3.1.1.2. バスライトサイクル時のアドレスビット構成	25
3.1.1.3. エリアアドレス(AA)、ブロックアドレス(BA) : コードフラッシュ	27
3.1.1.4. プロテクトビットの指定(PBA) : コードフラッシュ	27
3.1.1.5. ID-Read のコード(IA, ID) : コードフラッシュ	28
3.1.1.6. メモリースワップビットの指定(MSA)	28
3.2. フローチャート	29
3.2.1. 自動プログラム	29
3.2.2. 自動消去	31
3.2.3. プロテクトビット	33
3.2.4. セキュリティービット	35
3.2.5. メモリースワップ	37
4. フラッシュメモリー詳細	39
4.1. 機能	39
4.1.1. フラッシュメモリーの動作モード	40
4.1.2. コマンド実行方法	40
4.1.3. コマンド説明	42
4.1.3.1. 自動プログラム	42
4.1.3.2. 自動チップ消去	43
4.1.3.3. 自動エリア消去	43

4.1.3.4. 自動ブロック消去.....	44
4.1.3.5. 自動ページ消去.....	44
4.1.3.6. 自動プロテクトビットプログラム.....	44
4.1.3.7. 自動プロテクトビット消去.....	45
4.1.3.8. 自動セキュリティビットプログラム.....	45
4.1.3.9. 自動セキュリティビット消去.....	45
4.1.3.10. ID-Read.....	46
4.1.3.11. Read/リセットコマンド.....	46
4.1.3.12. 自動メモリスワップ.....	47
4.1.3.13. 自動メモリスワップ消去.....	47
4.1.4. 自動チップ消去動作の中止.....	48
4.1.5. 自動動作の完了検知.....	48
4.1.5.1. 手順.....	48
4.1.6. プロテクト機能.....	49
4.1.6.1. プロテクトの設定方法.....	49
4.1.6.2. プロテクトの解除方法.....	49
4.1.6.3. プロテクトの一時解除機能.....	49
4.1.7. セキュリティ機能.....	50
4.1.7.1. セキュリティの設定方法.....	50
4.1.7.2. セキュリティの解除方法.....	50
4.1.7.3. 動作.....	50
4.1.8. メモリスワップ機能.....	51
4.1.8.1. メモリスワップの設定方法.....	51
4.1.8.2. 操作方法.....	51
4.1.8.3. メモリスワップ情報の消去.....	52
4.1.9. ユーザーインフォメーションエリア.....	53
4.1.9.1. ユーザーインフォメーションエリアの切替え手順.....	53
4.1.9.2. ユーザーインフォメーションエリアへの書き込み方法.....	53
4.1.9.3. ユーザーインフォメーションエリアの消去方法.....	53
5. レジスター説明.....	54
5.1. レジスター一覧.....	54
5.2. レジスターの詳細.....	55
5.2.1. [FCSBMR] Flash Security Bit Mask Register.....	55
5.2.2. [FCSSR] Flash Security Status Register.....	55
5.2.3. [FCKCR] Flash Key Code Register.....	55
5.2.4. [FCSR0] Flash Status Register 0.....	56
5.2.5. [FCPSR0] Flash Protect Status Register 0.....	56
5.2.6. [FCPSR1] Flash Protect Status Register 1.....	57
5.2.7. [FCPMR0] Flash Protect Mask Register 0.....	57
5.2.8. [FCPMR1] Flash Protect Mask Register 1.....	58
5.2.9. [FCSR1] Flash Status Register 1.....	58
5.2.10. [FCSWPSR] Flash Memory SWAP Status Register.....	59

5.2.11. [FCAREASEL] Flash Area Selection Register	60
5.2.12. [FCCR] Flash Control Register	61
5.2.13. [FCSTSCCLR] Flash Status Clear Register	61
5.2.14. [FCBNKCR] Flash Bank Change Register	61
5.2.15. [FCBUFDISCLR] Flash Buffer Disable and Clear Register	62
6. プログラミング方法.....	63
6.1. 初期化.....	63
6.2. モードの説明.....	63
6.3. モードの決定.....	64
6.4. モードごとのメモリーマップ	64
6.5. 書き替え方法.....	65
6.5.1. (1-A)書き替えルーチンをフラッシュメモリーに内蔵する場合の手順例.....	65
6.5.1.1. Step-1	65
6.5.1.2. Step-2	66
6.5.1.3. Step-3	66
6.5.1.4. Step-4	67
6.5.1.5. Step-5	67
6.5.1.6. Step-6	68
6.5.2. (1-B)書き替えルーチンを外部から転送する手順例.....	68
6.5.2.1. Step-1	68
6.5.2.2. Step-2	69
6.5.2.3. Step-3	69
6.5.2.4. Step-4	70
6.5.2.5. Step-5	70
6.5.2.6. Step-6	71
6.6. シングルブートモードによる書き替え方法.....	72
6.6.1. 概要	72
6.6.2. モード設定.....	73
6.6.3. インターフェース仕様.....	73
6.6.4. メモリーの制約について	74
6.6.5. 動作コマンド	74
6.6.5.1. RAM 転送.....	74
6.6.5.2. フラッシュメモリー消去.....	74
6.6.6. コマンドによらず共通の動作	75
6.6.6.1. シリアル動作モード判定	75
6.6.6.2. ACK 応答データ	77
6.6.6.3. パスワード	78
6.6.6.4. パスワード判定	81
6.6.6.5. CHECKSUM の計算方法	81
6.6.7. シリアル動作モード判定の通信ルール.....	82
6.6.8. RAM 転送コマンドの通信ルール.....	83
6.6.9. フラッシュメモリー消去の通信ルール.....	85

6.6.10. 内蔵ブートプログラム全体フローチャート.....	86
6.6.11. 内蔵ブート ROM の書き替えアルゴリズムを利用した書き替え手順.....	87
6.6.11.1. Step-1	87
6.6.11.2. Step-2	87
6.6.11.3. Step-3	88
6.6.11.4. Step-4	88
6.6.11.5. Step-5	89
6.6.11.6. Step-6	89
6.7. ユーザーブートプログラムの書き替え方法.....	90
6.7.1. フラッシュ書き替えの手順例.....	90
6.7.1.1. Step-1	90
6.7.1.2. Step-2	91
6.7.1.3. Step-3	91
6.7.1.4. Step-4	92
6.7.1.5. Step-5	92
6.7.1.6. Step-6	93
6.7.1.7. Step-7	93
6.7.1.8. Step-8	94
6.7.1.9. Step-9	94
6.7.1.10. Step-10	95
7. 使用上のご注意およびお願い事項.....	96
8. 改訂履歴.....	97
製品取り扱い上のお願ひ	99

目次

図 1.1	256KB コードフラッシュのメモリーマップ例	14
図 2.1	フラッシュメモリーのブロック図	15
図 3.1	自動プログラムのフロー(1)	29
図 3.2	自動プログラムのフロー(2)	30
図 3.3	自動消去のフロー(1)	31
図 3.4	自動消去のフロー(2)	32
図 3.5	プロテクトのフロー(1)	33
図 3.6	プロテクトのフロー(2)	34
図 3.7	セキュリティーのフロー(1)	35
図 3.8	セキュリティーのフロー(2)	36
図 3.9	メモリースワップのフロー(1)	37
図 3.10	メモリースワップのフロー(2)	38
図 4.1	メモリースワップ実施手順の例	52
図 6.1	書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(1)	65
図 6.2	書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(2)	66
図 6.3	書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(3)	66
図 6.4	書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(4)	67
図 6.5	書き替えルーチンをフラッシュメモリーに内蔵場合の手順(5)	67
図 6.6	書き替えルーチンをフラッシュメモリーに内蔵場合の手順(6)	68
図 6.7	書き替えルーチンを外部から転送する場合の手順(1)	68
図 6.8	書き替えルーチンを外部から転送する場合の手順(2)	69
図 6.9	書き替えルーチンを外部から転送場合の手順(3)	69
図 6.10	書き替えルーチンを外部から転送場合の手順(4)	70
図 6.11	書き替えルーチンを外部から転送場合の手順(5)	70
図 6.12	書き替えルーチンを外部から転送場合の手順(6)	71
図 6.13	シリアル動作モード判定データ	75
図 6.14	シリアル動作モード受信フローチャート	76
図 6.15	シリアル動作モード判定フローチャート	77
図 6.16	パスワードの構成(送信例)	79
図 6.17	パスワードチェックフローチャート	81
図 6.18	内蔵ブートプログラム全体フローチャート	86
図 6.19	内蔵ブートプログラムを利用した書き替え手順(1)	87
図 6.20	内蔵ブートプログラムを利用した書き替え手順(2)	87
図 6.21	内蔵ブートプログラムを利用した書き替え手順(3)	88
図 6.22	内蔵ブートプログラムを利用した書き替え手順(4)	88
図 6.23	内蔵ブートプログラムを利用した書き替え手順(5)	89
図 6.24	内蔵ブートプログラムを利用した書き替え手順(6)	89
図 6.25	ユーザーブートプログラムの書き替え(1)	90
図 6.26	ユーザーブートプログラムの書き替え(2)	91
図 6.27	ユーザーブートプログラムの書き替え(3)	91
図 6.28	ユーザーブートプログラムの書き替え(4)	92
図 6.29	ユーザーブートプログラムの書き替え(5)	92
図 6.30	ユーザーブートプログラムの書き替え(6)	93
図 6.31	ユーザーブートプログラムの書き替え(7)	93
図 6.32	ユーザーブートプログラムの書き替え(8)	94
図 6.33	ユーザーブートプログラムの書き替え(9)	94
図 6.34	ユーザーブートプログラムの書き替え(10)	95

表目次

表 1.1	機能概要(コードフラッシュ).....	12
表 1.2	機能概要(ユーザーインフォメーションエリア).....	13
表 2.1	信号一覧.....	15
表 2.2	256KB コードフラッシュのブロック構成.....	16
表 2.3	128KB コードフラッシュのブロック構成.....	17
表 2.4	96KB コードフラッシュのブロック構成.....	17
表 2.5	64KB コードフラッシュのブロック構成.....	18
表 2.6	256KB コードフラッシュのページ構成(1).....	19
表 2.7	256KB コードフラッシュのページ構成(2).....	20
表 2.8	コードフラッシュのユーザーインフォメーションエリア.....	21
表 2.9	コードフラッシュのメモリー容量と構成.....	21
表 3.1	JEDEC 準拠機能.....	22
表 3.2	内部 CPU によるフラッシュメモリーアクセス.....	23
表 3.3	バスライトサイクル時のアドレスビット構成.....	25
表 3.4	プロテクトビットプログラムアドレス表.....	27
表 3.5	ID-Read コマンドのコード指定とコードの内容.....	28
表 3.6	メモリスワップコマンドによる[FCSWPSR]への設定値とアドレス例.....	28
表 4.1	フラッシュメモリーの機能.....	39
表 4.2	フラッシュの書き込み/消去動作などの完了検知.....	48
表 4.3	セキュリティー有効時の動作.....	50
表 6.1	モードと動作.....	63
表 6.2	動作モード設定表.....	64
表 6.3	機能とコマンド.....	72
表 6.4	使用端子例.....	73
表 6.5	シングルブート時のメモリーの制約.....	74
表 6.6	シングルブート時コマンド.....	74
表 6.7	転送可能なボーレートの設定例(fc=10MHz、誤差含まず).....	75
表 6.8	シリアル動作判定データに対する ACK 応答データ.....	77
表 6.9	動作コマンドデータに対する ACK 応答データ.....	77
表 6.10	CHECKSUM データに対する ACK 応答データ.....	77
表 6.11	フラッシュメモリー消去動作に対する ACK 対応データ.....	78
表 6.12	パスワードの設定値、設定範囲.....	80
表 6.13	シリアル動作モード判定の通信ルール.....	82
表 6.14	RAM 転送コマンドの通信ルール.....	83
表 6.15	フラッシュメモリー消去の通信ルール.....	85
表 8.1	改訂履歴.....	97

序章

関連するリファレンスマニュアル

文書名
クロック制御と動作モード
例外
入出力ポート
製品個別情報
非同期シリアル通信回路

表記規約

- 数値表記は以下の規則に従います。
 - 16 進数表記: 0xABC
 - 10 進数表記: 123 または 0d123 (10 進表記であることを示す必要のある場合だけ使用)
 - 2 進数表記: 0b111 (ビット数が本文中に明記されている場合は「0b」を省略可)
- ローアクティブの信号は信号名の末尾に「_N」で表記します。
- 信号がアクティブレベルに移ることを「アサート(assert)」アクティブでないレベルに移ることを「デアサート(deassert)」と呼びます。
- 複数の信号名は[m:n]とまとめて表記する場合があります。
例: S[3:0]は S3,S2,S1,S0 の 4 つの信号名をまとめて表記しています。
- 本文中[]で囲まれたものはレジスターを定義しています。
例: [ABCD]
- 同種で複数のレジスター、フィールド、ビット名は「n」で一括表記する場合があります。
例: [XYZ1], [XYZ2], [XYZ3] → [XYZn]
- 「レジスター一覧」中のレジスター名でユニットまたはチャンネルは「x」で一括表記しています。
ユニットの場合、「x」は A,B,C...を表します。
例: [ADACR0], [ADBCR0], [ADCCR0] → [ADxCR0]
チャンネルの場合、「x」は 0,1,2...を表します。
例: [T32A0RUNA], [T32A1RUNA], [T32A2RUNA] → [T32AxRUNA]
- レジスターのビット範囲は[m:n]と表記します。
例: [3:0]はビット 3 から 0 の範囲を表します。
- レジスターの設定値は 16 進数または 2 進数のどちらかで表記されています。
例: [ABCD]<EFG> = 0x01 (16 進数)、[XYZn]<VW> = 1 (2 進数)
- ワード、バイトは以下のビット長を表します。
 - バイト: 8 ビット
 - ハーフワード: 16 ビット
 - ワード: 32 ビット
 - ダブルワード: 64 ビット
- レジスター内の各ビットの属性は以下の表記を使用しています。
 - R: リードオンリー
 - W: ライトオンリー
 - R/W: リード / ライト
- 断りのない限り、レジスターアクセスはワードアクセスだけをサポートします。
- 本文中の予約領域「Reserved」として定義されたレジスターは書き換えを行わないでください。
また、読み出した値を使用しないでください。
- Default 値が「—」となっているビットから読み出した値は不定です。
- 書き込み可能なビットフィールドと、リードオンリー「R」のビットフィールドが共存するレジスターに書き込みを行う場合、リードオンリー「R」のビットフィールドには Default 値を書き込んでください。
Default 値が「—」となっている場合は、個々のレジスターの定義に従ってください。
- ライトオンリーのレジスターの Reserved ビットフィールドには Default 値を書き込んでください。
Default 値が「—」となっている場合は、個々のレジスターの定義に従ってください。
- 書き込みと読み出しで異なる定義のレジスターへのリードモディファイライト処理は行わないでください。

Arm, Cortex および Thumb は Arm Limited(またはその子会社)の US またはその他の国における登録商標です。 All rights reserved.



FLASH メモリーについては、米国 SST 社 (Silicon Storage Technology, Inc.) からライセンスを受けた Super Flash® 技術を使用しています。Super Flash® は SST 社の登録商標です。

本資料に記載されている社名・商品名・サービス名などは、それぞれ各社が商標として使用している場合があります。

用語・略語

この仕様書で使用されている用語・略語の一部を記載します。

ACK	Acknowledgement
Addr	Address
Adr	Address
BLK	Block
KB	Kilo Bytes
PG	Page
SFR	Special Function Register
UART	Universal Asynchronous Receiver Transmitter

1. 概要

プログラムを格納するコードフラッシュについて説明します。コードフラッシュは命令コードを格納し、CPU がリードして実行します。

コードフラッシュには、バンク切替えによってアクセスが可能なユーザーインフォメーションエリアがあります。ユーザーインフォメーションエリアは、チップ消去コマンドでは消去されませんので、チップごとにユニークな管理番号などを書き込んでおくことができます。

表 1.1 機能概要(コードフラッシュ)

領域	機能	基本動作	動作	備考
コードフラッシュ 256KB 128KB 96KB 64KB	フラッシュプログラム/ 消去動作	自動プログラム	4ワード(16バイト)でデータ書き込みを行います。	
		自動チップ消去	フラッシュメモリーの全領域の消去を自動で行います。 対象: コードフラッシュ	ユーザーインフォメーションエリアを除く
		自動エリア消去	エリア単位での消去を自動で行います。	
		自動ブロック消去	ブロック単位での消去を自動で行います。	
		自動ページ消去	ページ単位での消去を自動で行います。	
	リード/ライトプロテクト機能	プロテクト	ブロック単位で書き込みおよび消去を禁止することができます。(注)	
	セキュリティ	セキュリティ	フラッシュライターによるフラッシュメモリーの読み出しの禁止およびデバッグ機能の使用制限をすることができます。	
	メモリスワップ	自動メモリスワップ	コードフラッシュ領域のスワップ/スワップ解除/スワップサイズ指定を自動で行います。	
命令の実行	命令の実行	命令の実行が可能です		

注) 先頭の 32KB 分はページ単位でプロテクトします。

表 1.2 機能概要(ユーザーインフォメーションエリア)

領域	機能	基本動作	動作	備考
ユーザーインフォメーションエリア (コードフラッシュ) 4KB	フラッシュプログラム/ 消去動作	自動プログラム	4ワード(16バイト)単位でデータ書き込みを自動で行います。	
		自動ページ消去	ユーザーインフォメーションエリア(全て)を自動で消去します。	
	セキュリティ	セキュリティ	フラッシュライターによるフラッシュメモリーの読み出しの禁止およびデバッグ機能の使用制限をすることができます。	コードフラッシュ側の操作で同時に有効になります。
	命令の実行	—	—	命令の実行はできません。

1.1. メモリーマップ

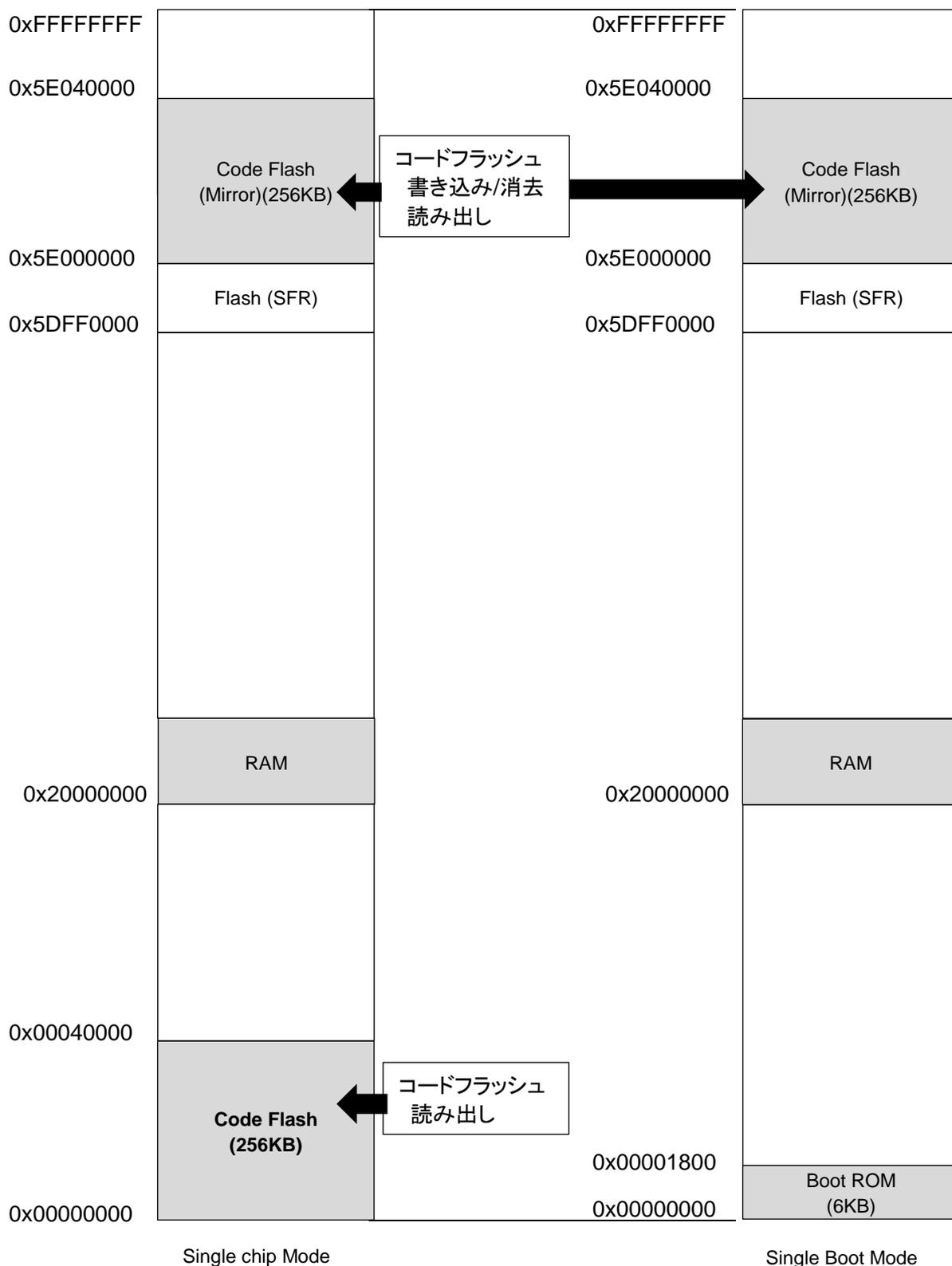


図 1.1 256KBコードフラッシュのメモリーマップ例

2. 構成

2.1. ブロック図

フラッシュメモリーのブロック図と信号一覧を示します。

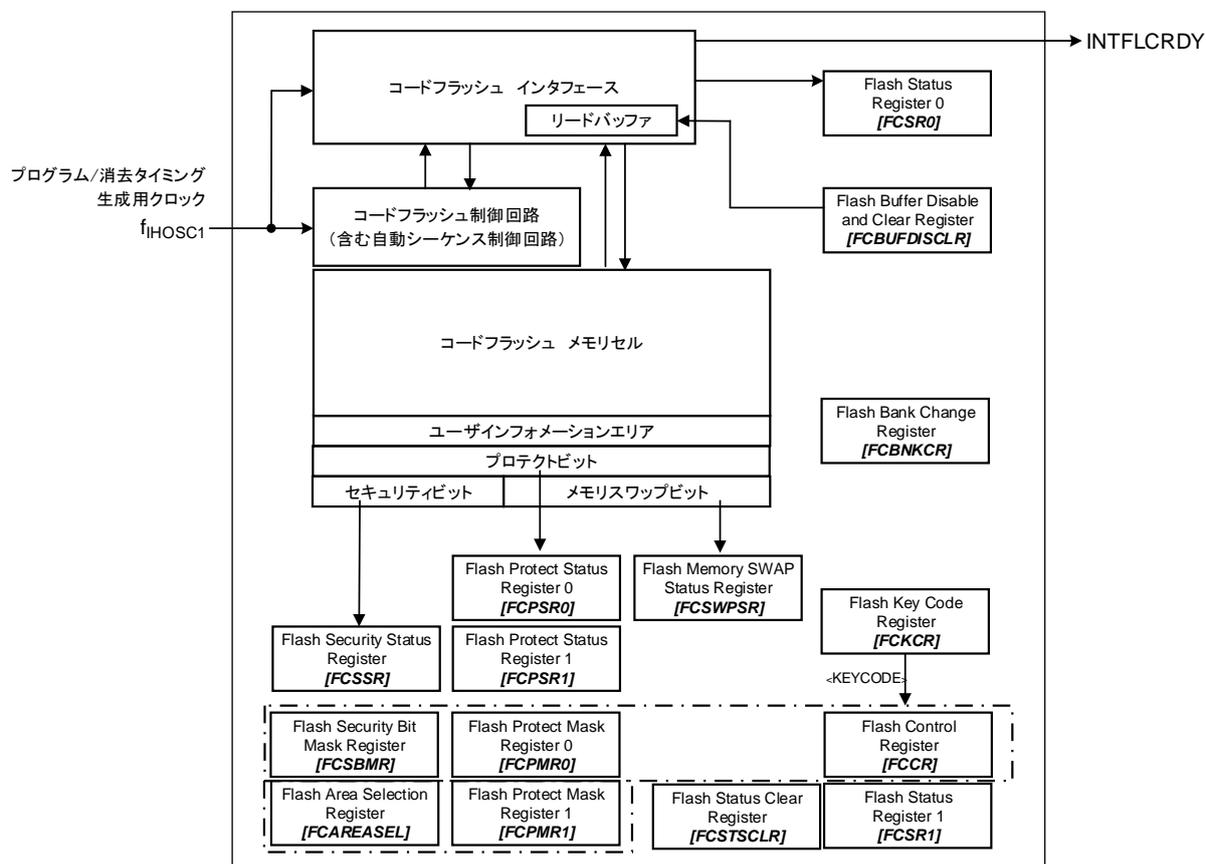


図 2.1 フラッシュメモリーのブロック図

表 2.1 信号一覧

No	記載名	信号名称	I/O	参照リファレンスマニュアル
1	f_{IHOSC1}	プログラム/消去タイミング生成用クロック	入力	クロック制御と動作モード
2	INTFLCRDY	Code FLASH Ready 割り込み	出力	例外

2.2. コードフラッシュの構成

2.2.1. コードフラッシュの構成単位

コードフラッシュの構成の単位として、「エリア」、「ブロック」、「ページ」があります。

- エリア
消去機能で使します。
1 エリアは最大 256KB です。製品のメモリーサイズによって異なります。
- ブロック
消去機能、プロテクト機能で使します。
1 ブロックは 32KB です。
- ページ
消去機能、プロテクト機能で使します。
1 ページは 4096 バイト固定です。

2.2.2. コードフラッシュのブロック構成

表 2.2 256KBコードフラッシュのブロック構成

エリア	ブロック名称	コード実行アドレス	書き込み/消去/読み出しアドレス	ブロックサイズ (KB)	
0	Block0 (注)	PG0	0x00000000-0x00000FFF	0x5E000000-0x5E000FFF	4
		PG1	0x00001000-0x00001FFF	0x5E001000-0x5E001FFF	4
		PG2	0x00002000-0x00002FFF	0x5E002000-0x5E002FFF	4
		PG3	0x00003000-0x00003FFF	0x5E003000-0x5E003FFF	4
		PG4	0x00004000-0x00004FFF	0x5E004000-0x5E004FFF	4
		PG5	0x00005000-0x00005FFF	0x5E005000-0x5E005FFF	4
		PG6	0x00006000-0x00006FFF	0x5E006000-0x5E006FFF	4
		PG7	0x00007000-0x00007FFF	0x5E007000-0x5E007FFF	4
	Block1	0x00008000-0x0000FFFF	0x5E008000-0x5E00FFFF	32	
	Block2	0x00010000-0x00017FFF	0x5E010000-0x5E017FFF	32	
	Block3	0x00018000-0x0001FFFF	0x5E018000-0x5E01FFFF	32	
	Block4	0x00020000-0x00027FFF	0x5E020000-0x5E027FFF	32	
	Block5	0x00028000-0x0002FFFF	0x5E028000-0x5E02FFFF	32	
	Block6	0x00030000-0x00037FFF	0x5E030000-0x5E037FFF	32	
	Block7	0x00038000-0x0003FFFF	0x5E038000-0x5E03FFFF	32	

注) Block0 は PG0～PG7 の総称です。PG0～PG7 としてアクセス可能です。

表 2.3 128KBコードフラッシュのブロック構成

エリア	ブロック名称		コード実行アドレス	書き込み/消去/読み出しアドレス	ブロックサイズ (KB)
0	Block0 (注)	PG0	0x00000000-0x00000FFF	0x5E000000-0x5E000FFF	4
		PG1	0x00001000-0x00001FFF	0x5E001000-0x5E001FFF	4
		PG2	0x00002000-0x00002FFF	0x5E002000-0x5E002FFF	4
		PG3	0x00003000-0x00003FFF	0x5E003000-0x5E003FFF	4
		PG4	0x00004000-0x00004FFF	0x5E004000-0x5E004FFF	4
		PG5	0x00005000-0x00005FFF	0x5E005000-0x5E005FFF	4
		PG6	0x00006000-0x00006FFF	0x5E006000-0x5E006FFF	4
		PG7	0x00007000-0x00007FFF	0x5E007000-0x5E007FFF	4
	Block1		0x00008000-0x0000FFFF	0x5E008000-0x5E00FFFF	32
	Block2		0x00010000-0x00017FFF	0x5E010000-0x5E017FFF	32
	Block3		0x00018000-0x0001FFFF	0x5E018000-0x5E01FFFF	32

注) Block0 は PG0～PG7 の総称です。PG0～PG7 としてアクセス可能です。

表 2.4 96KBコードフラッシュのブロック構成

エリア	ブロック名称		コード実行アドレス	書き込み/消去/読み出しアドレス	ブロックサイズ (KB)
0	Block0 (注)	PG0	0x00000000-0x00000FFF	0x5E000000-0x5E000FFF	4
		PG1	0x00001000-0x00001FFF	0x5E001000-0x5E001FFF	4
		PG2	0x00002000-0x00002FFF	0x5E002000-0x5E002FFF	4
		PG3	0x00003000-0x00003FFF	0x5E003000-0x5E003FFF	4
		PG4	0x00004000-0x00004FFF	0x5E004000-0x5E004FFF	4
		PG5	0x00005000-0x00005FFF	0x5E005000-0x5E005FFF	4
		PG6	0x00006000-0x00006FFF	0x5E006000-0x5E006FFF	4
		PG7	0x00007000-0x00007FFF	0x5E007000-0x5E007FFF	4
	Block1		0x00008000-0x0000FFFF	0x5E008000-0x5E00FFFF	32
	Block2		0x00010000-0x00017FFF	0x5E010000-0x5E017FFF	32

注) Block0 は PG0～PG7 の総称です。PG0～PG7 としてアクセス可能です。

表 2.5 64KBコードフラッシュのブロック構成

エリア	ブロック名称		コード実行アドレス	書き込み/消去/読み出しアドレス	ブロック サイズ (KB)
0	Block0 (注)	PG0	0x00000000-0x00000FFF	0x5E000000-0x5E000FFF	4
		PG1	0x00001000-0x00001FFF	0x5E001000-0x5E001FFF	4
		PG2	0x00002000-0x00002FFF	0x5E002000-0x5E002FFF	4
		PG3	0x00003000-0x00003FFF	0x5E003000-0x5E003FFF	4
		PG4	0x00004000-0x00004FFF	0x5E004000-0x5E004FFF	4
		PG5	0x00005000-0x00005FFF	0x5E005000-0x5E005FFF	4
		PG6	0x00006000-0x00006FFF	0x5E006000-0x5E006FFF	4
		PG7	0x00007000-0x00007FFF	0x5E007000-0x5E007FFF	4
		Block1	0x00008000-0x0000FFFF	0x5E008000-0x5E00FFFF	32

注) Block0 は PG0~PG7 の総称です。PG0~PG7 としてアクセス可能です。

2.2.3. コードフラッシュのページ構成

256KB コードフラッシュの例を表 2.6 に示します。

表 2.6 256KBコードフラッシュのページ構成(1)

エリア	ページ名称	コード実行アドレス	書き込み/消去/読み出しアドレス	ページ サイズ (KB)
0	Page0	0x00000000-0x00000FFF	0x5E000000-0x5E000FFF	4
	Page1	0x00001000-0x00001FFF	0x5E001000-0x5E001FFF	4
	Page2	0x00002000-0x00002FFF	0x5E002000-0x5E002FFF	4
	Page3	0x00003000-0x00003FFF	0x5E003000-0x5E003FFF	4
	Page4	0x00004000-0x00004FFF	0x5E004000-0x5E004FFF	4
	Page5	0x00005000-0x00005FFF	0x5E005000-0x5E005FFF	4
	Page6	0x00006000-0x00006FFF	0x5E006000-0x5E006FFF	4
	Page7	0x00007000-0x00007FFF	0x5E007000-0x5E007FFF	4
	Page8	0x00008000-0x00008FFF	0x5E008000-0x5E008FFF	4
	Page9	0x00009000-0x00009FFF	0x5E009000-0x5E009FFF	4
	Page10	0x0000A000-0x0000AFFF	0x5E00A000-0x5E00AFFF	4
	Page11	0x0000B000-0x0000BFFF	0x5E00B000-0x5E00BFFF	4
	Page12	0x0000C000-0x0000CFFF	0x5E00C000-0x5E00CFFF	4
	Page13	0x0000D000-0x0000DFFF	0x5E00D000-0x5E00DFFF	4
	Page14	0x0000E000-0x0000EFFF	0x5E00E000-0x5E00EFFF	4
	Page15	0x0000F000-0x0000FFFF	0x5E00F000-0x5E00FFFF	4
	Page16	0x00010000-0x00010FFF	0x5E010000-0x5E010FFF	4
	Page17	0x00011000-0x00011FFF	0x5E011000-0x5E011FFF	4
	Page18	0x00012000-0x00012FFF	0x5E012000-0x5E012FFF	4
	Page19	0x00013000-0x00013FFF	0x5E013000-0x5E013FFF	4
	Page20	0x00014000-0x00014FFF	0x5E014000-0x5E014FFF	4
	Page21	0x00015000-0x00015FFF	0x5E015000-0x5E015FFF	4
	Page22	0x00016000-0x00016FFF	0x5E016000-0x5E016FFF	4
	Page23	0x00017000-0x00017FFF	0x5E017000-0x5E017FFF	4
	Page24	0x00018000-0x00018FFF	0x5E018000-0x5E018FFF	4
	Page25	0x00019000-0x00019FFF	0x5E019000-0x5E019FFF	4
	Page26	0x0001A000-0x0001AFFF	0x5E01A000-0x5E01AFFF	4
	Page27	0x0001B000-0x0001BFFF	0x5E01B000-0x5E01BFFF	4
	Page28	0x0001C000-0x0001CFFF	0x5E01C000-0x5E01CFFF	4
	Page29	0x0001D000-0x0001DFFF	0x5E01D000-0x5E01DFFF	4
	Page30	0x0001E000-0x0001EFFF	0x5E01E000-0x5E01EFFF	4
	Page31	0x0001F000-0x0001FFFF	0x5E01F000-0x5E01FFFF	4

表 2.7 256KBコードフラッシュのページ構成(2)

エリア	ページ名称	コード実行アドレス	書き込み/消去/読み出しアドレス	ページサイズ (KB)
0	Page32	0x00020000-0x00020FFF	0x5E020000-0x5E020FFF	4
	Page33	0x00021000-0x00021FFF	0x5E021000-0x5E021FFF	4
	Page34	0x00022000-0x00022FFF	0x5E022000-0x5E022FFF	4
	Page35	0x00023000-0x00023FFF	0x5E023000-0x5E023FFF	4
	Page36	0x00024000-0x00024FFF	0x5E024000-0x5E024FFF	4
	Page37	0x00025000-0x00025FFF	0x5E025000-0x5E025FFF	4
	Page38	0x00026000-0x00026FFF	0x5E026000-0x5E026FFF	4
	Page39	0x00027000-0x00027FFF	0x5E027000-0x5E027FFF	4
	Page40	0x00028000-0x00028FFF	0x5E028000-0x5E028FFF	4
	Page41	0x00029000-0x00029FFF	0x5E029000-0x5E029FFF	4
	Page42	0x0002A000-0x0002AFFF	0x5E02A000-0x5E02AFFF	4
	Page43	0x0002B000-0x0002BFFF	0x5E02B000-0x5E02BFFF	4
	Page44	0x0002C000-0x0002CFFF	0x5E02C000-0x5E02CFFF	4
	Page45	0x0002D000-0x0002DFFF	0x5E02D000-0x5E02DFFF	4
	Page46	0x0002E000-0x0002EFFF	0x5E02E000-0x5E02EFFF	4
	Page47	0x0002F000-0x0002FFFF	0x5E02F000-0x5E02FFFF	4
	Page48	0x00030000-0x00030FFF	0x5E030000-0x5E030FFF	4
	Page49	0x00031000-0x00031FFF	0x5E031000-0x5E031FFF	4
	Page50	0x00032000-0x00032FFF	0x5E032000-0x5E032FFF	4
	Page51	0x00033000-0x00033FFF	0x5E033000-0x5E033FFF	4
	Page52	0x00034000-0x00034FFF	0x5E034000-0x5E034FFF	4
	Page53	0x00035000-0x00035FFF	0x5E035000-0x5E035FFF	4
	Page54	0x00036000-0x00036FFF	0x5E036000-0x5E036FFF	4
	Page55	0x00037000-0x00037FFF	0x5E037000-0x5E037FFF	4
	Page56	0x00038000-0x00038FFF	0x5E038000-0x5E038FFF	4
	Page57	0x00039000-0x00039FFF	0x5E039000-0x5E039FFF	4
	Page58	0x0003A000-0x0003AFFF	0x5E03A000-0x5E03AFFF	4
	Page59	0x0003B000-0x0003BFFF	0x5E03B000-0x5E03BFFF	4
	Page60	0x0003C000-0x0003CFFF	0x5E03C000-0x5E03CFFF	4
	Page61	0x0003D000-0x0003DFFF	0x5E03D000-0x5E03DFFF	4
	Page62	0x0003E000-0x0003EFFF	0x5E03E000-0x5E03EFFF	4
	Page63	0x0003F000-0x0003FFFF	0x5E03F000-0x5E03FFFF	4

2.2.4. コードフラッシュのユーザーインフォメーションエリアの構成

表 2.8 コードフラッシュのユーザーインフォメーションエリア

エリア	ユーザー インフォメーション エリア	書き込み/消去/読み出しアドレス	ページ サイズ (KB)
0	Page5	0x5E005000-0x5E005FFF	4

2.2.5. コードフラッシュの書き込み、消去時間

書き込みは 16 バイト単位(4 バイト×4 回)で行います。

消去はページ単位、ブロック単位、エリア単位またはチップ全体で行います。消去時間は使用するコマンドによって異なります。詳細は「2.2.6 メモリー容量と構成」を参照してください。

2.2.6. メモリー容量と構成

表 2.9 コードフラッシュのメモリー容量と構成

容量 (KB)	エリア		ブロック		ページ		書き込み時間 (注1)		消去時間(注 1)			
	サイズ (KB)	個数	サイズ (KB)	個数	サイズ (KB)	個数	ワード (注 2)	エリア	ページ	ブロック	エリア	チップ
256	256	1	32	8	4	64	29.5μs	1.85s	1.1ms	8.6ms	9.2ms	12.5ms (注 3)
128	128	1	32	4	32	0.97s						
96	96	1	32	3	24	0.73s						
64	64	1	32	2	16	0.49s						

注 1) 書き込み時間/消去時間は、IHOSC1 の発振周波数を基準(10MHz<Typ.>)に計算した参考値です。各レジスターがリセット後の初期値の場合を表しており、データ転送時間などは含まれていません。

注 2) 書き込みは 4 ワード(WORD)単位で行われますので上記の 4 倍の時間が必要です。

注 3) プロテクト状態のブロックが無い場合です。プロテクトビット、セキュリティービット、ユーザーインフォメーションエリアの消去時間を含みます。

3. 機能説明・動作説明

コードフラッシュは、一部の機能を除き JEDEC 標準機能に準拠しています。このため、外部メモリーとしてフラッシュメモリーをご使用になられている場合でも、本製品への移行が容易です。また、フラッシュメモリー内に書き込み、チップ消去など自動で行う回路を内蔵していますので、書き込み、消去動作を容易に実現できます。

表 3.1 JEDEC 準拠機能

JEDEC 準拠の機能	変更, 追加, 削除した機能
<ul style="list-style-type: none"> ・自動プログラム ・自動チップ消去 ・自動ブロック消去 	<p><追加>自動エリア消去、自動ページ消去、自動メモリースワップ</p> <p><変更>ライト/消去プロテクト(ソフトウェアプロテクトのみサポート)</p> <p><削除>消去レジューム/サスペンド機能</p>

使用上の注意事項

- (1) コードフラッシュ, ユーザーインフォメーションエリアへの書き込み/消去操作、プロテクト, セキュリティなどフラッシュに関連する書き込み/消去の操作を行う前には、必ず $[CGOSCCR] <IHOSCEN1> = 1$ にして内蔵高速発振器 1 (IHOSC1) を発振させてください。発振開始後 $[CGOSCCR] <IHOSC1F> = 1$ であることを確認してから操作してください。IHOSC1 のクロックはフラッシュメモリー書き込み/消去時のタイミングクロックとなります。
- (2) 内蔵高速発振器 1 (IHOSC1) および $[CGOSCCR] <IHOSC1F>$ については、リファレンスマニュアル「クロック制御と動作モード」を参照してください。
- (3) フラッシュメモリーが書き込み/消去中 BUSY のとき ($[FCSR0] <RDYBSY> = 0$) は、電源を切断しないでください。
- (4) フラッシュメモリーが書き込み/消去中 BUSY のとき ($[FCSR0] <RDYBSY> = 0$) は、STOP1, STOP2 モードを起動しないでください。
- (5) フラッシュメモリーが書き込み/消去中 BUSY のとき ($[FCSR0] <RDYBSY> = 0$) は、SIWDT/LVD などの要因でリセットがかからないようにしてください。

3.1. コードフラッシュ

3.1.1. コードフラッシュのコマンドシーケンス

3.1.1.1. コードフラッシュのコマンドシーケンス一覧

コードフラッシュの各コマンドのバスライトサイクルのアドレスとデータを示します。

ID-Read コマンドの第 5 バスサイクル以外は全て「バスライトサイクル」です。バスライトサイクルは 32 ビット(1 ワード)のデータ転送命令で実施します(「表 3.2 内部 CPU によるフラッシュメモリーアクセス」では、データの下位 8 ビットのデータのみ示しています)。

アドレスの詳細は、「表 3.3 バスライトサイクル時のアドレスビット構成」を参照してください。「コマンド」と記載された、Addr[11:4]に下記値を使用します。

注) 各コマンドのアドレスは、Flash 領域(Mirror) に設定してください。

表 3.2 内部CPU によるフラッシュメモリーアクセス

シーケンス コマンド	第1バス サイクル	第2バス サイクル	第3バス サイクル	第4バス サイクル	第5バス サイクル	第6バス サイクル	第7バス サイクル
	アドレス						
	データ						
Read/ リセット	0xYYYYXXXX	—	—	—	—	—	—
	0xF0	—	—	—	—	—	—
ID-Read	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	IA	0xYYYYXXXX	—	—
	0xAA	0x55	0x90	0x00	ID	—	—
自動プログラ ム	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	PA	PA	PA	PA
	0xAA	0x55	0xA0	PD0	PD1	PD2	PD3
自動ページ 消去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	PGA	—
	0xAA	0x55	0x80	0xAA	0x55	0x40	—
自動ブロック 消去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	BA	—
	0xAA	0x55	0x80	0xAA	0x55	0x30	—
自動エリア消 去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	AA	—
	0xAA	0x55	0x80	0xAA	0x55	0x20	—
自動コードエ リア消去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	—
	0xAA	0x55	0x80	0xAA	0x55	0x11	—
自動チップ消 去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	—
	0xAA	0x55	0x80	0xAA	0x55	0x10	—
自動プロテク トビットプロ グラム	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	PBA	—	—	—
	0xAA	0x55	0x9A	0x9A	—	—	—
自動プロテク トビット消 去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	PBA(注)	—
	0xAA	0x55	0x80	0xAA	0x55	0x60	—
自動メモリー スワッププロ グラム	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	MSA	—	—	—
	0xAA	0x55	0x9A	0x9A	—	—	—
自動メモリー	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	MSA(注)	—

スワップ消去	0xAA	0x55	0x80	0xAA	0x55	0x60	—
自動セキュリ ティービットプ ログラム	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	SBA	—	—	—
	0xAA	0x55	0x9A	0x9A	—	—	—
自動セキュリ ティービット 消去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	SBA(注)	—
	0xAA	0x55	0x80	0xAA	0x55	0x60	—

注) 「表 3.3 バスライトサイクル時のアドレスビット構成」を参照してください。

補足説明

IA: ID アドレス

ID: ID データ 出力

PGA: ページアドレス

BA: ブロックアドレス

AA: エリアアドレス

PA: プログラムアドレス(ライト)

PD: プログラムデータ(32ビットデータ)

第 4 バスサイクル以降 4 ワード分をアドレス順にデータ入力

PBA: プロテクトビットアドレス

MSA: メモリースワップアドレス

SBA: セキュリティービットアドレス

3.1.1.2. バスライトサイクル時のアドレスビット構成

「表 3.3 バスライトサイクル時のアドレスビット構成」は、「表 3.2 内部 CPU によるフラッシュメモリーアクセス」と併せてご使用願います。

第 1 バスサイクルから「通常コマンドのバスライトサイクルアドレス設定」に従い、アドレス設定を行ってください。

表 3.3 バスライトサイクル時のアドレスビット構成

[通常コマンド]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:4]	Adr [3:0]
通常コマンド	通常コマンドのバスライトサイクルアドレス設定					
	0x5E	"00000" 固定	エリア(注)	"0" 推奨	コマンド	"0" 推奨

注) エリアは"00"固定で使用してください。

[Read/リセット、ID-Read]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:16]	Adr [15:14]	Adr [13:0]
Read /リセット	Read/リセットの第 1 バスライトサイクルアドレス設定				
	0x5E	"00000" 固定			"0" 推奨
ID-Read	IA: ID アドレス(ID-Read の第 4 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	"000" 固定	ID アドレス	"0" 推奨

[自動エリア消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:0]
エリア消去	AA: エリアアドレス(エリア消去コマンドの第 6 バスライトサイクルアドレス設定)			
	0x5E	"00000" 固定	エリア(注)	"0" 推奨

注) エリアは"00"固定で使用してください。

[自動ブロック消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:15]	Adr [14:0]
ブロック 消去	BA: ブロックアドレス(ブロック消去コマンドの第 6 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	エリア(注)	ブロック アドレス	"0" 推奨

注) エリアは"00"固定で使用してください。

[自動ページ消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:0]
ページ消去	PGA: ページアドレス(ページ消去コマンドの第 6 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	エリア(注)	ページ アドレス	"0" 推奨

注) エリアは"00"固定で使用してください。

[自動プログラム]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:4]	Adr [3:0]
プログラム	PA: プログラムアドレス(プログラムの第 4~第 7 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	エリア(注)	プログラム アドレス	"0" 推奨

注) エリアは"00"固定で使用してください。

[自動プロテクトビットプログラム/消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:4]	Adr [3:0]
プロテクトビット 消去	PBA: プロテクトビットアドレス(プロテクトビット消去の第 6 バスライトサイクルアドレス設定)					
	0x5E	"00000" 固定	エリア(注)	"00010" 固定	"0" 推奨	
プロテクトビット プログラム	PBA: プロテクトビットアドレス(プロテクトビットプログラムの第 4 バスライトサイクルアドレス設定)					
	0x5E	"00000" 固定	エリア(注)	"00010" 固定	プロテクト ビット選択	"0" 推奨

注) エリアは"00"固定で使用してください。

[自動メモリスワッププログラム/消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:4]	Adr [3:0]
メモリスワップ 消去	MSA: メモリスワップアドレス(メモリスワップ消去の第 6 バスライトサイクルアドレス設定)					
	0x5E	"00000" 固定	"00" 固定	"00011" 固定	"0" 推奨	
メモリスワップ プログラム	MSA: メモリスワップアドレス(メモリスワッププログラムの第 4 バスライトサイクルアドレス設定)					
	0x5E	"00000" 固定	"00" 固定	"00011" 固定	メモリスワップ 選択	"0" 推奨

[自動セキュリティービットプログラム/消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:0]
セキュリティー ビット消去	SBA: セキュリティービット消去アドレス(セキュリティービット消去の第 6 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	"00" 固定	"00001" 固定	"0" 推奨
セキュリティー ビットプログラム	SBA: セキュリティービットプログラムアドレス(セキュリティービットプログラムの第 4 バスライトサイクルアドレス設定)				
	0x5E	"00000" 固定	"00" 固定	"00001" 固定	"0" 推奨

3.1.1.3. エリアアドレス(AA)、ブロックアドレス(BA) : コードフラッシュ

「表 2.2 256KB コードフラッシュのブロック構成」にエリアアドレスとブロックアドレスを示します。自動エリア消去コマンドと自動ブロック消去コマンドの第 6 バスライトサイクルで、消去するエリアまたはブロックに含まれる任意のアドレスを指定します。シングルチップモードでは、ミラー領域のアドレスを指定してください。

3.1.1.4. プロテクトビットの指定(PBA) : コードフラッシュ

プロテクトビットは、1 ビット単位の操作になります。

自動プロテクトビットプログラムのプロテクトビット選択表を表 3.4 に示します。

表 3.4 プロテクトビットプログラムアドレス表

ブロック	ページ	レジスター	プロテクトビット	PBA[11:4]						アドレス例 [31:0]
				Adr [11:9]	Adr [8]	Adr [7]	Adr [6]	Adr [5]	Adr [4]	
0 (注)	0	[FCPSR0]	<PG0>	000	0	0	0	0	0	0x5E002000
	1		<PG1>	000	0	0	0	0	1	0x5E002010
	2		<PG2>	000	0	0	0	1	0	0x5E002020
	3		<PG3>	000	0	0	0	1	1	0x5E002030
	4		<PG4>	000	0	0	1	0	0	0x5E002040
	5		<PG5>	000	0	0	1	0	1	0x5E002050
	6		<PG6>	000	0	0	1	1	0	0x5E002060
	7		<PG7>	000	0	0	1	1	1	0x5E002070
1	8~15	[FCPSR1]	<BLK1>	000	0	1	0	0	0	0x5E002080
2	16~23		<BLK2>	000	0	1	0	0	1	0x5E002090
3	24~31		<BLK3>	000	0	1	0	1	0	0x5E0020A0
4	32~39		<BLK4>	000	0	1	0	1	1	0x5E0020B0
5	40~47		<BLK5>	000	0	1	1	0	0	0x5E0020C0
6	48~55		<BLK6>	000	0	1	1	0	1	0x5E0020D0
7	56~63		<BLK7>	000	0	1	1	1	0	0x5E0020E0

注) ブロック 0 は<BLK0>ではなく、<PG0>~<PG7>の総称です。

3.1.1.5. ID-Read のコード(IA, ID) : コードフラッシュ

ID-Read コマンドでのコード指定方法と読み出される内容を「表 3.5 ID-Read コマンドのコード指定とコードの内容」に示します。

表 3.5 ID-Read コマンドのコード指定とコードの内容

Code	ID[15:0]	IA[15:14]	アドレス例[31:0]
メーカーコード	0x0098	00	0x5E000000
デバイスコード	0x005A	01	0x5E004000
—	Reserved	10	N/A
マクロコード	0x0207	11	0x5E00C000

3.1.1.6. メモリースワップビットの指定(MSA)

自動メモリースワップコマンドの第4バスライトサイクルで指定する[FCSWPSR]<SWP[1:0]> <SIZE[5:0]>への設定値を「表 3.6 メモリースワップコマンドによる[FCSWPSR]への設定値とアドレス例」に示します。

表 3.6 メモリースワップコマンドによる[FCSWPSR]への設定値とアドレス例

レジスター		MSA[11:4]						アドレス例 [31:0]
		Adr [11:9]	Adr [8]	Adr [7]	Adr [6]	Adr [5]	Adr [4]	
[FCSWPSR]	<SWP[0]>	000	0	0	0	0	0	0x5E003000
	<SWP[1]>	000	0	0	0	0	1	0x5E003010
	<SIZE[0]>	000	0	0	0	1	0	0x5E003020
	<SIZE[1]>	000	0	0	0	1	1	0x5E003030
	<SIZE[2]>	000	0	0	1	0	0	0x5E003040
	<SIZE[3]>	000	0	0	1	0	1	0x5E003050
	<SIZE[4]>	000	0	0	1	1	0	0x5E003060
	<SIZE[5]>	000	0	0	1	1	1	0x5E003070

3.2. フローチャート

コードフラッシュの例を以下に示します。

3.2.1. 自動プログラム

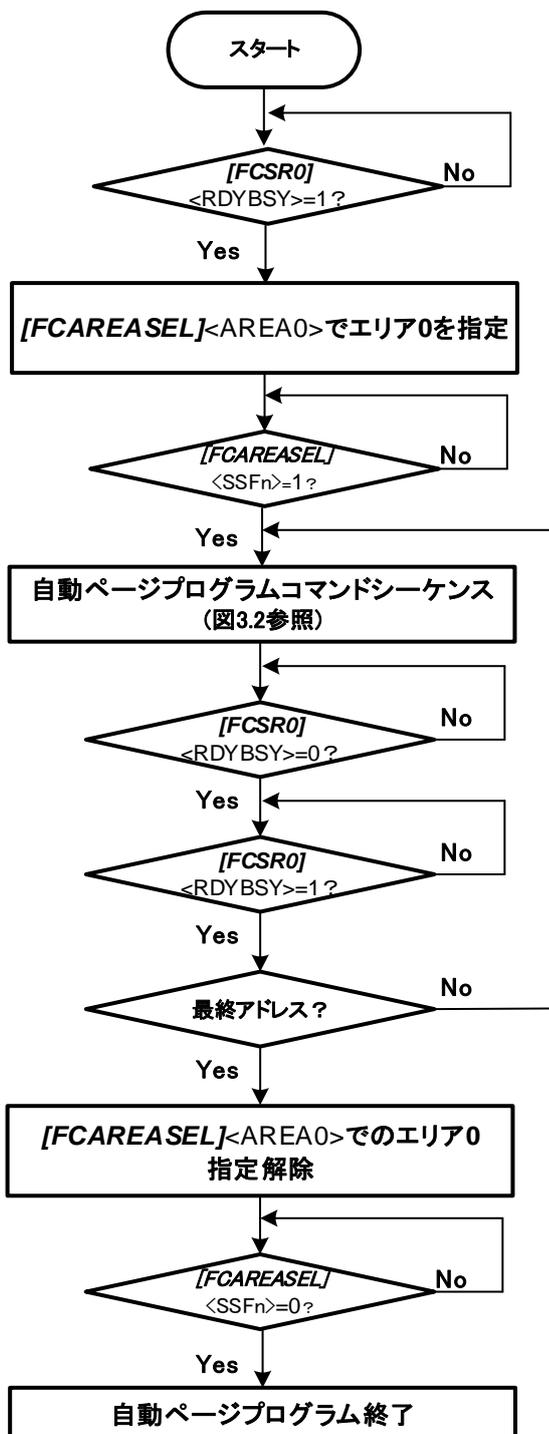


図 3.1 自動プログラムのフロー(1)

自動プログラムコマンドシーケンス(アドレス/コマンド)

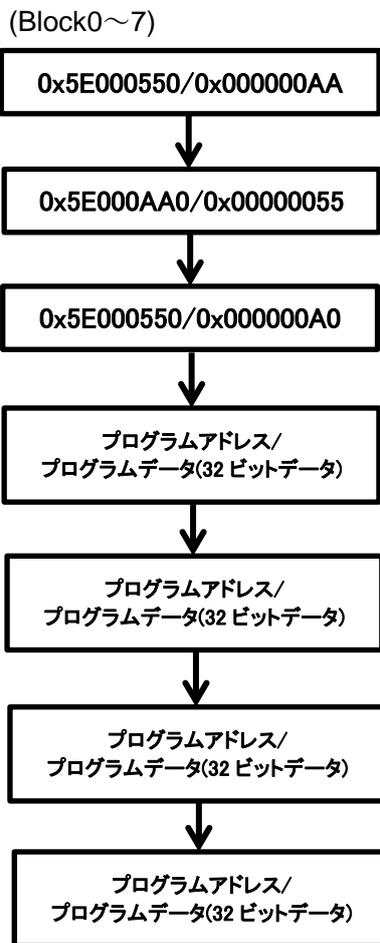


図 3.2 自動プログラムのフロー(2)

3.2.2. 自動消去

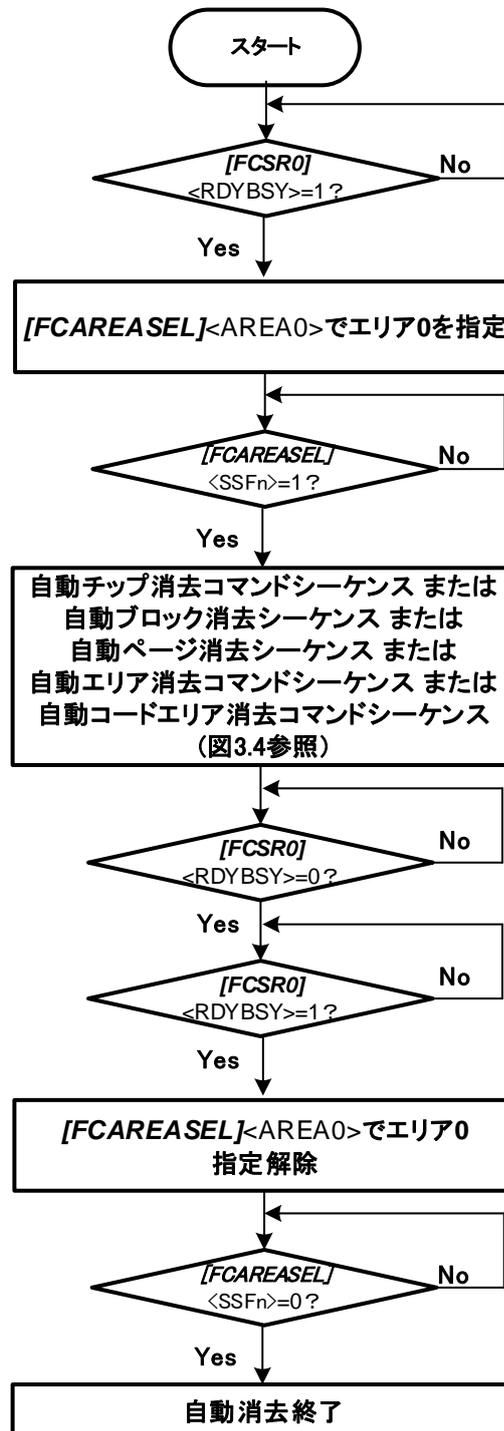


図 3.3 自動消去のフロー(1)

注) 自動消去後はblankチェックでデータ消去されていることを確認してください。

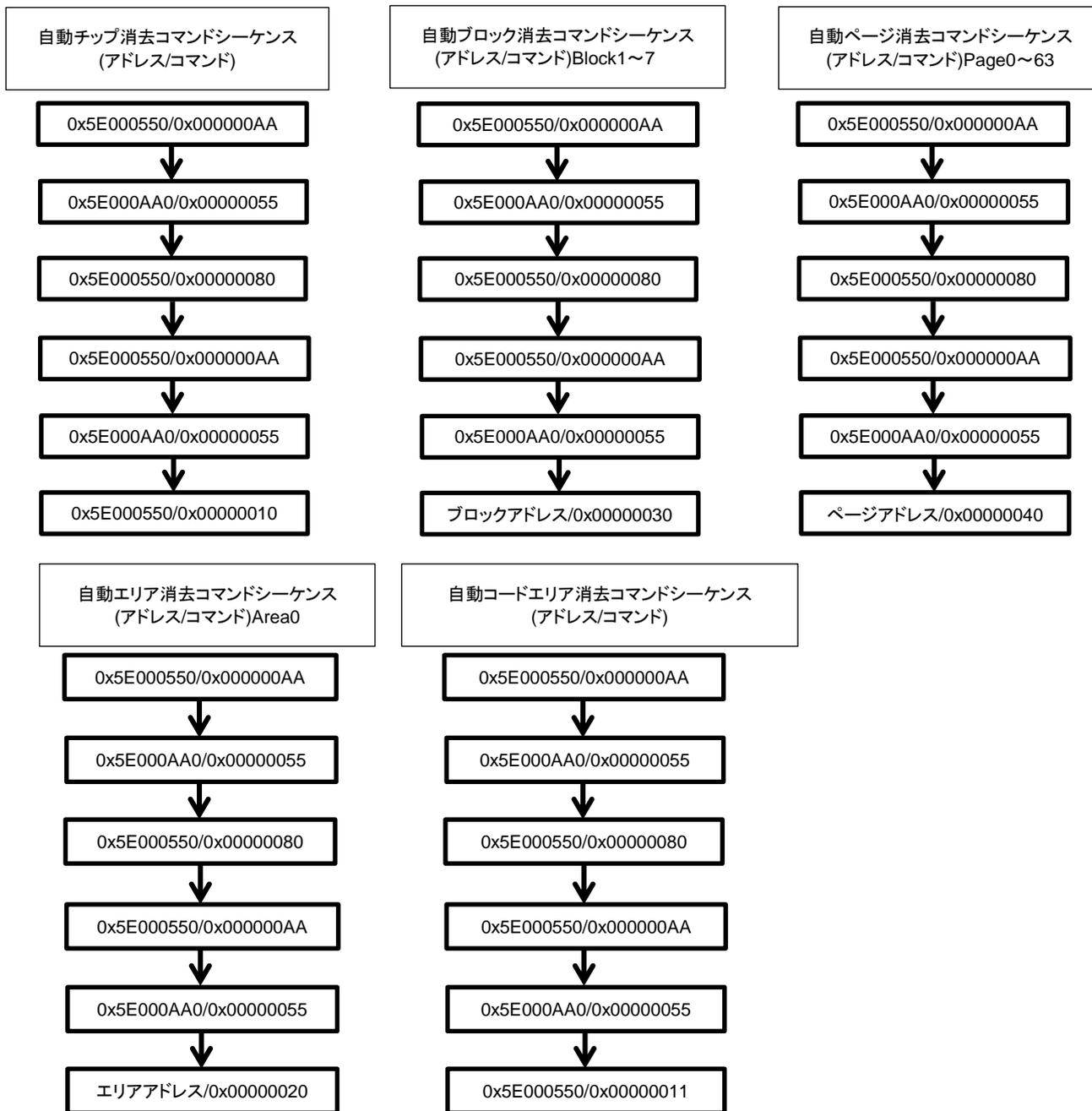


図 3.4 自動消去のフロー(2)

3.2.3. プロテクトビット

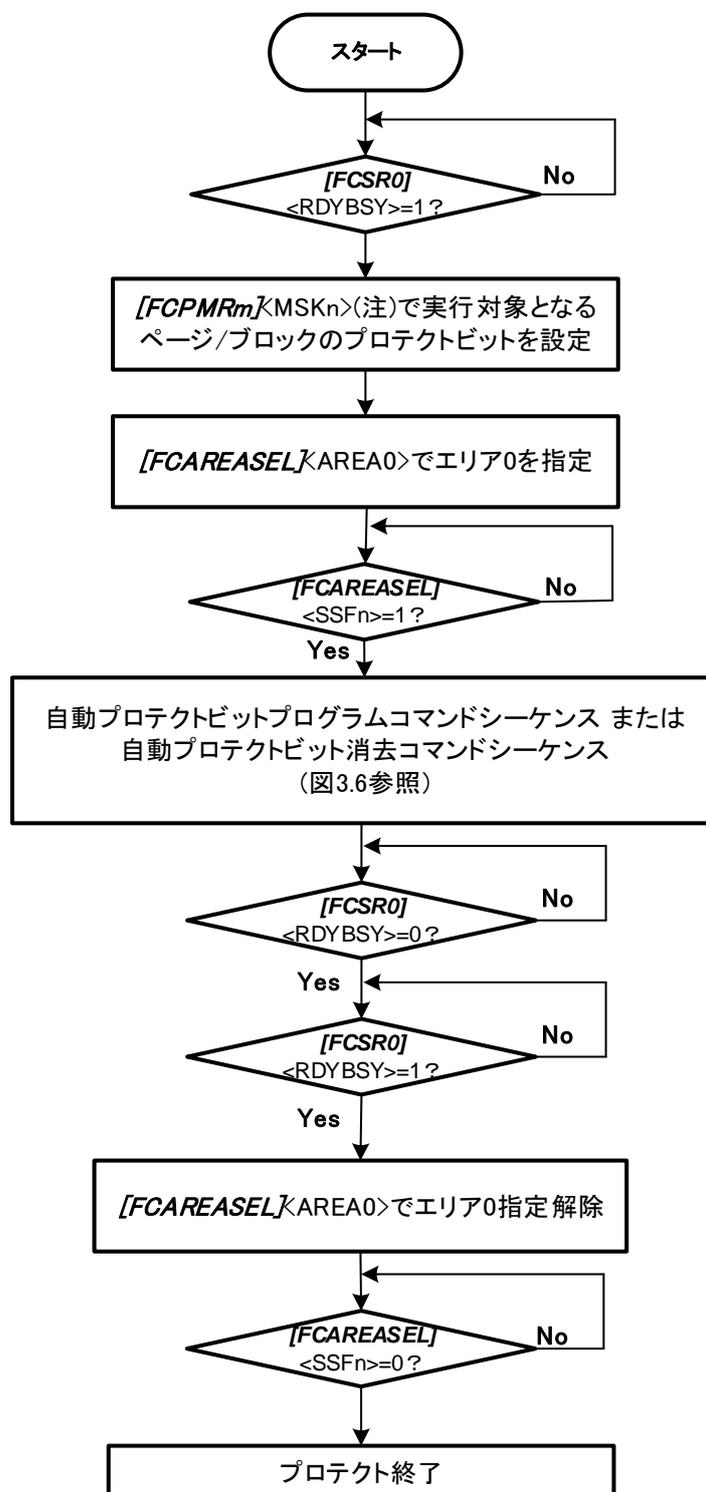


図 3.5 プロテクトのフロー(1)

注) <PMn>、<MSKn> は代表して<MSKn>で表します。

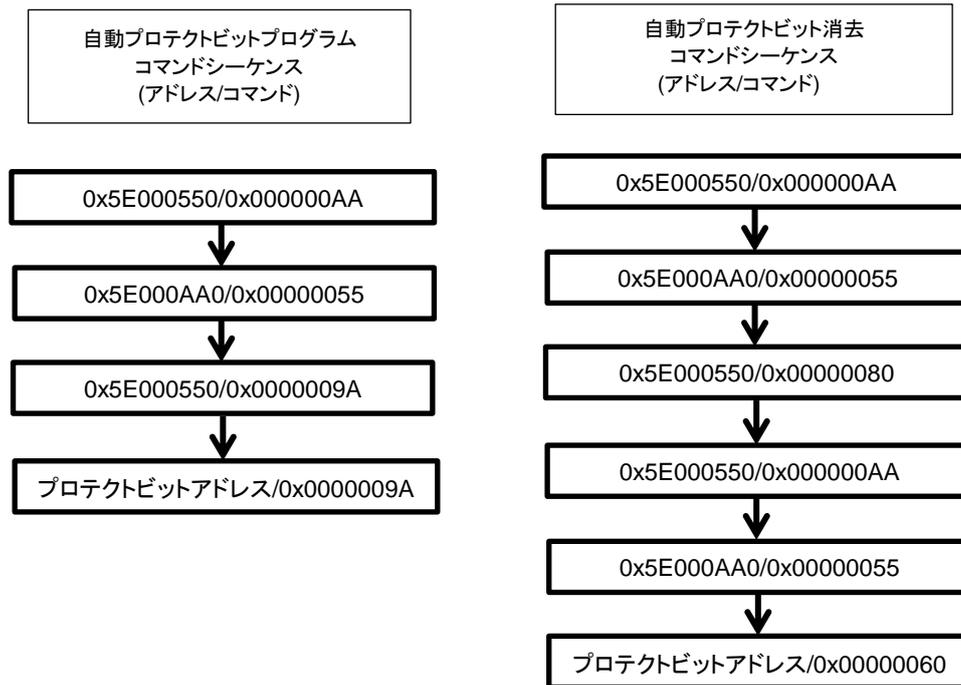


図 3.6 プロテクトのフロー(2)

3.2.4. セキュリティビット

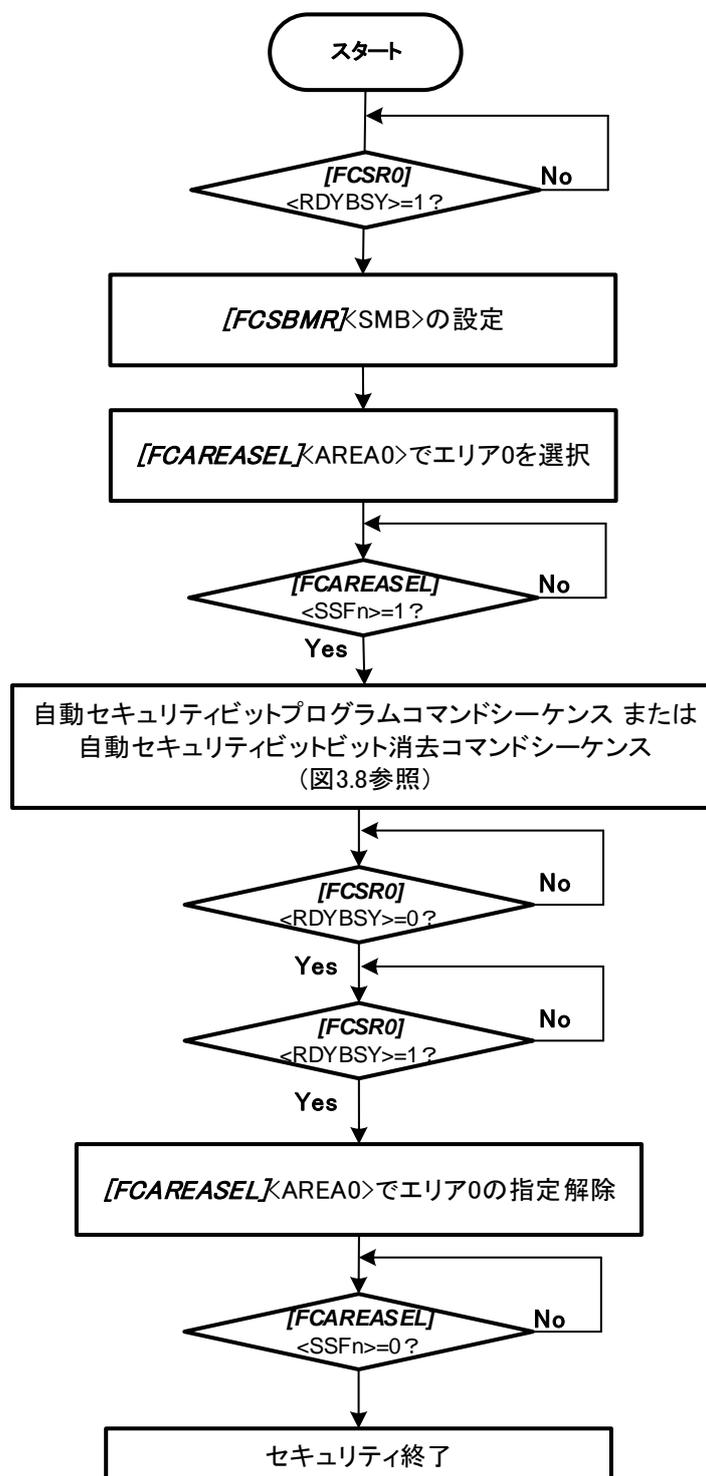


図 3.7 セキュリティのフロー(1)

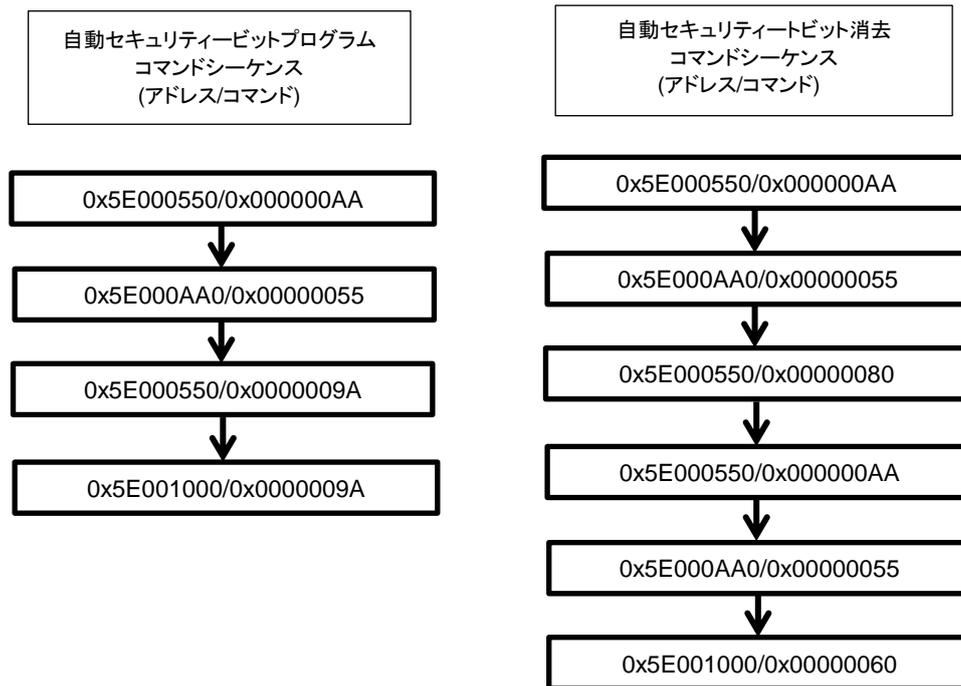


図 3.8 セキュリティーのフロー(2)

3.2.5. メモリスワップ

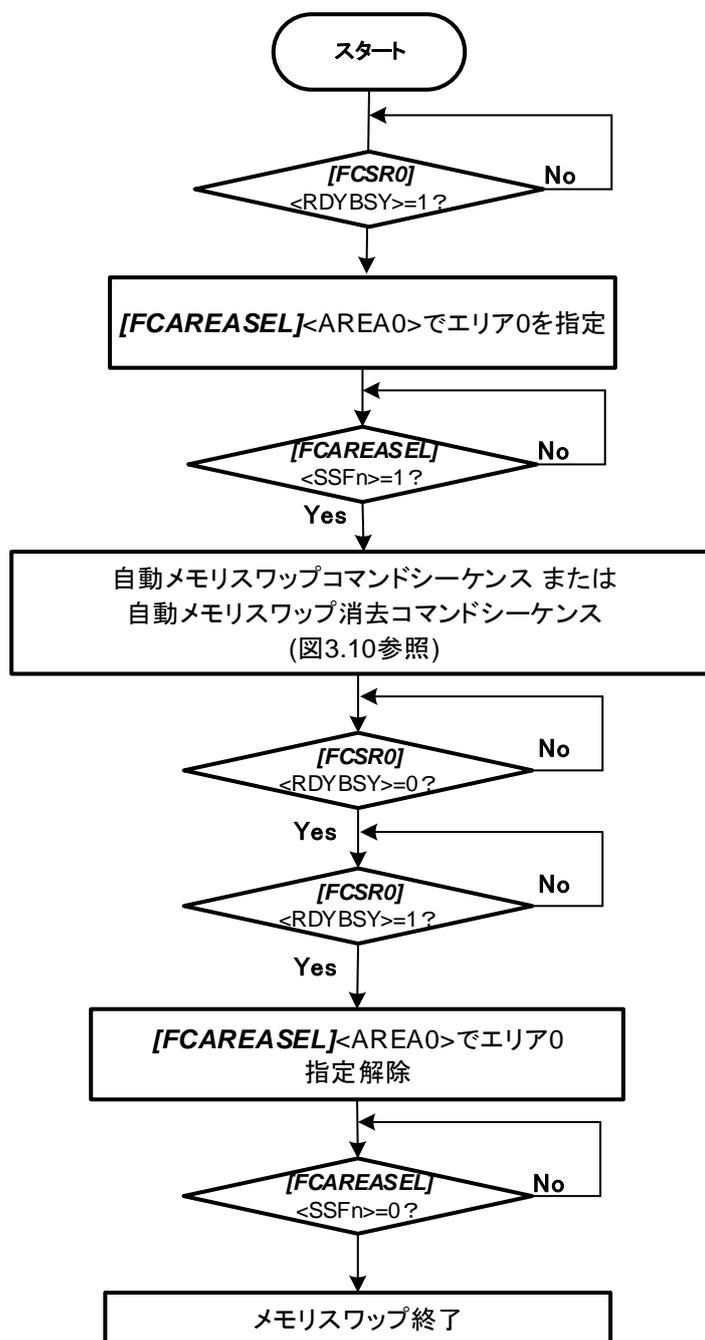


図 3.9 メモリスワップのフロー(1)

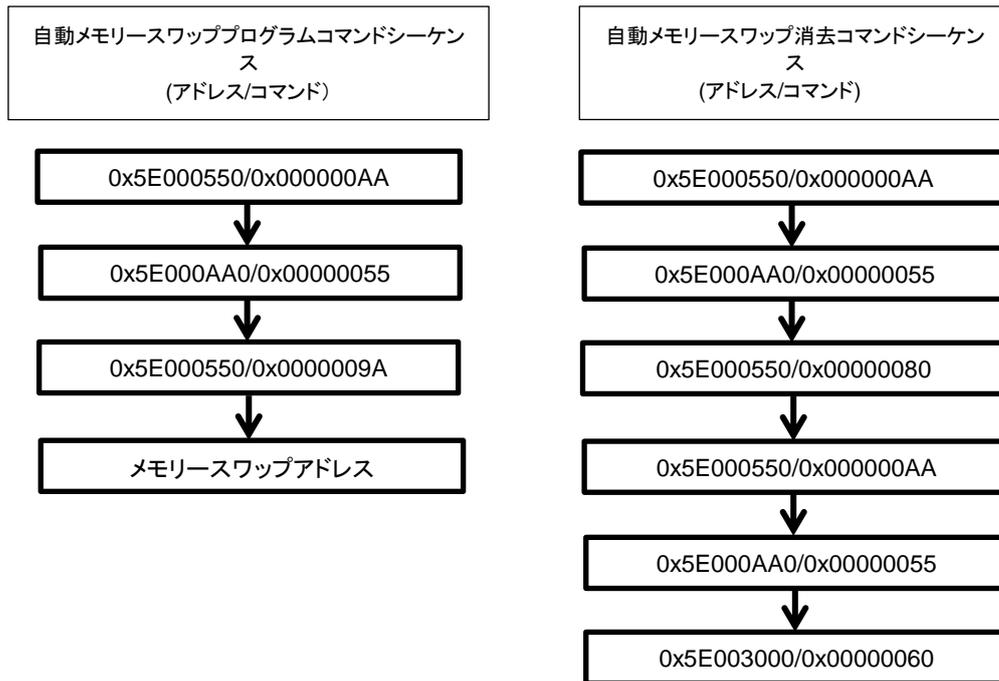


図 3.10 メモリスワップのフロー(2)

4. フラッシュメモリー詳細

制御プログラム中でコマンドを実行することで、フラッシュメモリーの書き込み/消去を行います。この書き込み/消去の制御プログラムはユーザーがあらかじめ用意しておきます。

4.1. 機能

一部の機能を除き、フラッシュメモリーの書き込みおよび消去などはJEDEC標準コマンドに準拠していますが、動作コマンドのアドレス指定が標準コマンドとは異なります。

書き込み、消去を行う場合、32ビット(1ワード)のストアー命令を用いてフラッシュメモリーへコマンドを入力します。コマンド入力後、書き込みおよび消去は内部で自動的に行われます。

表 4.1 フラッシュメモリーの機能

主な機能	説明
自動プログラム	コードフラッシュ:4ワード(16バイト)でデータ書き込みを自動で行います。
自動チップ消去	フラッシュメモリーの全領域の消去を自動で行います。(注1)
自動エリア消去	エリア単位での消去を自動で行います。
自動ブロック消去	ブロック単位での消去を自動で行います。(注2)
自動ページ消去	ページ単位での消去を自動で行います。
自動ライト/消去プロテクト	書き込みおよび消去を禁止することができます。
自動セキュリティービットプログラム/消去	セキュリティーの設定と解除ができます。
自動メモリスワップ/消去	コードフラッシュ領域のスワップ/スワップ解除/スワップサイズ指定を自動で行います。

注1) ユーザーインフォメーションエリアを除く

注2) コードフラッシュのブロック0は一括消去できません。自動ページ消去コマンドでページごとに消去してください。

4.1.1. フラッシュメモリーの動作モード

フラッシュメモリーには、大きく分けて以下の3種類の動作モードがあります。

- ・メモリーデータを読み出すモード(リードモード)
- ・メモリーデータ消去/書き込みコマンド入力モード(コマンドシーケンス入力モード)
- ・メモリーデータを自動的に消去/書き込むモード(自動動作)

電源投入後、リセット解除後、および自動動作の正常終了後エリア指定解除するとフラッシュメモリーはリードモードになります。フラッシュメモリーに書かれた命令の実行、およびデータの読み出しはリードモードで行います。

エリア指定をするとコマンド入力が可能になり、コマンドを入力すると自動動作へ移り、コマンドの処理が正常終了するとコマンドシーケンス入力モードに戻ります。コマンドシーケンス入力モード、自動動作中は、フラッシュメモリーデータの読み出しとフラッシュメモリー上の命令の実行ができません。

4.1.2. コマンド実行方法

コマンド実行は、エリア指定後ストアー命令を用いてフラッシュメモリーに対してコマンドシーケンスを書き込むことで行います。フラッシュメモリーは、入力されたアドレスとデータの組み合わせによって各自動動作コマンドを実行します。コマンド実行の詳細は、「4.1.3 コマンド説明」を参照してください。

フラッシュメモリーに対するストアー命令の実行を"バスライトサイクル"と呼びます。各コマンドは幾つかのバスライトサイクルで構成されています。フラッシュメモリーは、バスライトサイクルのアドレスとデータが規定の順番で実行されたときはコマンドの自動動作を実施します。規定の順番で実行されなかった場合、フラッシュメモリーはコマンドの実行を中止します。

コマンドシーケンスの途中でキャンセルしたい場合(注)や、間違ったコマンドシーケンス(未定義)を入力した場合は、Read/リセットコマンドを実行後コマンドシーケンスの入力モードに戻ります。その後エリア指定解除するとフラッシュメモリーはリードモードになります。

注) キャンセルは自動プログラムコマンドの第3バスサイクルまでに、その他コマンドは最終バスサイクルまでに行ってください。

コマンドシーケンスの書き込みが終了すると自動動作を開始し、 $[FCSR0]<RDYBSY>=0$ になります。自動動作が正常終了したときに $[FCSR0]<RDYBSY>=1$ となります。

自動動作中は、新たなコマンドシーケンスを受け付けません。
コマンドを実行するには以下の事項に留意してください。

1. 自動動作中は以下の操作を行わないでください。
 - ・電源遮断
 - ・全ての例外発生 (推奨)
2. コマンドシーケンサーがコマンドを認識するために、コマンド開始前の状態がリードモードである必要があります。エリア選択をしてコマンドシーケンス入力モードに移行する前に $[FCSR0]<RDYBSY>=1$ であることを確認してください。続いてエリア選択、Read/リセットコマンドを実行してください。

3. 以下のコマンドシーケンスは、内蔵 RAM 上で実行してください。
- ・自動チップ消去コマンド
 - ・ID-Read コマンド
 - ・自動セキュリティービットプログラムコマンド
 - ・自動セキュリティービット消去コマンド
 - ・自動プロテクトビットプログラムコマンド
 - ・自動プロテクトビット消去コマンド
 - ・自動メモリースワップコマンド
 - ・自動メモリースワップ消去コマンド
4. 各コマンドを実行する前に[FCAREASEL]レジスターの対象のエリア選択ビットを設定 (<AREAn>に"111"をライト) してください。
なお、下記コマンドを実行する場合は全てのエリア選択ビットを設定してください。
- ・自動チップ消去コマンド
 - ・ID-Read コマンド
 - ・自動セキュリティービットプログラムコマンド
 - ・自動セキュリティービット消去コマンド
 - ・自動プロテクトビットプログラムコマンド
 - ・自動プロテクトビット消去コマンド
 - ・自動メモリースワップコマンド
 - ・自動メモリースワップ消去コマンド
5. 各バスライトサイクルは連続して、1ワード(32ビット)のデータ転送命令で行います。
6. 各コマンドシーケンスの実行中に、実行対象となるフラッシュメモリーへのアクセスを行うとバスフォールトが発生します。
7. コマンド発行時、誤ったアドレスやデータをライトした場合は、必ずソフトウェアリセット後コマンドシーケンス入力モードに戻してください。
8. 各コマンド実行の終了確認手順は以下のとおりです。
- 1) 最終バスライトサイクルを実行します。
 - 2) [FCSR0]<RDYBSY>=0 (Busy)となるまでポーリングします。
 - 3) [FCSR0]<RDYBSY>=1 (Ready)となるまでポーリングします。
9. フラッシュメモリーからデータをリードする場合は、[FCAREASEL] レジスターのエリア選択ビットをクリア(<AREAn>に"000"をライト) してください。

4.1.3. コマンド説明

各コマンドの内容について説明します。具体的なコマンドシーケンスは「3.1.1 コードフラッシュのコマンドシーケンス」を参照してください。

4.1.3.1. 自動プログラム

(1) 動作内容

自動プログラムコマンドシーケンスにより、コードフラッシュは4ワード(16バイト)単位で書き込みができます。16バイトをまたがってデータを書き込むことはできません。

フラッシュメモリーへの書き込みは、"1"データセルを "0"データにすることです。"0"データセルを "1"データにすることはできません。"0"データセルを "1"データにするには消去動作を行う必要があります。

自動プログラムは消去後のページに対して1回のみ可能で、"1"データセルであっても"0"データセルであっても2回以上の実行はできません。一度書き込み動作を行ったアドレスに対して再度書き込みを行う場合は、自動ページ消去、自動ブロック消去または自動チップ消去コマンドを行った後に自動プログラムを実行しなおす必要があります。

自動プログラム中は、新たにコマンドシーケンスを受け付けません。
自動プログラム実行後はコマンドシーケンス入力モードに戻ります。

注1) 消去動作を伴わない同一アドレスへの2回以上プログラム実施はデータ破損の可能性がります。

注2) プロテクトされたブロックへの書き込み/消去はできません。

(2) 実行方法

第1～第3バスライトサイクルが自動プログラムのコマンドシーケンスです。

第4バスライトサイクル以降にページの先頭アドレスとデータを書き込みます。コードフラッシュでは、第5バスライトサイクルは4ワードの内の残りのデータを書き込みます。

コードフラッシュの16バイトの一部に書き込みを行う場合、書き込みが不要なアドレスのデータを"0xFFFFFFFF"として16バイト分の書き込みを行ってください。

4.1.3.2. 自動チップ消去

(1) 動作内容

自動チップ消去は、コードフラッシュの全アドレスのメモリーセルに対して消去動作を行います。プロテクトされているページまたはブロックがある場合は自動チップ消去を実行せず(注 1)、プロテクトされていないページまたはブロックを消去した後にコマンドシーケンス入力モードに戻ります。

消去対象：コードフラッシュ

プロテクトビットは消去されませんので、消去が必要な場合は自動プロテクトビット消去コマンドで消去してください。

自動消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「4.1.4 自動チップ消去動作の中止」を参考に中止してください。この場合、データの消去は正常に行われていない可能性があるため、あらためて自動チップ消去を実行する必要があります。

(2) 実行方法

第1～第6バスライトサイクルが自動チップ消去のコマンドシーケンスです。コマンドシーケンス入力後、自動チップ消去動作を行います。

- 注 1) プロテクトされているブロックまたはページがある場合は、フラッシュメモリー内部ではページ単位で消去動作を繰り返しますので、消去動作が完了するまでには消去するページ数分の時間がかかります。
- 注 2) 自動チップ消去を連続して実行することはできません。チップ消去を再実行する場合は、一度ブランクチェックを行った後に行ってください。

4.1.3.3. 自動エリア消去

(1) 動作内容

自動エリア消去コマンドは、指定されたエリアに対して消去動作を行います。プロテクトされているページまたはブロックがある場合は自動エリア消去を実行せず(注 1)、プロテクトされていないページまたはブロックを消去した後にコマンドシーケンス入力モードに戻ります。

自動エリア消去中は、新たにコマンドシーケンスを受け付けません。
自動エリア消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

第1～第5バスライトサイクルが自動エリア消去のコマンドシーケンスです。第6バスライトサイクルで消去するエリアを指定します。コマンドシーケンス入力後、自動エリア消去動作を行います。

- 注 1) プロテクトされているブロックまたはページがある場合は、フラッシュメモリー内部ではページ単位で消去動作を繰り返しますので、消去動作が完了するまでには消去するページ数分の時間がかかります。
- 注 2) 自動エリア消去を連続して実行することはできません。エリア消去を再実行する場合は、一度ブランクチェックを行った後に行ってください。

4.1.3.4. 自動ブロック消去

(1) 動作内容

自動ブロック消去コマンドは、指定されたブロックに対する消去動作を行います。指定されたブロックがプロテクトされている場合は、コマンドは実行しないでコマンドシーケンス入力モードに戻ります。

自動ブロック消去中は、新たにコマンドシーケンスを受け付けません。
自動ブロック消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

第1～第5バスライトサイクルが自動ブロック消去のコマンドシーケンスです。第6バスライトサイクルで消去するブロックを指定します。コマンドシーケンス入力後、自動ブロック消去動作を行います。

4.1.3.5. 自動ページ消去

(1) 動作内容

自動ページ消去コマンドは、指定されたページに対する消去動作を行います。指定されたページがプロテクトされている場合は消去を実行せず、コマンドシーケンスの入力後にコマンドシーケンス入力モードに戻ります。

自動ページ消去中は、新たにコマンドシーケンスを受け付けません。
自動ページ消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

第1～第5バスライトサイクルが自動ページ消去のコマンドシーケンスです。第6バスライトサイクルで消去するページを指定します。コマンドシーケンス入力後、自動ページ消去動作を行います。

4.1.3.6. 自動プロテクトビットプログラム

(1) 動作内容

自動プロテクトビットプログラムは、プロテクトビットにビット単位で"1"を書き込みます。プロテクトビットを"0"にするためには自動プロテクトビット消去コマンドを使用します。

プロテクトの機能については「4.1.6 プロテクト機能」を参照してください。

自動プロテクトビットプログラム中は、新たにコマンドシーケンスを受け付けません。
自動プロテクトビットプログラム実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

第1～第3バスライトサイクルが自動プロテクトビットプログラムのコマンドシーケンスになります。第4バスライトサイクルで書き込むプロテクトビットを指定します。コマンドシーケンス入力後、自動プロテクトビットプログラム動作を行います。正しく書き込みができたかどうか、[FCPSRn]レジスタの各ビットを確認してください。

4.1.3.7. 自動プロテクトビット消去

(1) 動作内容

自動プロテクトビット消去コマンドは、実行する際のセキュリティーの状態に関係無くプロテクトビットを消去することができます。

プロテクトの機能については、「4.1.6 プロテクト機能」を参照してください。

自動プロテクトビット消去中は、新たにコマンドシーケンスを受け付けません。
自動プロテクトビット消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

コマンドシーケンス"自動プロテクトビット消去"を入力します。コマンドシーケンス入力後、自動動作を行います。

全てのプロテクトビットが一括消去されます。正常に消去が行われたか、**[FCPSRn]**レジスターの各ビットを確認してください。

4.1.3.8. 自動セキュリティービットプログラム

(1) 動作内容

自動セキュリティービットプログラムは、セキュリティービットに"1"を書き込みます。セキュリティービットを"0"にするためには自動セキュリティービット消去コマンドを使用します。

セキュリティーの機能については「4.1.7. セキュリティー機能」を参照してください。

自動セキュリティービットプログラム中は、新たにコマンドシーケンスを受け付けません。
自動セキュリティービットプログラム実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

コマンドシーケンス"自動セキュリティービットプログラム"を入力します。コマンドシーケンス入力後、自動セキュリティービットプログラム動作を行います。システムリセット後、セキュリティーは有効になります。セキュリティーが有効になると、デバッグツールは接続できなくなります。

4.1.3.9. 自動セキュリティービット消去

(1) 動作内容

自動セキュリティービット消去コマンドは、セキュリティービットを消去します。
実行する際のセキュリティーの状態によって動作内容が異なります。

- セキュリティー状態でない場合
セキュリティービットを"0"にクリアします。
- セキュリティー状態の場合
フラッシュメモリーの全アドレスのデータを消去した後、セキュリティービットを消去します。

セキュリティーの機能については、「4.1.7. セキュリティー機能」を参照してください。

自動セキュリティービット消去中は、新たにコマンドシーケンスを受け付けません。
自動セキュリティービット消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

コマンドシーケンス"自動セキュリティービット消去"を入力します。コマンドシーケンス入力後、自動動作を行います。

セキュリティー状態でない場合、セキュリティービットが消去されます。システムリセット後、セキュリティーは解除されます。正常に消去が行われたか、**[FCSSR]<SEC>**を確認してください。

セキュリティー状態でコマンドシーケンス"自動セキュリティービット消去"を実行した場合、コードフラッシュの全アドレスのデータ消去およびセキュリティービットが消去されます。システムリセット後、セキュリティーは解除されます。データの消去およびセキュリティービットの消去が正常に行われたか読み出して確認してください。さらに必要に応じてコマンドシーケンス"自動プロテクトビット消去"を実行してプロテクトビットを消去してください。

4.1.3.10. ID-Read

(1) 動作内容

ID-Read コマンドは、フラッシュメモリーのタイプなどの情報を読み出すことができます。読み出せる内容は、メーカーコード、デバイスコード、マクロコードの3種類です。

(2) 実行方法

第1～第3バスライトサイクルがID-Readのコマンドシーケンスになります。第4バスライトサイクルでIDアドレスを指定します。第4バスライトサイクルが終わったら、エリア選択を解除し、リードモードにしてから第5バスサイクルでフラッシュ領域からのリード動作でIDデータが得られます。

異なる種類のIDをリードする場合は、第1バスサイクルからコマンドシーケンスを再実行してください。

注) ID-Read 実行後は必ず Read/リセットコマンドを実行してください。

4.1.3.11. Read/リセットコマンド

(1) 動作内容

フラッシュメモリーをコマンドシーケンス入力モードに戻すコマンドです。

(2) 実行方法

Read/リセットコマンドでは第1バスライトサイクルがコマンドシーケンスになります。
コマンドシーケンス実行後、フラッシュメモリーはコマンドシーケンス入力モードになります。

4.1.3.12. 自動メモリスワップ

(1) 動作内容

自動メモリスワップは、**[FCSWPSR]<SWP[1:0]><SIZE[5:0]>**の各ビットにビット単位で"1"を書き込むコマンドです。各ビットを"0"にすることはできず、自動メモリスワップ消去コマンドを使用して全てのビットを"0"クリアします。

自動メモリスワップ動作中は、新たにコマンドシーケンスを受け付けません。
自動メモリスワップ実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

第1～第4バスライトサイクルが自動メモリスワップのコマンドシーケンスです。コマンドシーケンス入力後、**[FCSWPSR]**レジスタの指定ビットに"1"が書き込まれます。正しく書き込みができたかどうか、**[FCSWPSR]<SWP[1:0]><SIZE[5:0]>**を確認してください。

4.1.3.13. 自動メモリスワップ消去

(1) 動作内容

自動メモリスワップ消去は、**[FCSWPSR]<SWP[1:0]><SIZE[5:0]>**の各ビットを一括で消去します。

自動メモリスワップ消去動作中は、新たにコマンドシーケンスを受け付けません。
自動メモリスワップ消去実行後はコマンドシーケンス入力モードに戻ります。

(2) 実行方法

コマンドシーケンス"自動メモリスワップ消去"を入力します。コマンドシーケンス入力後、自動動作を行います。正常に消去が行われたか、**[FCSWPSR]<SWP[1:0]><SIZE[5:0]>**を確認してください。

4.1.4. 自動チップ消去動作の中止

自動チップ消去を実行中にこれを中止する必要がある場合、以下の手順で自動チップ消去動作を中止して、リードモードへ復帰します。

1. $[FCSR0]<RDYBSY>$ をリードします。
2. 手順1のリード結果が"1"(Ready)の場合は、9で終了してください。"0"(Busy)の場合は3へ進みます。
3. $[FCCR]<WEABORT>$ に"0x7"をライトします。
4. $[FCCR]<WEABORT>$ に"0x0"をライトします。
5. $[FCSR0]<RDYBSY>=1$ (Ready) となるまでポーリングします。
6. $[FCSRI]<WEABORT>$ をリードします。
7. Read/リセットコマンドを発行します。
8. 手順6のリード結果が"0"の場合は、9で終了してください。"1"の場合は以下の操作を行い、このフラグをクリアします。
 - 1) $[FCSTSCLR]<WEABORT>$ に"0x7"をライトします。
 - 2) $[FCSTSCLR]<WEABORT>$ に"0x0"をライトします。
 - 3) $[FCSRI]<WEABORT>=0$ となるまでポーリングします。
9. 終了

注) $[FCCR]$ への書き込み前に $[FCKCR]$ への特定コード書き込みが必要ですが、上記手順では省略して書いてあります。

4.1.5. 自動動作の完了検知

フラッシュの書き込み/消去動作などの完了を検知する割り込み機能があります。

表 4.2 フラッシュの書き込み/消去動作などの完了検知

	信号名	割り込み名称
コードフラッシュの書き込み/消去動作の完了	INTFLCRDY	Code FLASH Ready 割り込み

4.1.5.1. 手順

自動動作の完了検知割り込みを使用する手順は以下のとおりです。
割り込み処理の詳細については、リファレンスマニュアル「例外」の章"割り込み"を参照してください。

1. RAM 上に消去/書き込みプログラムを配置します。割り込み用のベクターテーブルおよび割り込みサービスルーチンも同様に RAM 上に配置します。
2. INTFLCRDY 割り込みの許可の設定をします。
3. コードフラッシュに対し書き込み/消去コマンドを発行後、 $[FCSR0]<RDYBSY>=0$ で自動動作中(BUSY 状態)を確認します。
4. フラッシュの自動動作終了後、INTFLCRDY 割り込みが発生します。
5. 継続して書き込みを行わない場合は、割り込み処理ルーチンの中で、INTFLCRDY 割り込みの禁止をしてからリターンしてください。継続して書き込みを行う場合は、INTFLCRDY 割り込みは禁止しないで新しいコマンドシーケンスを発行してからリターンしてください。
6. 書き込みを継続する場合は、3~5を繰り返します。

注) コードフラッシュ上のプログラムでコードフラッシュの消去/書き込みはできません。

4.1.6. プロテクト機能

コードフラッシュではページ0~7はページ単位で行い、残りのブロックはブロック単位で行います。プロテクト設定の消去は、一括消去となります。

4.1.6.1. プロテクトの設定方法

プロテクト機能は、フラッシュメモリーへの書き込み、消去をブロック単位で禁止することができます。プロテクト機能が有効となる条件は以下のとおりです。

1. $[FCPMRm] \langle MSKn \rangle = 1$ (注)
2. プロテクトビット $n=1$

この時ブロック n が書き込み／消去禁止状態となります。

プロテクトビットの状態を確認する場合は、 $[FCPMRm] \langle MSKn \rangle = 1$ にして $[FCPSRm]$ をモニターしてください。(注)

注) $\langle PMn \rangle$ 、 $\langle MSKn \rangle$ は代表して $\langle MSKn \rangle$ で表します。

4.1.6.2. プロテクトの解除方法

プロテクトビット消去コマンドによりプロテクトビットを"0"にすることでブロックプロテクトは解除されます。

注)プロテクトビットはプロテクトビット消去コマンドにより全ビット"0"になります。

4.1.6.3. プロテクトの一時解除機能

プロテクトビットを消去しないでプロテクト機能を一時解除することができます。指定したブロックのみ解除することができます。

$[FCPMRm] \langle MSKn \rangle = 0$ の時、当該のブロック n に対するプロテクトビットの状態にかかわらず書き込み、消去禁止機能は無効となります。

レジスター設定は「5.2 レジスターの詳細」の $[FCPMRm]$ を参照してください。

注) $\langle PMn \rangle$ 、 $\langle MSKn \rangle$ は代表して $\langle MSKn \rangle$ で表します。

4.1.7. セキュリティー機能

セキュリティー機能は、フラッシュライターによるフラッシュメモリーの読み出しの禁止およびデバッグ機能の使用制限をすることができます。

4.1.7.1. セキュリティーの設定方法

セキュリティー機能を有効にするには、セキュリティービットプログラムコマンドでセキュリティービットを"1"にします。セキュリティー機能が有効となる条件は以下のとおりです。

1. $[FCSBMR]<SMB>=1$
2. セキュリティービット=1

セキュリティービットの状態を確認する場合は、 $[FCSBMR]<SMB>=1$ にして $[FCSSR]<SEC>$ をリードしてください。

注) セキュリティービット書き込み後は、システムリセットでセキュリティーは有効になります。

4.1.7.2. セキュリティーの解除方法

セキュリティー機能を解除するには以下の手順で行います。

1. $[FCSBMR]<SMB>=0$ にします。
2. セキュリティービット消去コマンドでセキュリティービットを"0"に消去します。

$[FCSBMR]<SMB>=1$ 、 $[FCSSR]<SEC>=1$ の状態で、セキュリティービット消去コマンドを実行すると、チップ消去機能が起動してコードフラッシュおよびセキュリティービットが消去されます。

注) セキュリティービット消去後は、システムリセットでセキュリティーは解除されます。

4.1.7.3. 動作

セキュリティーが有効な場合の動作を表 4.3 に示します。

表 4.3 セキュリティー有効時の動作

項目	内容
フラッシュメモリーの読み出し	CPU からの読み出しは可能です。
デバッグモード	デバッグできなくなります。
フラッシュライターモード	フラッシュメモリーの読み出し、書き込みはできません

4.1.8. メモリースワップ機能

コードフラッシュの書き替え操作の途中で中断した場合、例えばプログラム消去後に電源がOFFになって書き込みを継続できなくなるケースが考えられます。このようなケースを回避するために、本機能を利用して書き込みプログラムを残すことができます。

4.1.8.1. メモリースワップの設定方法

スワップ領域の対象は 0 番地で始まる領域と続く同じサイズの領域です。スワップサイズは **[FCSWPSR]<SIZE>** で決まります。このサイズを変更するには、自動メモリースワッププログラムコマンドで **[FCSWPSR]<SIZE>** の対象サイズのビットを"1"にセットします。

メモリースワップを行うには、自動メモリースワッププログラムコマンドで **[FCSWPSR]<SWP[0]>** に"1"を設定します。スワップ状態を解除するには、自動メモリースワップコマンドで **[FCSWPSR]<SWP[1]>** に"1"を設定します。スワップ状態は **[FCSWPSR]<SWP>** で確認することができます。

自動メモリースワップコマンドの詳細は「4.1.3.12 自動メモリースワップ」の章を参照してください。

4.1.8.2. 操作方法

メモリースワップ操作の基本的な流れを以下に示します。メモリースワップ操作の具体例は「6.7 ユーザーブートプログラムの書き替え方法」を参照してください。

プロテクト機能が有効の場合は、プロテクトを一時解除してください。

プロテクトの一時解除方法は「4.1.6.3 プロテクトの一時解除機能」を参照してください。プロテクトの一時解除をしない場合、手順におけるコマンド実行が行われません。

1. 0 番地で始まる領域の次の領域がブランク状態であることを確認します。(以後、0 番地で始まる領域を Page0、次の領域を Page1 として説明します。)ブランク状態でなければ消去してください。

Page0 : 旧オリジナルデータ

Page1 : ブランク

2. 0 番地で始まる領域のオリジナルデータを次の領域にも書き込みます。(両方の領域のデータを同じにします)

Page0 : 旧オリジナルデータ

Page1 : コピーデータ(旧オリジナルデータ)

3. メモリースワップを行います。

Page0 : コピーデータ(旧オリジナルデータ)

Page1 : 旧オリジナルデータ

4. 旧オリジナルデータを消去して、ブランク状態にします。

Page0 : コピーデータ(旧オリジナルデータ)

Page1 : ブランク

5. ブランク領域に新しいデータを書き込みます。

Page0 : コピーデータ(旧オリジナルデータ)

Page1 : 新オリジナルデータ

6. スワップ状態を解除します。

Page0 : 新オリジナルデータ

Page1 : コピーデータ(旧オリジナルデータ)

7. 自動プロテクトビット消去コマンドを実行します。

8. 必要により以下を行ってください。

- コピーデータ(旧オリジナルデータ)消去。
- スワップ領域以外のフラッシュメモリーのデータ書き換え。
- プロテクト機能の有効化
- セキュリティー機能の有効化

手順	1	2	3	4	5	6
内蔵 RAM	消去ルーチン	書き替えルーチン	スワップルーチン	消去ルーチン	書き替えルーチン	スワップルーチン
フラッシュメモリー	Page0	旧オリジナル	旧オリジナルのコピー	旧オリジナルのコピー	旧オリジナルのコピー	新オリジナル
	Page1	ブランク	旧オリジナルのコピー	旧オリジナル	ブランク	旧オリジナルのコピー

消去ルーチン: フラッシュメモリーの消去を行うためのプログラム

書き替えルーチン: フラッシュメモリーの書き替えを行うためのプログラム

スワップルーチン: フラッシュメモリーのスワップを行うためのプログラム

図 4.1 メモリースワップ実施手順の例

4.1.8.3. メモリースワップ情報の消去

メモリースワップ状態を解除後、再びメモリースワップを行うには、自動メモリースワップ消去コマンドで[FCSWPSR]のレジスターを全て"0"に初期化してください。

4.1.9. ユーザーインフォメーションエリア

ユーザーインフォメーションエリアでは命令実行はできません。データリードはCPUから命令で行います。

[FCBNKCR]によるバンク切替えによってアクセスが可能となります。アドレス配置などは「表 2.8 コードフラッシュのユーザーインフォメーションエリア」を参照してください。切替後は、コードフラッシュ(エリア 0)はアクセスしないでください。

チップ消去コマンドでは消去されませんので、チップごとにユニークな管理番号などを書き込んでおくことができます。

コードフラッシュ(エリア 0)と同時に使用できません。排他的に使用してください。

4.1.9.1. ユーザーインフォメーションエリアの切替え手順

- (1) RAM 上切替えプログラムを転送し、Jump します
- (2) [FCBUFDISCLR]<BUFDISCLR[2:0]>に"111"を書き込みます
- (3) [FCBNKCR]<BANK0[2:0]>に"111"を書き込みます
- (4) [FCBNKCR]<BANK0[2:0]>をリードして"111"となったことを確認します
- (5) ユーザーインフォメーションエリアを操作します
データリード、データ書き込み、消去
- (6) [FCBNKCR]<BANK0[2:0]>に"000"を書き込みます
- (7) [FCBNKCR]<BANK0[2:0]>をリードして"000"となったことを確認します
- (8) [FCBUFDISCLR]<BUFDISCLR[2:0]>に"000"を書き込みます。
- (9) 元のプログラムへリターンします。

4.1.9.2. ユーザーインフォメーションエリアへの書き込み方法

4.1.9.1 の手順(5)の工程で、コードフラッシュ(エリア 0)への書き込みと同様の手順で書き込みができます。

4.1.9.3. ユーザーインフォメーションエリアの消去方法

4.1.9.1 の手順(5)の工程で、コードフラッシュ(エリア 0)のページ消去と同様の手順で消去ができます。全てが一括で消去されます。

5. レジスター説明

5.1. レジスター一覧

フラッシュメモリー関連のレジスター一覧を示します。

周辺機能		チャンネル/ユニット	ベースアドレス Type1
フラッシュメモリー	FC	-	0x5DFF0000

レジスター名		アドレス(Base+)
Flash Security Bit Mask Register	[FCSBMR]	0x010
Flash Security Status Register	[FCSSR]	0x014
Flash Key Code Register	[FCKCR]	0x018
Flash Status Register 0	[FCSR0]	0x020
Flash Protect Status Register 0	[FCPSR0]	0x030
Flash Protect Status Register 1	[FCPSR1]	0x034
Flash Protect Mask Register 0	[FCPMR0]	0x050
Flash Protect Mask Register 1	[FCPMR1]	0x054
Flash Status Register 1	[FCSR1]	0x0100
Flash Memory SWAP Status Register	[FCSWPSR]	0x0104
Flash Area Selection Register	[FCAREASEL]	0x0140
Flash Control Register	[FCCR]	0x0148
Flash Status Clear Register	[FCSTCLR]	0x014C
Flash Bank Change Register	[FCBNKCR]	0x0150
Flash Buffer Disable and Clear Register	[FCBUFDISCLR]	0x0158

注) レジスター割り当てが無いアドレスにはアクセスしないでください。

5.2. レジスタの詳細

5.2.1. [FCSBMR] Flash Security Bit Mask Register

Bit	Bit Symbol	リセット後	Type	機能
31:1	-	0	R	リードすると"0"が読めます。
0	SMB	1	R/W	セキュリティーマスクビット 1: マスクしません 0: マスクします(セキュリティー一時解除) セキュリティーが有効([FCSSR]<SEC>=1)の時に本レジスターに"0"を書き込むとセキュリティーを一時的に解除します。パワーオンリセットでのみ初期化されます。

注) 本レジスターを書き換える場合は以下の手順で行ってください。

1. [FCKCR]に対して特定のコード(0xA74A9D23)を書き込む。
2. "1"の書き込みから16クロック以内に[FCSBMR]<SMB>のデータを書き換える。

5.2.2. [FCSSR] Flash Security Status Register

Bit	Bit Symbol	リセット後	Type	機能
31:1	-	0	R	リードすると"0"が読めます。
0	SEC	0/1	R	セキュリティーステータス:セキュリティーの状態を示します。 1: セキュリティーがかかっています 0: セキュリティーはかかっていません システムリセットでセキュリティーの状態がロードされます。

5.2.3. [FCKCR] Flash Key Code Register

Bit	Bit Symbol	リセット後	Type	機能
31:0	KEYCODE[31:0]	0x00000000	W	レジスターロック解除用キーコード [FCSBMR]、[FCPMRn]、[FCCR]、[FCAREASEL] を書き換える場合は、事前に本レジスターに対して特定のコード(0xA74A9D23)を書き、その後16クロック以内に当該のレジスター値を書き換えてください。 16クロック以内に有効な書き込みが行われた場合は、許可状態はリセットされます。

5.2.4. [FCSR0] Flash Status Register 0

Bit	Bit Symbol	リセット後	Type	機能
31:16	-	0	R	リードすると"0"が読めます。
15	-	不定	R	リードすると不定値が読めます。
14:11	-	0	R	リードすると"0"が読めます。
10	-	1	R	リードすると"1"が読めます。
9	-	1	R	リードすると"1"が読めます。
8	RDYBSY0	1	R	エリア 0 の ReadyBusy 0: 自動動作中 1: 自動動作終了
7:1	-	0	R	リードすると"0"が読めます。
0	RDYBSY	1	R	全てのフラッシュエリアの ReadyBusy 0: 自動動作中 1: 自動動作終了 自動プログラムまたは自動消去コマンド実行時の Ready/Busy。自動動作の状態を認識できます。フラッシュメモリーが自動動作中は "0" になり、ビジー状態であることを示します。自動動作が終了するとレディー状態となり"1" を出力し、次のコマンドを受け付けます。

5.2.5. [FCPSR0] Flash Protect Status Register 0

Bit	Bit Symbol	リセット後	Type	機能
31:8	-	0	R	リードすると"0"が読めます。
7	PG7	0/1	R	コードフラッシュ(エリア 0)のプロテクトステータス 1: プロテクト状態 0: プロテクト状態ではない Page0 から Page7 まで(Block0)のページごとのプロテクトの状態を示します。該当ビットが"1"の時は対応するページがプロテクト状態であることを示します。プロテクト状態のページは書き換えできません。システムリセットでプロテクトの状態がロードされます。
6	PG6	0/1	R	
5	PG5	0/1	R	
4	PG4	0/1	R	
3	PG3	0/1	R	
2	PG2	0/1	R	
1	PG1	0/1	R	
0	PG0	0/1	R	

5.2.6. [FCPSR1] Flash Protect Status Register 1

Bit	Bit Symbol	リセット後	Type	機能
31:8	-	0	R	リードすると"0"が読めます。
7	BLK7	0/1	R	コードフラッシュ(エリア 0)のプロテクトステータス 1: プロテクト状態 0: プロテクト状態ではない Block1 から Block7 までのブロックごとのプロテクトの状態を示します。該当ビットが"1"の時は対応するブロックがプロテクト状態であることを示します。プロテクト状態のブロックは書き換えできません。システムリセットでプロテクトの状態がロードされます。
6	BLK6	0/1	R	
5	BLK5	0/1	R	
4	BLK4	0/1	R	
3	BLK3	0/1	R	
2	BLK2	0/1	R	
1	BLK1	0/1	R	
0	-	0	R	

5.2.7. [FCPMR0] Flash Protect Mask Register 0

Bit	Bit Symbol	リセット後	Type	機能
31:8	-	0	R	リードすると"0"が読めます。
7	PM7	1	R/W	コードフラッシュ(エリア 0)のプロテクトマスク 1: マスクしません(プロテクトがかかります) 0: マスクします(プロテクトはかかりません) Page0 から Page7 までブロック 0 のページごとにプロテクト状態をマスクします。 システムリセットで初期化されます。
6	PM6	1	R/W	
5	PM5	1	R/W	
4	PM4	1	R/W	
3	PM3	1	R/W	
2	PM2	1	R/W	
1	PM1	1	R/W	
0	PM0	1	R/W	

注) 本レジスターを書き換える場合は以下の手順で行ってください。

- [FCKCR]に対して特定のコード (0xA74A9D23) を書き込む。
- "1"の書き込みから 16 クロック以内に[FCPMR0]<PMn>のデータを書き換える。

5.2.8. [FCPMR1] Flash Protect Mask Register 1

Bit	Bit Symbol	リセット後	Type	機能
31:8	-	1	R/W	"1"を書いてください。
7	MSK7	1	R/W	コードフラッシュ(エリア 0)のプロテクトマスク 1: マスクしません(プロテクトがかかります) 0: マスクします(プロテクトはかかりません) Block1 から Block7 までのブロックごとにプロテクト状態をマスクします。 システムリセットで初期化されます。
6	MSK6	1	R/W	
5	MSK5	1	R/W	
4	MSK4	1	R/W	
3	MSK3	1	R/W	
2	MSK2	1	R/W	
1	MSK1	1	R/W	
0	-	0	R	リードすると"0"が読めます。

注) 本レジスターを書き換える場合は以下の手順で行ってください。

1. [FCKCR]に対して特定のコード (0xA74A9D23) を書き込む。
2. "1"の書き込みから 16 クロック以内に[FCPMR1]<MSKn>のデータを書き換える。

5.2.9. [FCSR1] Flash Status Register 1

Bit	Bit Symbol	リセット後	Type	機能
31:25	-	0	R	リードすると"0"が読めます。
24	WEABORT	0	R	[FCCR]<WEABORT>=111 がセットされると本ビットは"1"がセットされます。
23:0	-	0	R	リードすると"0"が読めます。

5.2.10. [FCSWPSR] Flash Memory SWAP Status Register

Bit	Bit Symbol	リセット後	Type	機能
31:14	-	0	R	リードすると"0"が読めます。
13:8	SIZE[5:0]	000000	R	メモリスワップサイズの設定状態(エリア 0) 000000: スワップしない(初期値) 000001: 4KB (ページ 0 ↔ ページ 1) 000010: 8KB (ページ 0-1 ↔ ページ 2-3) 000100: 16KB (ページ 0-3 ↔ ページ 4-7) 001000: 32KB (ブロック 0 ↔ ブロック 1) 上記以外は設定禁止
7:2	-	0	R	リードすると"0"が読めます。
1:0	SWP[1:0]	00	R	スワップの状態 00: スワップ解除(初期化状態) 01: スワップ中 10: 設定禁止 11: スワップ解除

注 1) メモリスワップを行う場合は場合は、RAM 上のプログラムで行ってください。

注 2) <SWP[1:0]>を"11:スワップ解除"から"00:スワップ解除"の状態に初期化するためには自動スワップ消去コマンドを実行します。この時、スワップサイズ<SIZE[5:0]>も一緒に"000000"に初期化されま
す。この操作は、スワップするメモリーの両方にプログラムが書かれている状態で行ってください。

5.2.11. [FCAREASEL] Flash Area Selection Register

Bit	Bit Symbol	リセット後	Type	機能
31:27	-	0	R	リードすると"0"が読めます。
26	SSF0	0	R	エリア 0 の選択状態(注 1) 1: エリア 0 を選択(コマンドシーケンス入力モード) 0: エリア 0 を非選択(リードモード)
25:23	-	0	R	リードすると"0"が読めます。
22:20	-	000	R/W	"000"を書いてください。
19	-	0	R	リードすると"0"が読めます。
18:16	-	000	R/W	"000"を書いてください。
15	-	0	R	リードすると"0"が読めます。
14:12	-	000	R/W	"000"を書いてください。
11	-	0	R	リードすると"0"が読めます。
10:8	-	000	R/W	"000"を書いてください。
7	-	0	R	リードすると"0"が読めます。
6:4	-	000	R/W	"000"を書いてください。
3	-	0	R	リードすると"0"が読めます。
2:0	AREA0[2:0]	000	R/W	フラッシュメモリー操作コマンドにより実行の対象(コマンドシーケンス入力モード)となるコードフラッシュのエリア 0 を指定します。 (注 1) 111: エリア 0 を選択 上記以外: エリア 0 を非選択

注 1) <AREA0[2:0]>を書き換えた場合は、<SSF0>をリードして書き換えた結果が反映されるまで待つから、次の操作を行ってください。

注 2) 本レジスターを書き換える場合は以下の手順で行ってください。

1. [FCKCR]に対して特定のコード (0xA74A9D23) を書き込む。
2. "1"の書き込みから 16 クロック以内に[FCAREASEL]<AREAn[2:0]>のデータを書き換える。

注 3) 本レジスターを書き換える場合は RAM 上のプログラムで行ってください。

5.2.12. [FCCR] Flash Control Register

Bit	Bit Symbol	リセット後	Type	機能
31:3	-	0	R	リードすると"0"が読めます。
2:0	WEABORT[2:0]	000	R/W	自動チップ消去動作の中止 111: 自動消去動作を中止する 000: 何もしない 上記以外: 使用禁止

注 1) 本レジスターを書き換える場合は以下の手順で行ってください。

1. [FCKCR]に対して特定のコード (0xA74A9D23) を書き込む。
2. "1"の書き込みから 16 クロック以内に[FCCR]<WEABORT>のデータを書き換える。

注 2) 本レジスターを書き換える場合は RAM 上のプログラムで行ってください。

5.2.13. [FCSTSCLR] Flash Status Clear Register

Bit	Bit Symbol	リセット後	Type	機能
31:3	-	0	R	リードすると"0"が読めます。
2:0	WEABORT[2:0]	000	R/W	[FCSR1]<WEABORT>を"0"にクリア。 111: クリアする 上記以外: 何もしない

注) 本レジスターを書き換える場合は RAM 上のプログラムで行ってください。

5.2.14. [FCBNKCR] Flash Bank Change Register

Bit	Bit Symbol	リセット後	Type	機能
31:7	-	0	R	リードすると"0"が読めます。
6:4	-	000	R/W	"000"を書いてください。
3	-	0	R	リードすると"0"が読めます。
2:0	BANK0[2:0]	000	R/W	コードフラッシュ(エリア 0)のアドレス"0x5E005000"~"0x5E005FFF" がユーザーインフォメーションエリアになります。 111: 入れ替わります(ユーザーインフォメーションエリア) 000: 元に戻ります(コードフラッシュ) 上記以外: 何もしない

注 1) BANK0 を操作の前後でコードフラッシュバッファの操作が必要です。「5.2.15 [FCBUFDISCLR] Flash Buffer Disable and Clear Register」を参照してください。

注 2) 本レジスターに値を設定する場合は、本レジスターに値をライトし、その後ライトした値がリードできることを確認してください。

注 3) 本レジスターを書き換える場合は、RAM 上のプログラムで行ってください。

注 4) ユーザーインフォメーションエリアを使用中はコードフラッシュ(エリア 0)の "0x5E005000" ~ "0x5E005FFF" 以外の領域はアクセスしないでください。

5.2.15. [FCBUFDISCLR] Flash Buffer Disable and Clear Register

Bit	Bit Symbol	リセット後	Type	機能
31:3	-	0	R	リードすると"0"が読めます。
2:0	BUFDISCLR[2:0]	000	R/W	<p>コードフラッシュのバッファを停止してクリアします。</p> <p>111: バッファ機能停止、バッファークリア 000: バッファ機能開始 上記以外: 何もしない</p> <p>[FCBNKCR]でコードフラッシュ(エリア 0)とユーザーインフォメーションエリアをバンク切り替える場合には、必ず切り替える前に本レジスターを操作してフラッシュバッファを停止して内容をクリアしてください。さらに、ユーザーインフォメーションエリアに対する操作終了後、必ず"000"を書いてバッファ機能を開始してください。</p>

- 注 1) 本レジスターに値を設定する場合は、本レジスターに値をライトし、その後ライトした値がリードできることを確認してください。
- 注 2) 本レジスターを書き換える場合は、RAM 上のプログラムで行ってください。

6. プログラミング方法

6.1. 初期化

コードフラッシュへの書き込み，消去操作を行う前には，必ず内蔵高速発振器（IHOSC1）を発振させてください。発振開始後 $[CGOSCCR] < IHOSC1F > = 1$ であることを確認してから操作してください。内蔵高速発振器（IHOSC1）および $[CGOSCCR] < IHOSC1F >$ については，リファレンスマニュアル「クロック制御と動作モード」を参照してください。

6.2. モードの説明

シングルチップモードとシングルブートモードがあります。表 6.1 にモードと動作を示します。

表 6.1 モードと動作

モード	動作
シングルブートモード	リセット解除後、内蔵するブート ROM(マスク ROM)のプログラムが起動します。 通信機能を経由して外部から内蔵 RAM に「フラッシュメモリーの書き替えプログラム」をダウンロードし、その「フラッシュメモリーの書き替えプログラム」を実行することができます。 フラッシュメモリーの書き替え方法は「6.6 シングルブートモードによる書き替え方法」を参照してください。
シングルチップモード	ユーザーのアプリケーションプログラムを実行します。また、RAM に「フラッシュメモリー書き替えプログラム」を配置して実行することで、内蔵フラッシュメモリーを書き換える事ができます。 内蔵する全てのフラッシュメモリーに対して操作が可能ですが、フラッシュメモリー書き換え中はフラッシュメモリー上のユーザーのアプリケーションプログラムは実行できません。 フラッシュメモリーの書き替え方法は「6.5 書き替え方法」を参照してください。

6.3. モードの決定

シングルチップ、シングルブートの各モードへの遷移は、RESET_N 端子からのリセットを解除するときの BOOT_N 端子の状態により決定されます。

表 6.2 動作モード設定表

動作モード	端子	
	RESET_N	BOOT_N
シングルチップモード	0→1	1
シングルブートモード	0→1	0

6.4. モードごとのメモリーマップ

「[図 1.1 256KB コードフラッシュのメモリーマップ例](#)」を参照してください。

6.5. 書き替え方法

ユーザーのセット上で内蔵 RAM に置くフラッシュメモリーの書き替えプログラムでフラッシュメモリーの書き替えを実行する方法です。ユーザーアプリケーション上で用意されているフラッシュメモリー書き替え用のプログラムで用いる通信機能が、UART と異なる場合やシングルブートとは異なるチャンネルを使用する場合に使用します。動作はシングルチップモードで行います。このため、シングルチップモードで通常のユーザーアプリケーションプログラムが動作しているノーマルモードから、フラッシュを書き替えるためのユーザーブートモードに移行する必要があります。従って、条件判定を行うプログラムをユーザーアプリケーションの中で、リセット処理プログラムの中に組み込んでください。

このモード切り替えの条件設定は、ユーザーのシステムセット条件に合わせて独自に構築してください。また、ユーザーブートモード移行後に使用するユーザー独自のフラッシュメモリー書き替えルーチンも同様にユーザーアプリケーションの中にあらかじめ組み込んでおき、ユーザーブートモード移行後にこれらのルーチンを使用して書き替えを行ってください。また、シングルチップモード（通常動作モード）中に誤ってフラッシュの内容を書き替えないよう、書き替え処理が完了した後、必要なブロックにライト/消去プロテクトをかけておくことを推奨します。ユーザーブートモード中は、全ての例外発生を禁止してください。

書き替えルーチンをフラッシュメモリーに置く場合と、外部から転送する場合の2ケースを例に、以下(1-A)、(1-B)にその手順を説明します。フラッシュメモリーへの書き込み/消去方法の詳細は、「4 フラッシュメモリー詳細」を参照してください。

6.5.1. (1-A)書き替えルーチンをフラッシュメモリーに内蔵する場合の手順例

6.5.1.1. Step-1

ユーザーは、あらかじめどのような条件(例えば端子状態)に設定されたらユーザーブートモードに移行するか、どの通信機能を使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリー上の任意のブロックにライターなどを使用して以下に示す3つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン: 書き替え動作に移るためのプログラム
- (b) フラッシュ書き替えルーチン: 書き替えデータを外部から取り込み、フラッシュメモリーを書き替えるためのプログラム
- (c) コピールーチン: 上記(b)を内蔵 RAM にコピーするためのプログラム

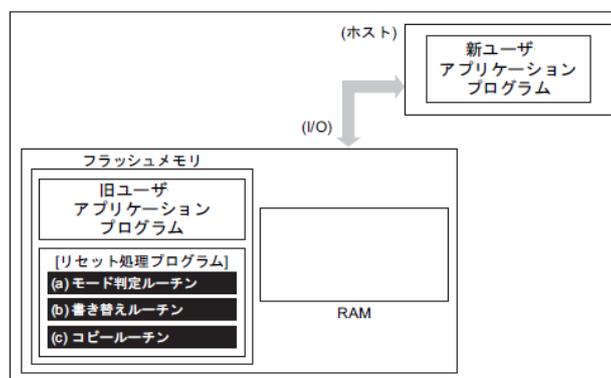


図 6.1 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(1)

6.5.1.2. Step-2

以下リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。まず、リセット処理プログラムでユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き替えのためのユーザーブートモードに移ります。(ユーザーブートモードに移行した場合は、これ以降例外を発生させないでください)

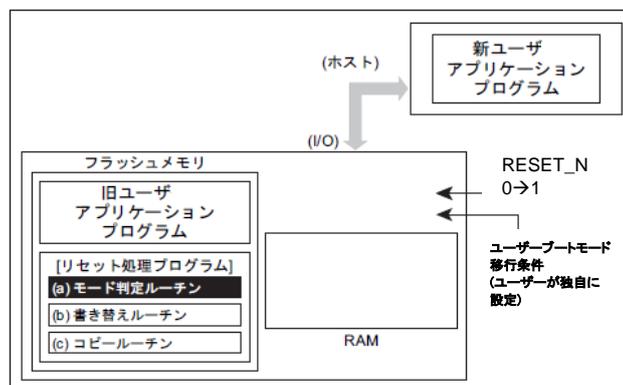


図 6.2 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(2)

6.5.1.3. Step-3

ユーザーブートモードに移ると、(c)コピールーチンを使用して、(b)書き替えルーチンを内蔵 RAM にコピーします。

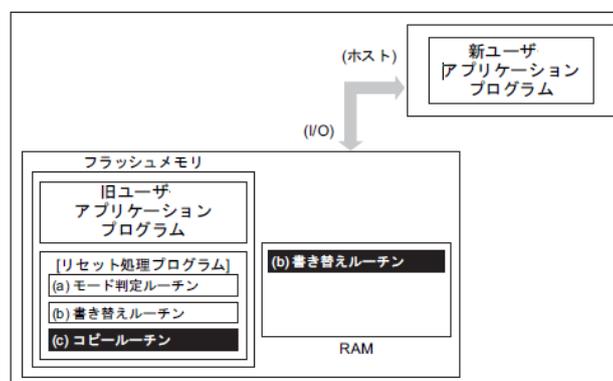


図 6.3 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(3)

6.5.1.4. Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラム領域のライト/消去プロテクトを解除して、消去(任意の消去単位)を行います。

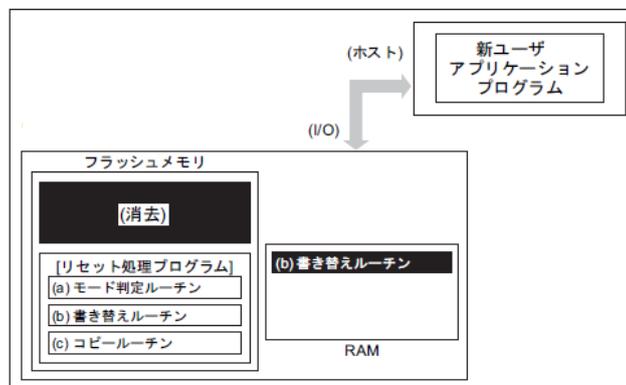


図 6.4 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(4)

6.5.1.5. Step-5

さらに、RAM上の書き替えルーチンを実行して、転送元(ホスト)より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリーの消去した領域に書き込みを行います。書き込みが完了したら、ユーザープログラム領域のライト/消去プロテクトをオンにします。

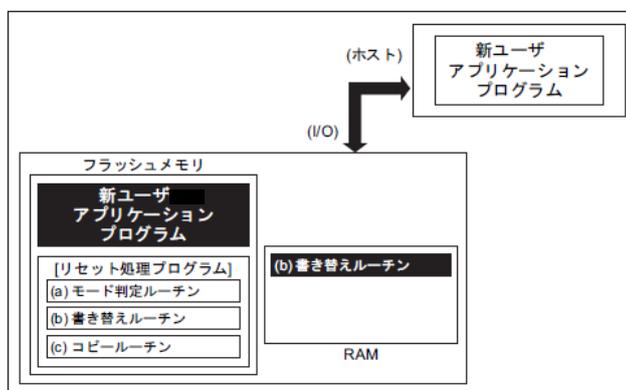


図 6.5 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(5)

6.5.1.6. Step-6

RESET_N端子を"0"にしてリセットを行い、設定条件をノーマルモードの設定にします。リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。

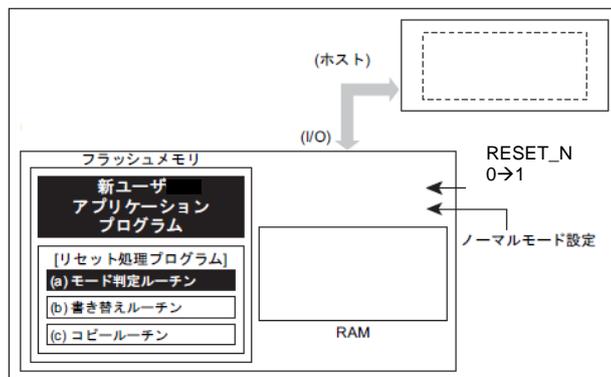


図 6.6 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(6)

6.5.2. (1-B)書き替えルーチンを外部から転送する手順例

6.5.2.1. Step-1

ユーザーは、あらかじめどのような条件(例えば端子状態)に設定されたらユーザーブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリー上の任意のブロックにライターなどを使用して以下に示す 2 つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン: 書き替え動作に移るためのプログラム
- (b) 転送ルーチン: 書き替えプログラムを外部から取り込むためのプログラム

また、下記に示すプログラムはホスト上に用意します。

- (c) 書き替えルーチン: 書き替えを行うためのプログラム

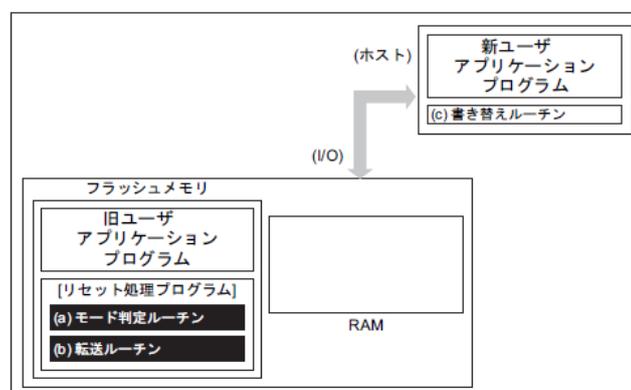


図 6.7 書き替えルーチンを外部から転送する手順(1)

6.5.2.2. Step-2

以下、リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。

まず、リセット解除後のリセット処理プログラムでユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き替えのためのユーザーブートモードに移ります。(ユーザーブートモードに移行した場合は、これ以降例外を発生させないでください)

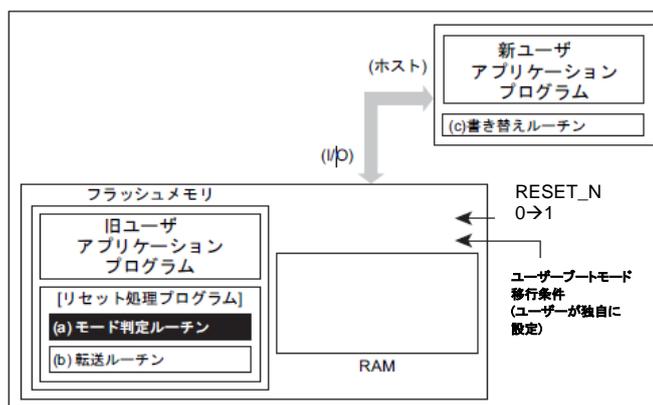


図 6.8 書き替えルーチンを外部から転送する手順(2)

6.5.2.3. Step-3

ユーザーブートモードに移ると、(b)転送ルーチンを使用して、転送元(ホスト)より(c)書き替えルーチンを内蔵 RAM にロードします。

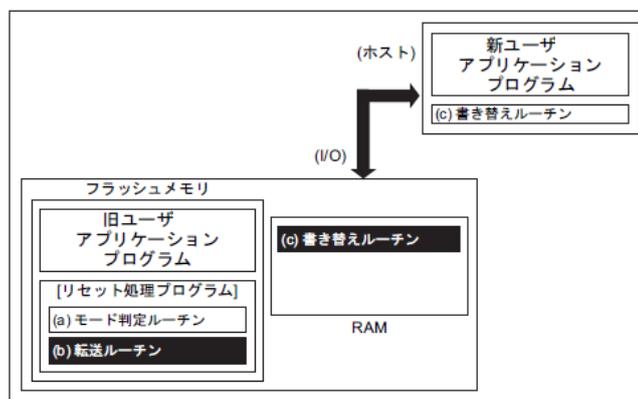


図 6.9 書き替えルーチンを外部から転送する手順(3)

6.5.2.4. Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラム領域のライト/消去プロテクトを解除して、消去(任意の消去単位)を行います。

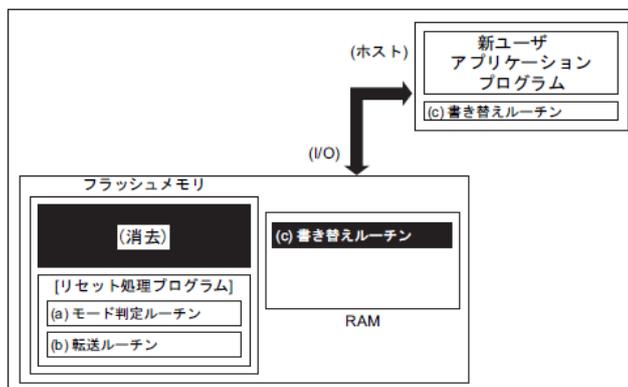


図 6.10 書き替えルーチンを外部から転送する手順(4)

6.5.2.5. Step-5

さらに、RAM上の(c)書き替えルーチンを実行して、転送元(ホスト)より新ユーザーアプリケーションプログラムのデータをロードし、消去した領域に書き込みを行います。書き込みが完了したら、ユーザープログラム領域のライト/消去プロテクトをオンにします。

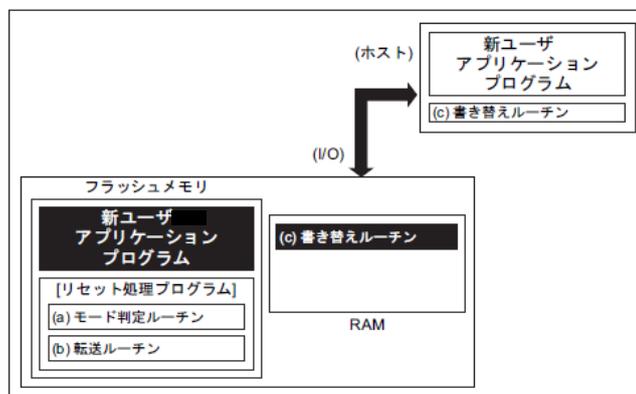


図 6.11 書き替えルーチンを外部から転送する手順(5)

6.5.2.6. Step-6

RESET_N 入力端子を"0"にしてリセットを行い、設定条件をノーマルモードに設定します。リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。

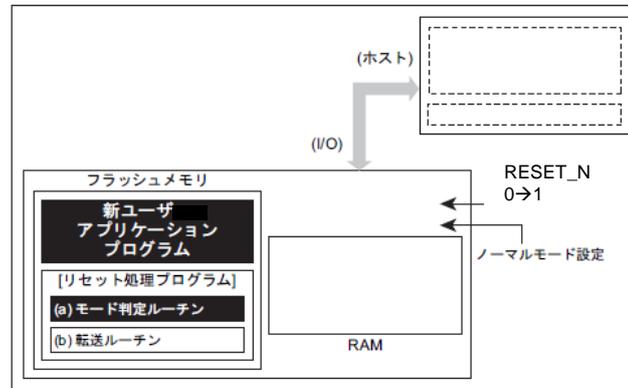


図 6.12 書き替えルーチンを外部から転送する手順(6)

6.6. シングルブートモードによる書き替え方法

6.6.1. 概要

シングルブートモードは、リセット解除後 内蔵するブート ROM(マスク ROM)から起動します。

ブート ROM が割り込みベクターテーブルを含む領域にマッピングされ、フラッシュメモリーはブート ROM 領域とは別のアドレス空間にマッピングされます。

シングルブートモードでは、コマンドおよびデータをシリアル転送してフラッシュメモリーの書き替えを行います。

表 6.3 機能とコマンド

機能/コマンド	基本動作	動作の説明	備考/参照先
通信機能	通信	UART を使用して通信を行います。	
	通信レート	外部ホストコントローラーからあらかじめ決められたレートで送られた信号を解析して、通信レートを自動設定します。	「表 6.7 転送可能なボーレートの設定例(fc=10MHz、誤差含まず)」
RAM 転送コマンド	RAM 転送	通信機能を使って外部ホストコントローラーから送られてくるフラッシュ書き換えプログラムを内蔵 RAM へ格納し、これを実行します。	
	パスワード	8 バイト～255 バイトの任意の長さのパスワードを使用可能です。パスワードが一致しない場合はエラーとなり RAM 転送は実行しません。	ユーザープログラムの一部をパスワードとして利用します。
フラッシュメモリー消去コマンド	フラッシュメモリー消去	フラッシュメモリー消去コマンドは、書き込み/消去プロテクトおよびセキュリティの状態にかかわらず、パスワード無しでユーザーインフォメーションエリアを除く全てのフラッシュメモリーを消去します。	対象: コードフラッシュ、 プロテクトビット、 メモリスワップ設定、 セキュリティビット

ターゲット(TXZ マイクロコントローラー)の UART(注)と外部ホストコントローラー(以降コントローラー)を接続し、コントローラー側から送られて来る「フラッシュ書き換えプログラム」を内蔵 RAM に格納し、RAM 上の「フラッシュ書き換えプログラム」を実行してフラッシュメモリーの書き替えを行います。コントローラー側との通信の詳細は後述のプロトコルに従ってください。

シングルブートモード中は、全ての例外発生を禁止してください。

シングルチップモード(通常動作)中に誤ってフラッシュメモリーの内容を書き替えないよう、書き替え処理が完了したら必要なブロックに書き込み/消去プロテクトをかけておくことを推奨します。

注) UART についての詳細はリファレンスマニュアル「非同期シリアル通信回路」を参照してください。

6.6.2. モード設定

オンボードプログラミングを実行するためには、本デバイスをシングルブートモードで立ち上げます。シングルブートモードで立ち上がるための設定は「6.3 モードの決定」を参照してください。

6.6.3. インターフェース仕様

シングルブートモードでのシリアル通信フォーマットを以下に示します。オンボードプログラミングを実行するためには、コントローラー側の通信フォーマットも同様に設定する必要があります。

通信チャネル: UART チャネル x (製品によって異なります)
シリアル転送モード: UART(非同期通信) モード, 半 2 重通信, LSB ファースト
データ長: 8 ビット
パリティビット: なし
STOP ビット: 1 ビット
ボーレート: 任意のボーレート
(「表 6.7 転送可能なボーレートの設定例(fc=10MHz、誤差含まず)」参照)
WDT : 停止

内蔵ブートプログラムは、クロック/モード制御ブロックの設定は初期状態のまま動作します(fc=10MHz、使用する機能ブロックへのクロック供給は行います)。

ボーレートは、「6.6.6.1 シリアル動作モード判定」で説明しているようにタイマーカウンターを用いて判定します。判定時のボーレートはタイマーで計測可能な範囲である必要があります。

内蔵ブートプログラムで使用する端子を「表 6.4 使用端子例」にまとめます。これ以外の端子は内蔵ブートプログラムでは操作しません。通信端子については、リファレンスマニュアル「製品個別情報」を参照してください。

表 6.4 使用端子例

端子種類	端子名
モード設定端子	BOOT_N
リセット端子	RESET_N
通信端子	UTxTXD
	UTxRXD

6.6.4. メモリーの制約について

シングルブートモードでは、内蔵 RAM、内蔵フラッシュメモリーに対して「表 6.5 シングルブート時のメモリーの制約」のような制約がありますのでご注意ください。

表 6.5 シングルブート時のメモリーの制約

メモリー	制約内容
内蔵 RAM	“0x20000000”～“0x200003FF”番地は BOOT プログラムのワークエリアになります。 受信したプログラムは“0x20000400”から“転送可能最終アドレス”までに格納してください。転送可能最終アドレスは、リファレンスマニュアルの「製品個別情報」を参照してください。
内蔵フラッシュメモリー	コードフラッシュの“0x5E001000”～(最大容量-N) までをパスワード領域として使用可能です(N はパスワード長)。

6.6.5. 動作コマンド

内蔵ブートプログラムには、以下の動作コマンドが準備されています。

表 6.6 シングルブート時コマンド

動作コマンドデータ	動作コマンド
0x10	RAM 転送
0x40	フラッシュメモリー消去

6.6.5.1. RAM 転送

RAM 転送は、コントローラーから送られてくるユーザープログラムのデータを内蔵 RAM へ格納します。転送が正常に終了するとユーザープログラムの実行を開始します。ユーザープログラム領域として、内蔵ブートプログラムで使用する領域(“0x20000000”～“0x200003FF”)を除く、“0x20000400”以降を使用可能です。実行開始アドレスは、RAM 格納開始アドレスになります。

この RAM 転送機能により、ユーザー独自のオンボードプログラミング制御を行うことができます。ユーザープログラムでオンボードプログラミングを実行するためには、「6.5. 書き替え方法」を参照してください。

6.6.5.2. フラッシュメモリー消去

フラッシュメモリー消去コマンドはユーザーインフォメーションエリアを除く全てのフラッシュメモリーを消去します。書き込み/消去プロテクトおよびセキュリティーの状態にかかわらず、パスワード無しでコードフラッシュ、プロテクトビット、セキュリティービットを消去します。

本コマンドではユーザーインフォメーションエリアは消去されません。消去が必要な場合は、本コマンドを実行後、RAM 転送でユーザーインフォメーションエリア消去用プログラムを転送して、これを実行してください。

6.6.6. コマンドによらず共通の動作

内蔵ブートプログラム実行で、共通に行われる動作について説明します。

6.6.6.1. シリアル動作モード判定

コントローラーは、表 6.7 のボーレートで 1 バイト目を "0x86" にして送信してください。「図 6.13 シリアル動作モード判定データ」に波形を示します。通信ができない場合は、ボーレートを下げてください。

表 6.7 転送可能なボーレートの設定例($f_c=10\text{MHz}$ 、誤差含まず)

ボーレート ※()は算出値	<BRN>	<BRK>
9600 (9599)	65	57
19200 (19203)	32	29
38400 (38388)	16	46
57600 (57637)	10	10
62500 (62500)	9	0
76800 (76923)	8	55
115200 (115274)	5	37
128000 (127796)	4	7

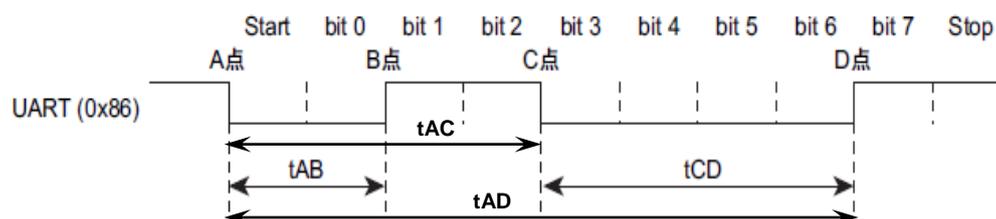


図 6.13 シリアル動作モード判定データ

内蔵ブートプログラムはリセット解除後の 1 バイト目のシリアル動作モード判定データ "0x86" を、タイマーカウンタを用いて「図 6.13 シリアル動作モード判定データ」の t_{AB} , t_{AC} と、 t_{AD} の時間から求めています。「図 6.14 シリアル動作モード受信フローチャート」のフローチャートに示すように、CPU が受信端子のレベルをモニターしてレベルの変化があると、そのときのタイマー値を取り込みます。このため、 t_{AB} , t_{AC} と、 t_{AD} のタイマー値には誤差が生じます。また、ボーレートが速いときには、CPU は受信端子のレベルの変化を判断できない場合がありますので注意してください。

「図 6.15 シリアル動作モード判定フローチャート」に示すように、シリアル動作モードの判定は、受信端子が "L" レベルのときの時間幅の大小関係で判定しています。 $t_{AB} \leq t_{CD}$ の場合 UART と判定し、ボーレートの自動設定が可能かどうかを t_{AD} の時間から判定します。 $t_{AB} > t_{CD}$ の場合、UART と判定しません。なお、先に述べたように、 t_{AB} , t_{AC} , t_{AD} のタイマー値には誤差が生じているため、ボーレートが速く、動作周波数が低い場合、各タイマー値が小さくなり、意図しない判断を行うことがありますので注意してください(必要に応じて、書き換えルーチン内で UART の再設定を行ってください)。

例えば、コントローラーは UART で通信したいのに、UART と判定されないことがあります。また、コントローラーから送出したボーレートと認識しないことがあります。このようなことを考慮して、コントローラーは 1 バイト目のデータを送信後、タイムアウト時間内にデータ "0x86" を正常受信できなければ通信不可能と判断してください。

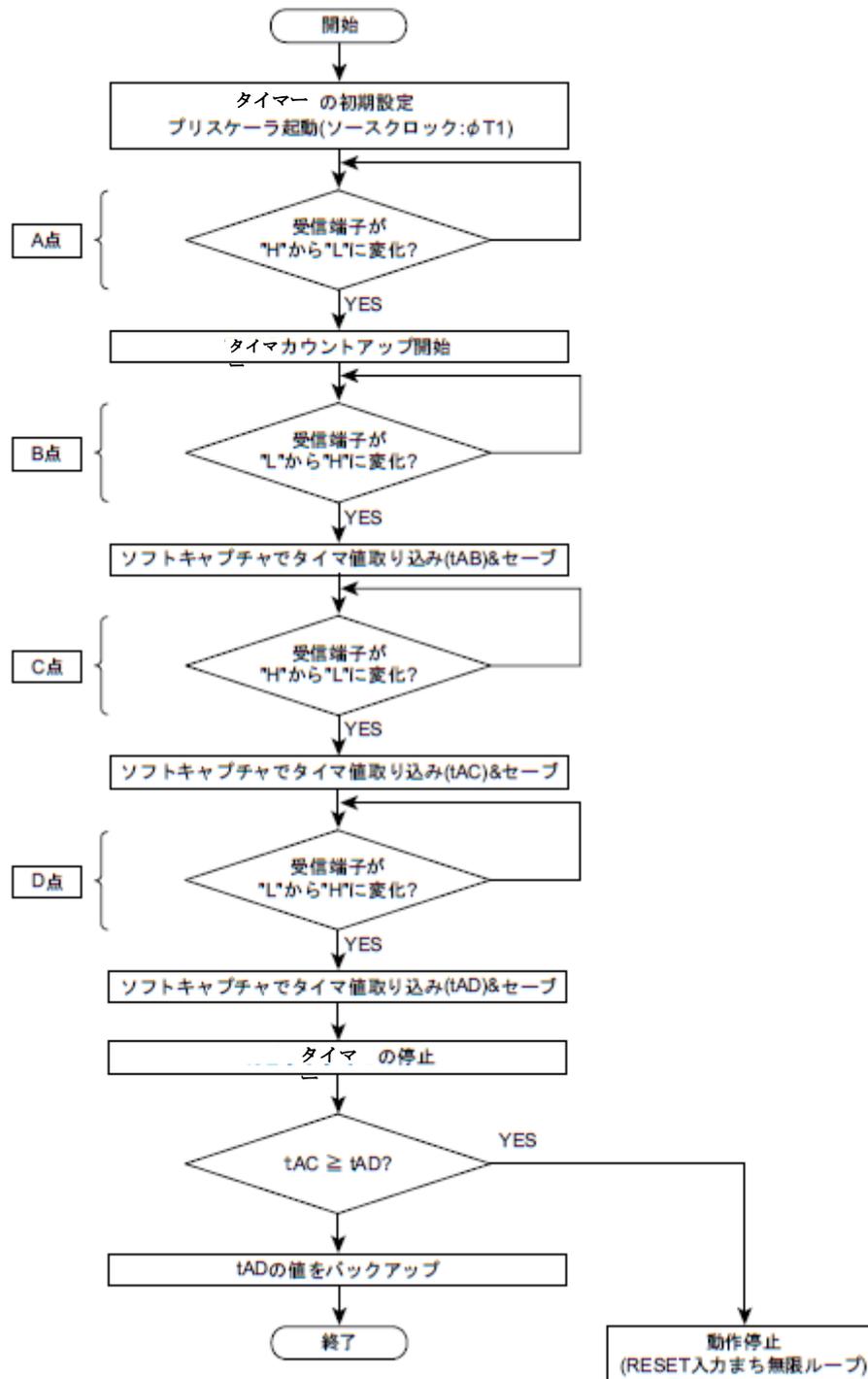


図 6.14 シリアル動作モード受信フローチャート

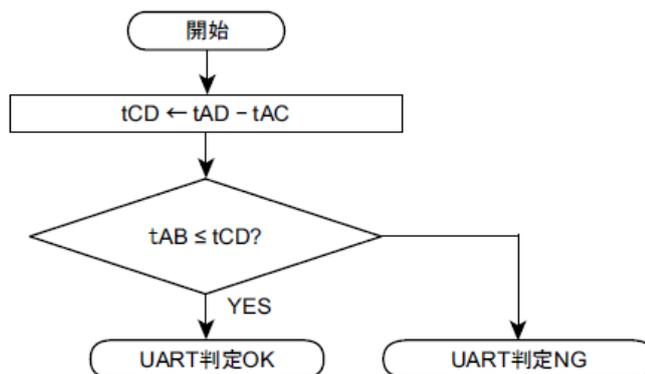


図 6.15 シリアル動作モード判定フローチャート

6.6.6.2. ACK 応答データ

内蔵ブートプログラムは処理状況を各種コードによってコントローラーに送信します。表 6.8 シリアル動作判定データに対する ACK 応答データから「表 6.11 フラッシュメモリー消去動作に対する ACK 対応データ」に各受信データに対する ACK 応答データを示します。

ACK 応答データの上位 4 ビットは、動作コマンドデータの上位 4 ビットになります。また 3 ビット目は受信エラーを表し、0 ビット目は動作コマンドエラー、CHECKSUM エラー、パスワードエラーの状態を表します。1 ビット目と 2 ビット目は常に"0"になります。

表 6.8 シリアル動作判定データに対するACK応答データ

送信データ	送信データの意味
0x86	UART での通信が可能と判断した(注)

注) ボーレートの設定が不可能と判定したら、何も送信しないで動作を停止します。

表 6.9 動作コマンドデータに対するACK応答データ

送信データ	送信データの意味
0x78(注)	動作コマンドデータに受信エラーが発生した。
0x71(注)	未定義の動作コマンドデータを正常受信した。
0x10	RAM 転送コマンドと判定した。
0x40	フラッシュメモリー消去コマンドと判定した。

注) 上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになります。

表 6.10 CHECKSUMデータに対するACK応答データ

送信データ	送信データの意味
0xN8(注)	受信エラーが発生していた。
0xN1(注)	CHECKSUM エラーが発生した。あるいは、パスワードエラーが発生した。
0xN0(注)	CHECKSUM 値は正常な値と判定した。

注) 上位 4 ビットは動作コマンドデータの上位 4 ビットになります。

表 6.11 フラッシュメモリー消去動作に対するACK対応データ

送信データ	送信データの意味
0x54	フラッシュメモリー消去イネーブルコマンドと判定した。
0x4F	フラッシュメモリー消去コマンド終了
0x4C	フラッシュメモリー消去コマンドが不正に終了した
0x47	フラッシュメモリー消去コマンドを中止した。

6.6.6.3. パスワード

フラッシュメモリー内の任意のデータ(ユーザーメモリーの一部)を使用してパスワードを設定することができます。パスワードが設定されると、RAM 転送コマンドはパスワードの認証が必要となります。

(1) パスワードの仕組み

フラッシュメモリー内の任意のデータ(連続する 8 バイト以上のデータ)をパスワードとして設定することができ、外部コントローラーから送信されるパスワード列と、パスワードが設定された MCU のメモリーデータ列を比較することによりパスワード列の認証を行います。

(2) パスワードの構成

パスワードは、PLEN、PNSA、PCSA、パスワード列の 4 つで構成されています。「図 6.16 パスワードの構成(送信例)」にパスワードの構成(送信例)を示します。

- PLEN (パスワード長データ)

パスワードの長さを指定します。“0x08”～“0xFF”の範囲で指定します。PNSA で示されるアドレスのデータが“0x07”以下、またはパスワード長データが PNSA で示されるアドレスのデータと一致しない場合はパスワードエラーとなります。

- PNSA (パスワード数格納アドレス)

“0x5E001000”～(最大メモリーアドレス)内のアドレスを 4 バイトで指定します。指定されたアドレスのメモリーデータがパスワード列のバイト数になります。PNSA で示されるアドレスのデータが“0x07”以下、またはパスワード長データが PNSA で示されるアドレスのデータと一致しない場合はパスワードエラーとなります。なお、ここで指定されたメモリーデータは N と定義します。

- PCSA (パスワード比較開始アドレス)

“0x5E001000”～(最大メモリーアドレス-N)内のアドレスを 4 バイトで指定します。指定されたアドレスは、パスワード列と比較するための開始アドレスとなります。

PCSA が上記の範囲を超える場合はパスワードエラーとなります。

- パスワード列

8～255 (=N)バイトのデータを指定します。PCSA で指定されたアドレスを開始アドレスとして、メモリーデータとパスワード列を N バイト分だけ比較します。比較の結果、不一致があった場合、または連続して 3 バイト以上同一のデータが検出された場合はパスワードエラーとなります。

- パスワードエラー

パスワードエラーが発生すると、それ以降外部デバイスは TXZ と通信ができなくなります。通信を再開する場合はリセット端子(RESET_N)からリセットをかけてシングルブートモードを再起動してください。

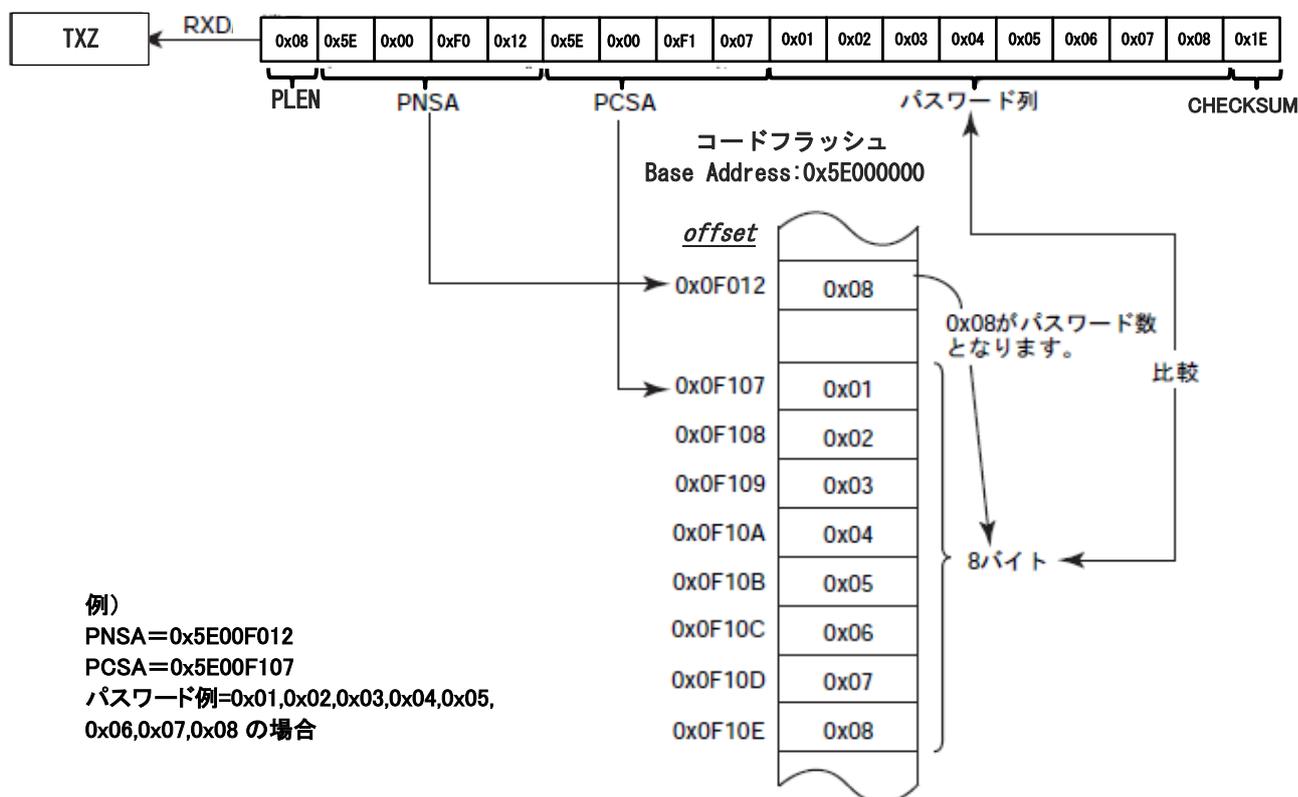


図 6.16 パスワードの構成(送信例)

(3) パスワードの設定/解除/認証

- パスワードの設定

パスワードはユーザープログラムの一部を利用することで実現しています。よって、パスワード設定のために特別な処理は不要で、コードフラッシュにプログラムを書き込んだ時点でパスワードが設定されたこととなります。

- パスワードの解除

パスワードを解除するためにはコードフラッシュの全面消去が必要です。ユーザーインフォメーションエリアを除くコードフラッシュが全て“0xFF”に初期化された時点でパスワードは解除されます。

- パスワードの認証が不要なケース

コードフラッシュの全領域が“0xFF”の場合にのみブランク品と判定され、パスワードの認証は行われません。

(4) パスワードの設定値、設定範囲

パスワードは「表 6.12 パスワードの設定値、設定範囲」の条件に従って設定してください。この条件を満たさない場合パスワードエラーとなります。

表 6.12 パスワードの設定値、設定範囲

パスワード	ブランク品	非ブランク品
PNSA (パスワード数格納アドレス)	0x5E001000 ≤ PNSA ≤ 最大メモリーアドレス	0x5E001000 ≤ PNSA ≤ 最大メモリーアドレス
PCSA (パスワード比較開始アドレス)	0x5E001000 ≤ PCSA ≤ 最大メモリーアドレス	0x5E001000 ≤ PCSA ≤ 最大メモリーアドレス-N
N (パスワード数)	必要 (注 2)	8 ≤ N
パスワード	必要 (注 2)	必要 (注 1)

注 1) 3 バイト以上同一となるデータは、パスワード列として設定できません

注 2) ブランク品に対してもダミーのパスワード数とパスワード列を送信してください

6.6.6.4. パスワード判定

(1) RAM 転送コマンドでのパスワード判定

「6.6.8.RAM 転送コマンドの通信ルール」における No.5 からのパスワード判定について説明します。パスワードエリアのデータのアドレスが範囲外の場合はパスワードアドレスエラーとなります。また、「図 6.17 パスワードチェックフローチャート」に示すようにパスワードに 3 バイト以上同一データが連続した場合、“0xFF”以外の同一データになっていた場合、パスワードエリアエラーと判定します。パスワードアドレスエラー、パスワードエリアエラーと判定された場合、パスワードデータの照合結果に関わらず、ACK 対応は“0x11”を送信します。

次に、受信データ(パスワードデータ)の照合を行います。N バイト分全てがフラッシュメモリー内のパスワードと一致しないと、パスワードエラーになります。パスワードエラーと判定された場合、ACK 応答は、パスワードエラーとなります。

セキュリティー機能が有効な状態でもパスワードの参照を行います。

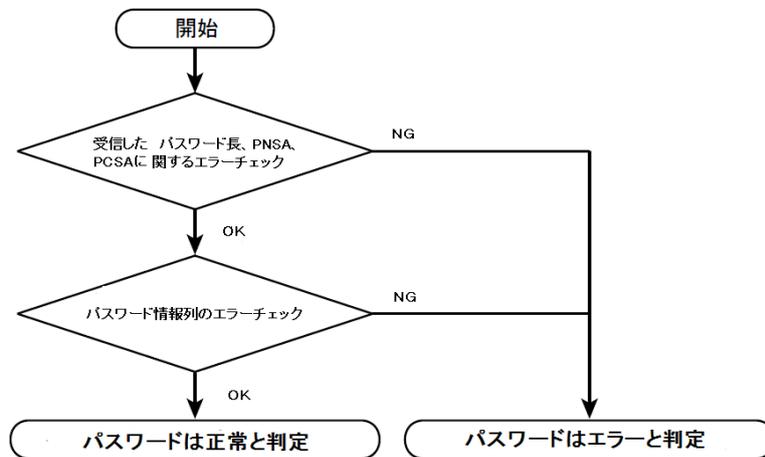


図 6.17 パスワードチェックフローチャート

6.6.6.5. CHECKSUM の計算方法

CHECKSUM の計算方法は、送信データを符号なし 8 ビット加算(オーバーフローを無視)して得られた下位 8 ビット値の 2 の補数値を求めています。コントローラーは CHECKSUM 値を送信するときは、本計算方法を使用してください。

例) CHECKSUM 計算例

2 バイトのデータ“0xE5”、“0xF6”の CHECKSUM 値を求める場合、まず符号なし 8 ビット加算を行います。

$$0xE5 + 0xF6 = 0x1DB$$

この値の下位 8 ビットに対しての 2 の補数をとると以下のようになり、この値が CHECKSUM 値になります。従って、コントローラーには“0x25”を送信します。

$$0 - 0xDB = 0x25$$

6.6.7. シリアル動作モード判定の通信ルール

シリアル動作モード判定の通信ルールを示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」：コントローラーからターゲット(TXZ マイクロコントローラー)へ

転送方向「T→C」：ターゲット(TXZ マイクロコントローラー)からコントローラーへ

表 6.13 シリアル動作モード判定の通信ルール

No	転送方向	転送データ	内容
1	C→T	シリアル動作モード、ボーレート設定	コントローラー側はシリアル動作モードを判定するデータを送信します。ターゲット側のモード判定の詳細は「6.6.6.1 シリアル動作モード判定」を参照してください。
		0x86	コントローラー側は“0x86”を送信してください。ターゲット側が UART モード OK と判定した場合、ボーレートの設定が可能かどうかを判定します。設定が不可能と判断した場合は動作を停止するため、通信が行えなくなります。
2	T→C	シリアル動作モードに対する ACK 応答	コントローラー側の受信データは、1 バイト目のシリアル動作モード設定データに対する ACK 応答データになります。ターゲット側は設定が可能と判定した場合、UART の設定を行います。受信を許可するタイミングは、送信バッファにデータを書き込む前に行っています。
		正常の場合: 0x86	ターゲット側が設定可能と判定した場合“0x86”を送信し、設定が不可能と判定した場合、動作を停止するため何も送信しません。コントローラー側は、1 バイト目のデータの送信が終了した後、タイムアウト時間(5 秒)を設けてください。タイムアウト時間内に、データ“0x86”を正常受信できなければ、通信不能と判断してください。
3	—	—	コントローラー側は動作コマンドデータを送信してください。各動作コマンドの転送フォーマットは「6.6.8 RAM 転送コマンドの通信ルール」または「6.6.9 フラッシュメモリー消去の通信ルール」を参照してください。

6.6.8. RAM 転送コマンドの通信ルール

RAM 転送コマンドの通信ルールを示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」：コントローラーからターゲット(TXZ マイクロコントローラー)へ

転送方向「T→C」：ターゲット(TXZ マイクロコントローラー)からコントローラーへ

表 6.14 RAM 転送コマンドの通信ルール

No	転送方向	転送データ	内容
1	C→T	動作コマンドデータ(0x10)	コントローラー側は RAM 転送コマンドデータ"0x10"を送信してください。
2	T→C	動作コマンドに対する ACK 応答 正常の場合: 0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ "0x18"を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、「表 6.6 シングルブート時コマンド」に記載の動作コマンドデータとの照合を行います。照合に失敗した場合、異常の ACK 応答データ "0x11"を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ "0x10"を返信して、次の送信データを待ちます。
3	C→T	パスワード長(PLEN) (1 バイト)	コントローラー側はコードフラッシュのパスワード長のデータを送信してください。
4	C→T	パスワード長格納アドレス (PNISA) (4 バイト)	コントローラー側はコードフラッシュのパスワード長を格納しているアドレスのデータを送信してください。
5	C→T	パスワード格納開始アドレス (PCISA) (4 バイト)	コントローラー側はコードフラッシュのパスワードを格納している先頭のアドレスのデータを送信してください。
6	C→T	パスワード列(8 バイト~255 バイト)	コントローラー側はコードフラッシュのパスワードデータを送信してください。 消去済みの場合はダミーのデータを送信してください。
7	C→T	送信データ(No.3~6)の CHECKSUM	コントローラー側は送信データ(No.3~6)の CHECKSUM 値を送信してください。 CHECKSUM の計算方法は「6.6.6.5CHECKSUM の計算方法」を参照してください。
8	T→C	パスワード長エラーチェック、パスワード格納アドレスエラーチェック、パスワード照合、CHECKSUM 値に対する ACK 応答 ・ブランク品の場合: 0x14 ・正常の場合: 0x10 ・異常の場合: 0x11 ・通信異常の場合: 0x18	ターゲット側は受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ "0x18"を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、CHECKSUM の確認とパスワードの照合を行います。パスワードの照合の詳細は「6.6.6.4 パスワード判定」を参照してください。 照合に失敗した場合、異常の ACK 応答データ "0x11"を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ "0x10"を返信して、次の送信データを待ちます。ブランク品の場合は、ACK 応答データ "0x14"を返信して、次の送信データを待ちます。
9	C→T	RAM 格納開始アドレス 31~24	コントローラー側は次の送信データとして RAM 格納デー

10	C→T	RAM 格納開始アドレス 23~16	タの格納先の RAM の開始アドレスを 4 回に分けて送信してください。送信順番は、1 番目がアドレスの 31 ビット~24 ビットに対応し、4 番目が 7 ビット~0 ビットに対応します。RAM のアドレス“0x20000400”から“RAM の転送可能最終アドレス”に収まるように指定してください。ターゲット側は受信データをチェックします。受信エラーがある場合は通信異常の ACK 応答データ“0x18”を返信して初めの動作コマンドデータ待ち状態に戻ります。受信エラーがない場合は何も ACK 応答データは返信しないで次の送信データを待ちます。
11	C→T	RAM 格納開始アドレス 15~8	
12	C→T	RAM 格納開始アドレス 7~0	コントローラー側はブロック転送するバイト数を送信してください。送信順番は、1 番目が転送バイト数の 15 ビット~8 ビット目に対応し、2 番目が 7 ビット~0 ビット目に対応します。RAM のアドレス“0x20000400”から“RAM の転送可能最終アドレス”に収まるように指定してください。ターゲット側は受信データをチェックします。受信エラーがある場合は通信異常の ACK 応答データ“0x18”を返信して初めの動作コマンドデータ待ち状態に戻ります。受信エラーがない場合は何も ACK 応答データは返信しないで次の送信データを待ちます。
13	C→T	RAM 格納バイト数 15~8	
14	C→T	RAM 格納バイト数 7~0	送信データ(No.9~14)の CHECKSUM 値を送信してください。
15	C→T	送信データ(No.9~14)の CHECKSUM 値	
16	T→C	CHECKSUM 値に対する ACK 応答 正常の場合: 0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。受信エラーがある場合、通信異常の ACK 応答データ“0x18”を返信して、初めの動作コマンドデータ待ち状態に戻ります。受信エラーがない場合、CHECKSUM の確認を行います。確認に失敗した場合、異常の ACK 応答データ“0x11”を返信して、初めの動作コマンドデータ待ち状態に戻ります。確認に成功した場合、正常の ACK 応答データ“0x10”を返信して、次の送信データを待ちます。
17	C→T	RAM 格納データ	コントローラー側は RAM に格納するデータを送信してください。ターゲット側は RAM 格納バイト数分のデータを受信します。
18	C→T	送信データ(No.17)の CHECKSUM 値	コントローラー側は送信データ(No.17)の CHECKSUM 値を送信してください。
19	T→C	チェックサム照合に対する ACK 応答 ・正常の場合: 0x10 ・異常の場合: 0x11 ・通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。受信エラーがある場合、通信異常の ACK 応答データ“0x18”を返信して、初めの動作コマンドデータ待ち状態に戻ります。受信エラーがない場合、CHECKSUM の確認を行います。確認に失敗した場合、異常の ACK 応答データ“0x11”を返信して、初めの動作コマンドデータ待ち状態に戻ります。確認に成功した場合、正常の ACK 応答データ“0x10”を返信して、RAM 格納開始アドレス(No.9~12)を分岐先アドレスとしてジャンプします。(注)

注) BootROM プログラムは、使用した機能(ポート、UART、タイマー、RAM など)の設定は初期化しません。

6.6.9. フラッシュメモリー消去の通信ルール

フラッシュメモリーチップ消去を示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」：コントローラーからターゲット(TXZ マイクロコントローラー)へ

転送方向「T→C」：ターゲット(TXZ マイクロコントローラー)からコントローラーへ

表 6.15 フラッシュメモリー消去の通信ルール

No	転送方向	転送データ	内容
1	C→T	動作コマンドデータ(0x40)	コントローラー側はフラッシュメモリー消去コマンドデータ“0x40”を送信してください。
2	T→C	動作コマンドに対する ACK 応答 正常の場合: 0x40 異常の場合: 0x41 通信異常の場合: 0x48	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ“0x48”を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、「表 6.6 シングルブート時コマンド」に記載の動作コマンドデータとの照合を行います。照合に失敗した場合、異常の ACK 応答データ“0x41”を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ“0x40”を返信して、次の送信データを待ちます。
3	C→T	消去イネーブルコマンドデータ(0x54)	コントローラー側は消去イネーブルコマンドデータ“0x54”を送信してください。
4	T→C	消去イネーブルコマンドに対する ACK 応答 ・正常の場合: 0x54 ・異常の場合: 0x51 ・通信異常の場合: 0x58	ターゲット側は受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ“0x58”を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、消去イネーブルコマンド“0x54”であることを確認します。 確認に失敗した場合、異常の ACK 応答データ“0x51”を返信して、初めの動作コマンドデータ待ち状態に戻ります。 確認に成功した場合、正常の ACK 応答データ“0x54”を返信して、チップ消去処理を行います
5	—	—	チップ消去処理実行中
6	T→C	消去完了確認に対する ACK 応答 ・消去完了の場合: 0x4F ・異常(ブランクチェックエラー)の場合: 0x4C ・タイムアウトエラーの場合: 0x47	ターゲット側はチップ消去処理の結果を返信します。 問題なく消去できた場合、正常の応答データ“0x4F”を返信します。 ブランクチェックエラーが起きた場合は、異常の応答データ“0x4C”を返します。 タイムアウトでフラッシュメモリー消去コマンドを中止した場合は、中止の応答データ“0x47”を返します。 その後、初めの動作コマンドデータ待ち状態に戻ります。

6.6.10. 内蔵ブートプログラム全体フローチャート

内蔵ブートプログラム全体フローチャートを示します。

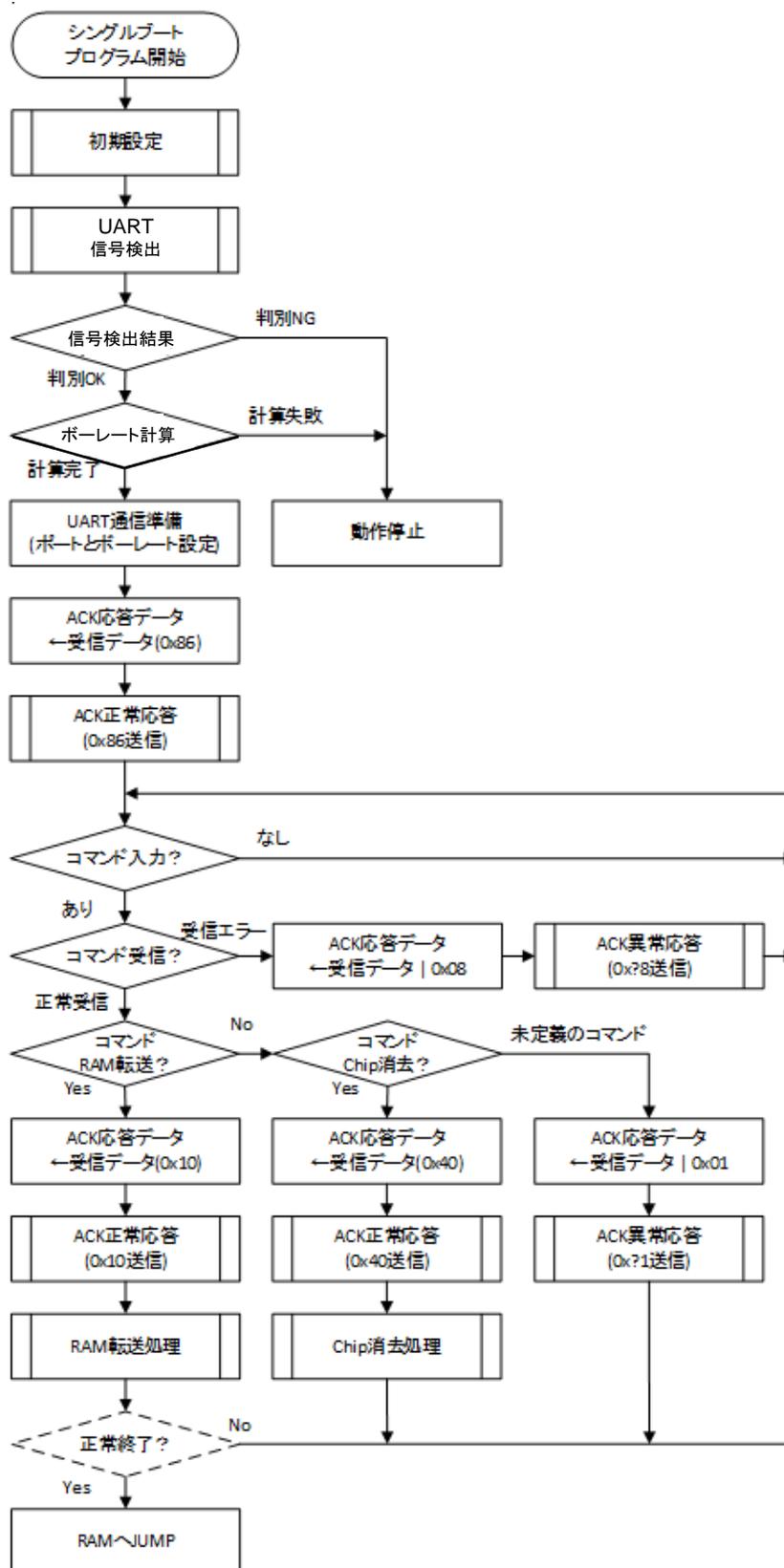


図 6.18 内蔵ブートプログラム全体フローチャート

6.6.11. 内蔵ブート ROM の書き替えアルゴリズムを利用した書き替え手順

内蔵ブートプログラムを利用した書き替え手順を示します。(UART を使った例)

6.6.11.1. Step-1

フラッシュメモリーの状態は旧バージョンのユーザープログラムが書かれた状態でも、消去されている状態でも構いません。書き替えルーチン、書き替えデータなどの転送は UART を経由して行いますので、ボード上で本デバイスの UART と外部ホストとをつなげます。書き替えを行うための(a)書き替えルーチンはホスト上に用意します。

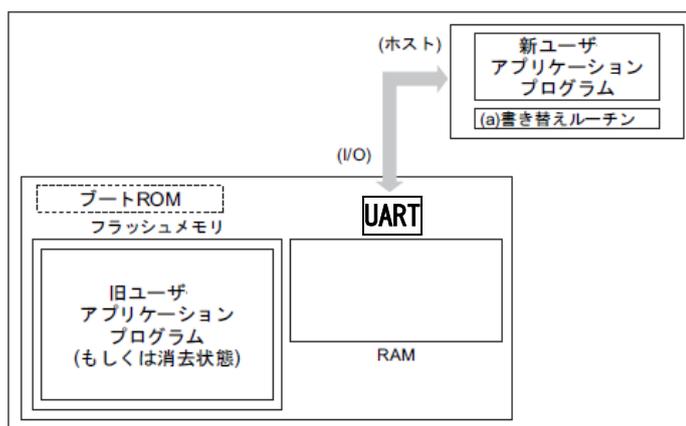


図 6.19 内蔵ブートプログラムを利用した書き替え手順(1)

6.6.11.2. Step-2

シングルブートモードの端子条件設定でリセットを解除し、ブート ROM で起動します。シングルブートモードの手順に従い、UART を経由して転送元(ホスト)より(a)書き替えルーチンの転送を行いますが、最初にユーザーアプリケーションプログラム上に記録されているパスワードとの照合を行います。詳細は「6.6.6.3 パスワードの(4)パスワードの設定値、設定範囲」を参照してください。

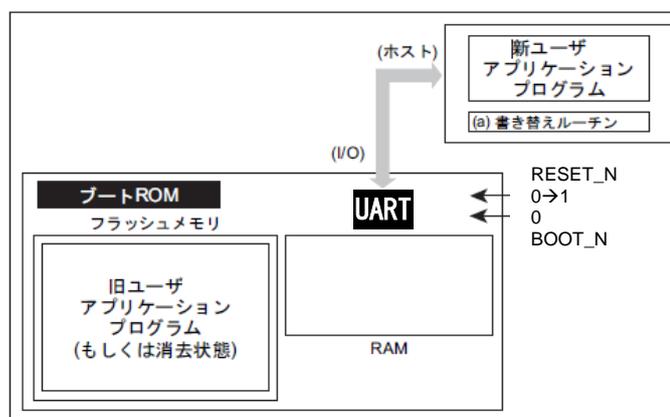


図 6.20 内蔵ブートプログラムを利用した書き替え手順(2)

6.6.11.3. Step-3

パスワードの照合が終了すると、転送元(ホスト)から(a)書き替えルーチンを転送します。ブート ROM はそのルーチンを内部 RAM にロードします。ただし、RAM 上のアドレス “0x20000400”から”RAM の転送可能最終アドレス”の範囲に格納してください。

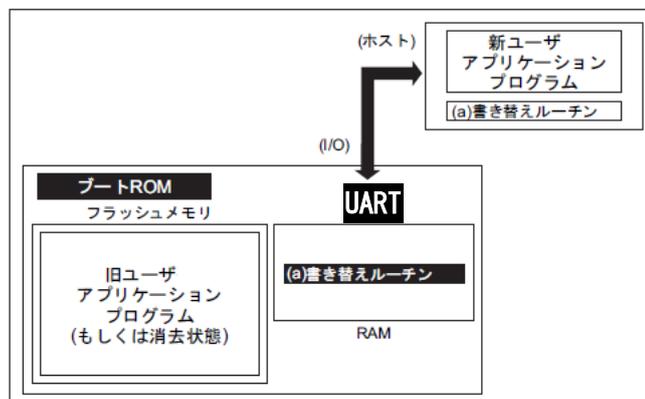


図 6.21 内蔵ブートプログラムを利用した書き替え手順(3)

6.6.11.4. Step-4

RAM 上の(a)書き替えルーチンへジャンプし、旧ユーザーアプリケーションプログラム領域の消去を行います。(任意の消去単位)

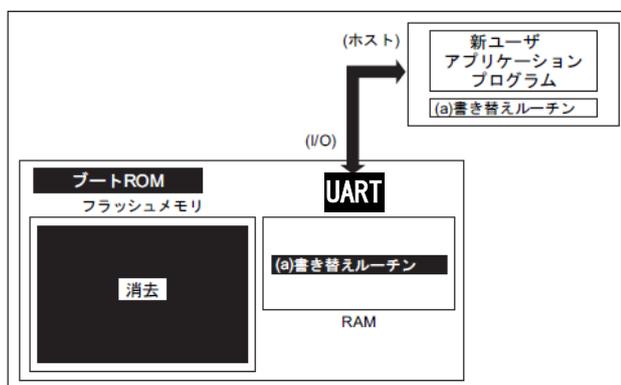


図 6.22 内蔵ブートプログラムを利用した書き替え手順(4)

6.6.11.5. Step-5

さらに、RAM上の(a)書き替えルーチンを実行して、転送元(ホスト)より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリーの消去した領域に書き込みを行います。書き込みが完了したら、ユーザープログラム領域のライト/消去プロテクトをオンにします。

下の例の場合、書き替えルーチンを転送したときと同じホストおよびUART経由で書き替えデータも転送されていますが、RAM上で動作を開始した以降では、ユーザー独自にデータバスおよび転送元を設定することもできます。方法に応じて、ボードのハードおよび書き替えルーチンを組み立ててください。

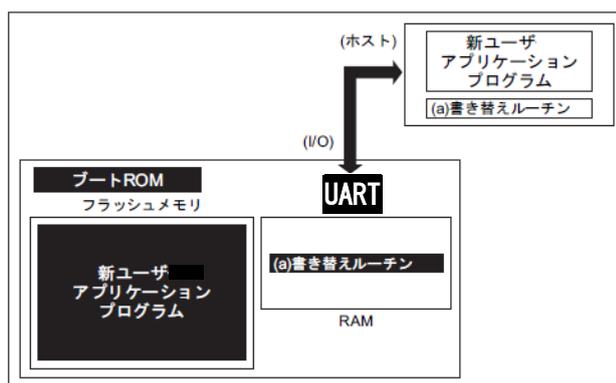


図 6.23 内蔵ブートプログラムを利用した書き替え手順(5)

6.6.11.6. Step-6

書き込みが完了したら、一度ボードの電源を切断し、ホストと接続していたケーブルをはずします。この後、再度電源を入れ直し、シングルチップモード(ノーマルモード) 起動し、新しいユーザーアプリケーションプログラムを実行します。

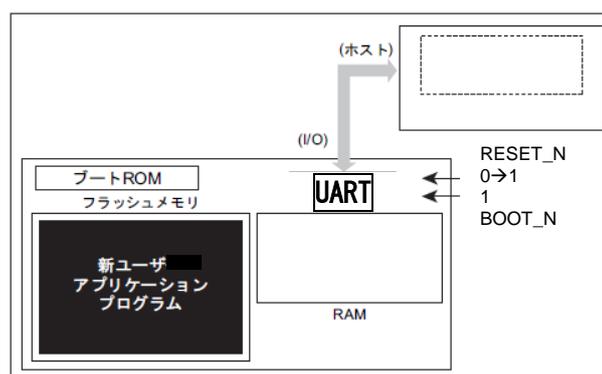


図 6.24 内蔵ブートプログラムを利用した書き替え手順(6)

6.7. ユーザーブートプログラムの書き替え方法

メモリスワップ機能を利用して、ユーザーブートプログラムが残るように Page0 と Page1 の領域を交換させてフラッシュメモリーの書き替えを実行する方法です。

ユーザーブートプログラムの書き替え手順の参考例を以下に示します。

(以下では、スワップサイズは 4K バイト(設定済み)、Page1 のプログラムは、Page0 からコピーするものとして記載します)

6.7.1. フラッシュ書き替えの手順例

6.7.1.1. Step-1

[FCSWPSR]<SWP[1:0]>から"00"が読み出せることを確認します。

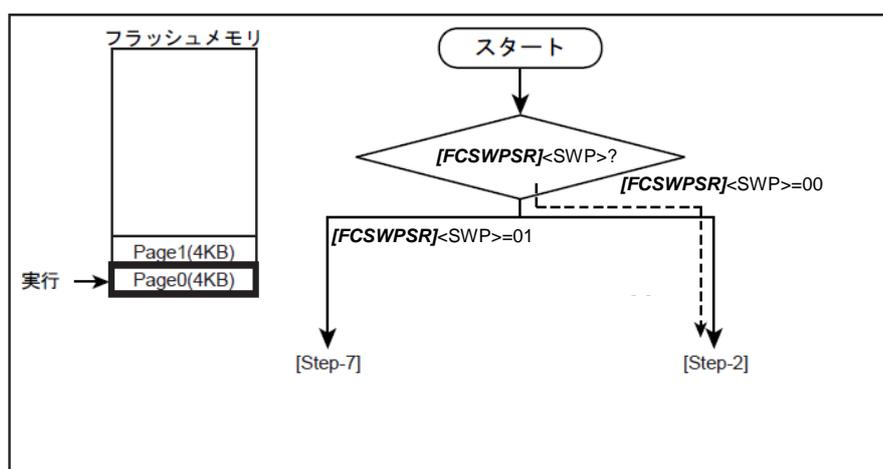


図 6.25 ユーザーブートプログラムの書き替え(1)

6.7.1.2. Step-2

$[FCPSR0]<PG1>=0$ であるかチェックします。プロテクト状態がイネーブル ($<PG1>=1$) の場合は、 $[FCPMR0]<PM1>$ に "0" を書いてプロテクトを一時解除してください。

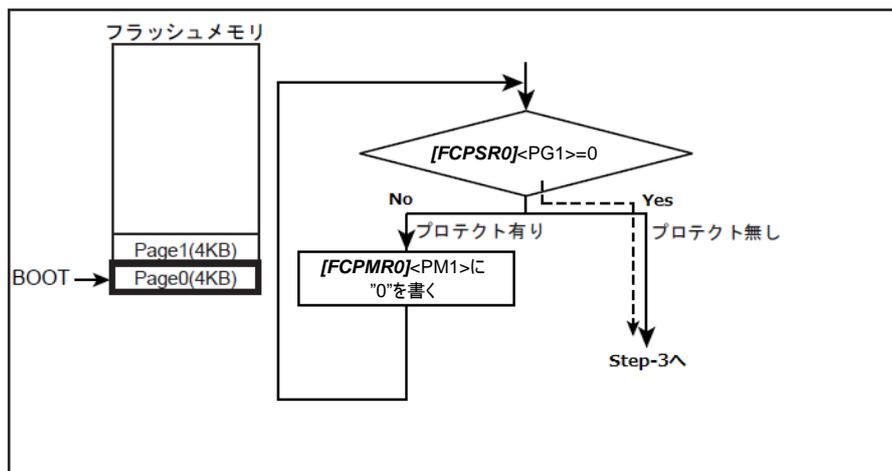


図 6.26 ユーザーブートプログラムの書き替え(2)

6.7.1.3. Step-3

内蔵RAMに書き替えルーチンを転送し、PC(プログラムカウンター)を転送したプログラムに移動します。

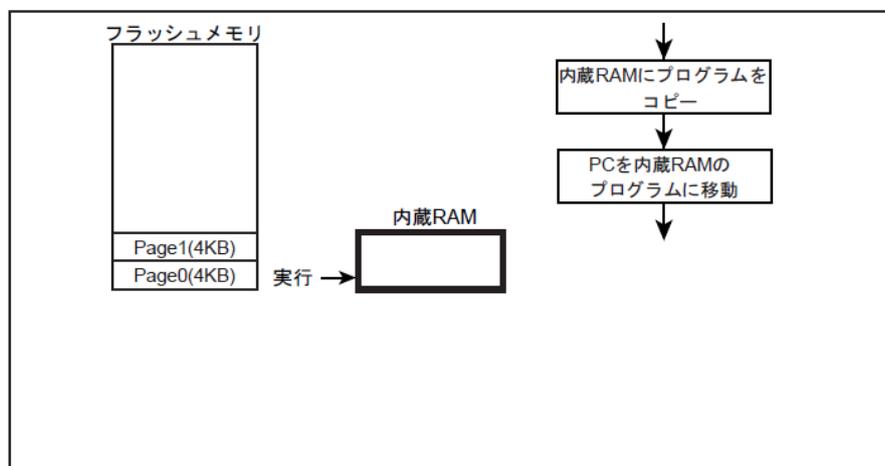


図 6.27 ユーザーブートプログラムの書き替え(3)

6.7.1.4. Step-4

Page1 を消去し、その後 Page0 のプログラムを Page1 に書き込みます。

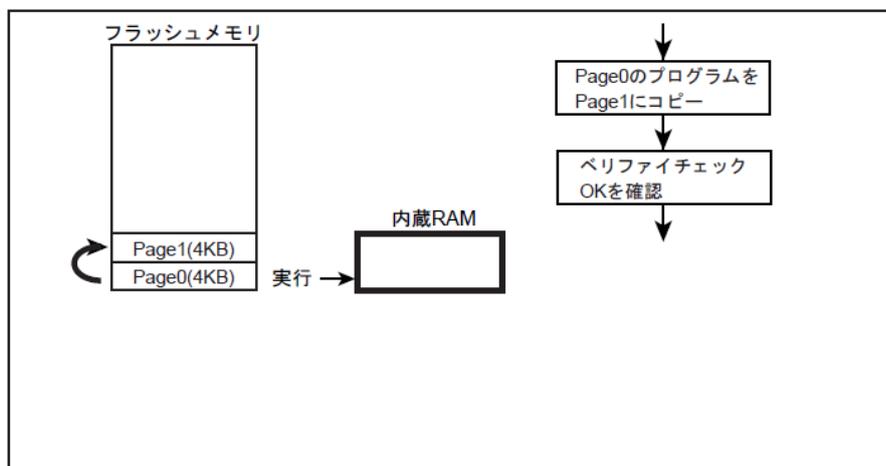


図 6.28 ユーザーブートプログラムの書き替え(4)

6.7.1.5. Step-5

自動メモリースワップコマンドで[FCSWPSR]<SWP[1:0]>に"01"をセットし、Page0 と Page1 をスワップします。

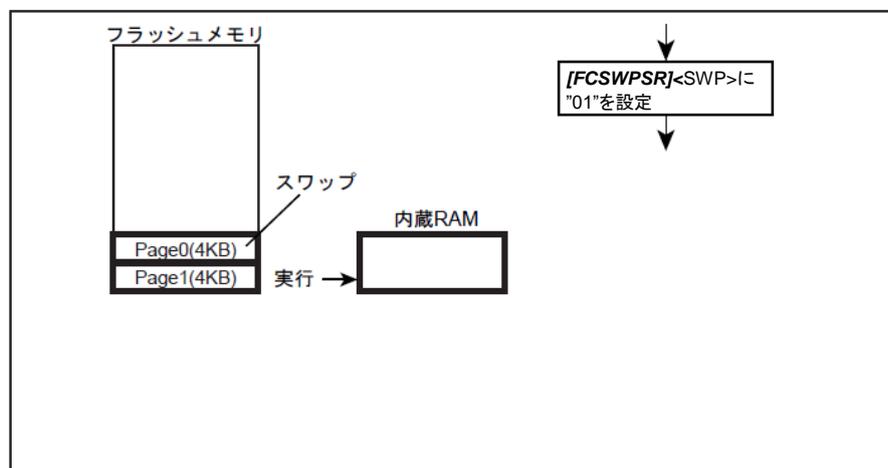


図 6.29 ユーザーブートプログラムの書き替え(5)

6.7.1.6. Step-6

リセット&リセット解除を行います。
Page1 が 0 番地に割り付けられ、Page1 から起動します。
プログラムは、 $[FCSWPSR]\langle SWP \rangle$ に "01" の条件用ルーチンへ分岐します。 ([Step-7]へ)

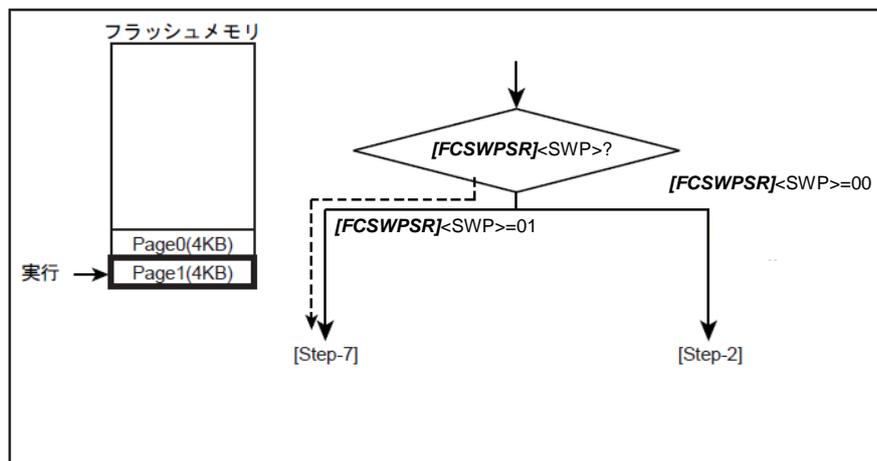


図 6.30 ユーザーブートプログラムの書き替え(6)

6.7.1.7. Step-7

$[FCPSR0]\langle PG1 \rangle = 0$ であるかチェックします。プロテクト状態がイネーブル ($\langle PG1 \rangle = 1$) の場合は、 $[FCPMR0]\langle PM1 \rangle$ に "0" を書いてプロテクトを一時解除してください。

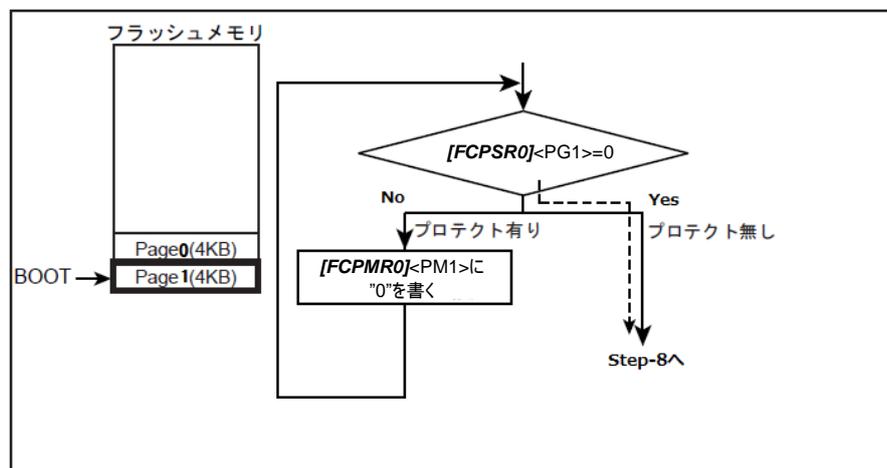


図 6.31 ユーザーブートプログラムの書き替え(7)

注) プロテクト機能はアドレスに対して有効です。したがって、PAGE0 と PAGE1 でメモリースワップ実行時は $\langle PG0 \rangle / \langle PM0 \rangle$ が PAGE1 に対応し、 $\langle PG1 \rangle / \langle PM1 \rangle$ が PAGE0 に対応します。

6.7.1.8. Step-8

内蔵RAMに書き替えルーチンを転送し、PC(プログラムカウンタ)を転送したプログラムに移動します。

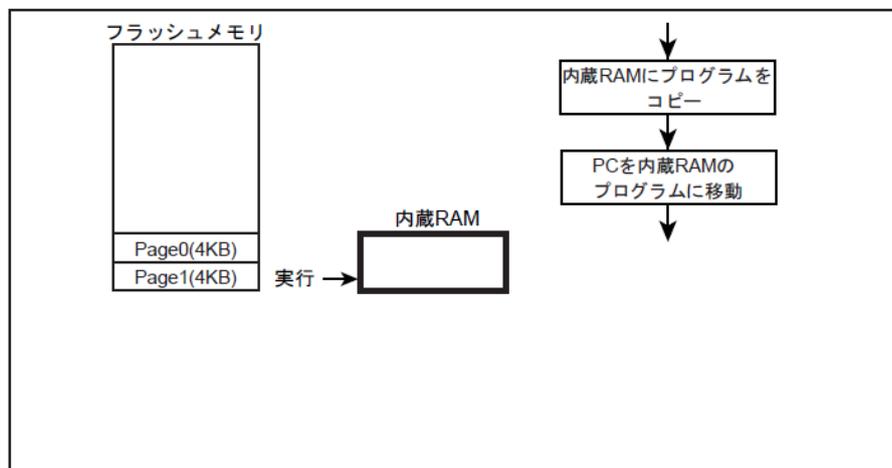


図 6.32 ユーザーブートプログラムの書き替え(8)

6.7.1.9. Step-9

新しいブートプログラムを Page0 に書き込みます。

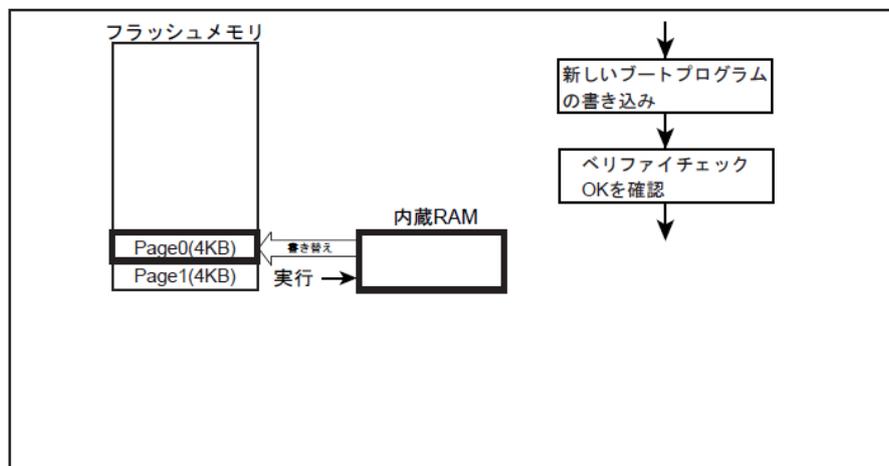


図 6.33 ユーザーブートプログラムの書き替え(9)

6.7.1.10. Step-10

自動メモリースワップ消去コマンドを実行します(下図)。または、自動メモリースワップコマンドで `[FCSWPSR]<SWP[1:0]>` に "11" をセットし、Page0 と Page1 をスワップ解除します。

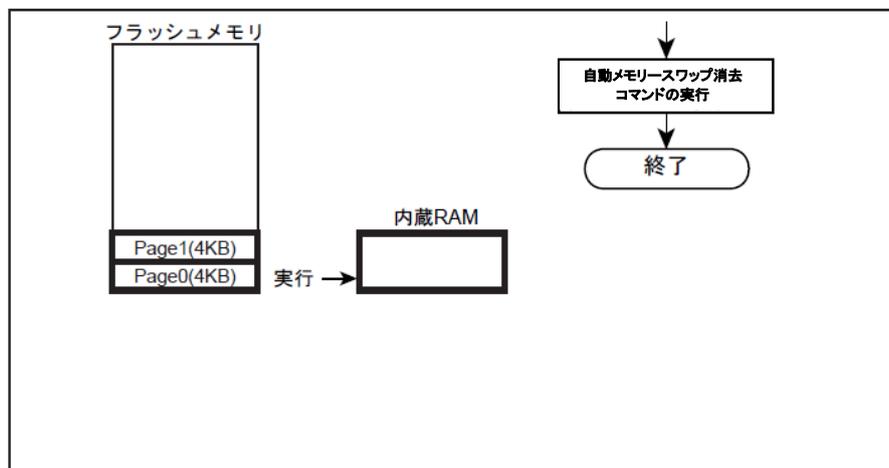


図 6.34 ユーザーブートプログラムの書き替え(10)

7. 使用上のご注意およびお願い事項

- 本ドキュメントに記載の無い操作を行わないでください。
- 本ドキュメントでレジスター割り当てが無いアドレスへのアクセスはしないでください。

8. 改訂履歴

表 8.1 改訂履歴

Revision	Date	Description
1.0	2017-11-01	新規
1.1	2018-05-07	<ul style="list-style-type: none"> ・表紙の構成変更。ヘッダー変更 ・1.概要:表 1.1 に注)を追加、表 1.2 の機能分離を変更 ・2.2.6.メモリー容量と構成:表 2.9 の消去時間修正“8.5ms”→“8.6ms”,“9.1ms”→“9.2ms” ・3.1.1.2.バースライトサイクル時のアドレスビット構成:表 3.2 参照先更新 ・3.2.1.自動プログラム:図 3.1 変更 ・3.2.2.自動消去:図 3.3 変更 ・3.2.3.~3.2.5.:章追加 ・4.1.2.コマンド実行方法:3と4.に“-自動メモリースワップ消去コマンド”を追加 ・4.1.3.8.自動セキュリティービットプログラム:(2)実行方法の説明修正“システムリセット後、セキュリティーは有効になります。…” ・4.1.3.9.自動セキュリティービット消去:(2)実行方法の説明修正“システムリセット後、セキュリティーは解除されます。” ・4.1.7.1/2:注)を追加 ・4.1.9.1/2/3 のタイトル:“ユーザーインフォメーションメモリー”→“ユーザーインフォメーションエリア” ・5.レジスター説明:表 5.1 の表題を削除、ベースアドレスに“Type1”を追加 ・5.1.レジスター一覧:章第追加 ・5.2.:“レジスター詳細”に変更 ・5.2.4.[FCSR0]:bit9,bit10 説明追加、<RDYBSY0>の説明修正 ・5.2.8.[FCPMR1]:bit31:8 リセット後“1”,説明修正“0xFFFFFFFF”→“1”,文末に“。”追加 ・5.2.10.[FCSWPSR]:<SWP>の記載順番逆転 ・5.2.11.[FCAREASEL]:bit15を独立、bit14:12を“000”に修正、文末に“。”追加、“書き込みモード”→“コマンドシーケンス入力モード”に修正 ・5.2.14.[FCBNKCR]:bit6:4 文末に“。”追加,説明追加 ・5.2.15.[FCBUFDISCLR]:“ユーザーインフォメーションメモリー”→“ユーザーインフォメーションエリア” ・6.6.6.2.~6.6.6.8.:“CHECK SUM”→“CHECKSUM” ・6.6.6.3.ACK 応答データ:図 6.16 の“チェックサム”→“CHECKSUM”に統一、(3)パスワードの設定/解除/認証の●パスワードの解除の“ユーザーインフォメーションメモリー”→“ユーザーインフォメーションエリア” ・6.6.8.RAM 転送コマンドの通信ルール:表 6.14 下の注)“RAM 等”→“RAM など” ・6.6.10.内蔵ブートプログラム全体フローチャート:図 6.18 の一部説明変更 ・6.7.1.5.Step-5:図 6.29 の右上の枠を削除
1.2	2019-03-18	<ul style="list-style-type: none"> ・1.1 メモリーマップ 図 1.1 RAM の追加 ・2.2.6 メモリー容量と構成 注 3)の条件に“ユーザーインフォメーションエリア”を追加 ・3. 機能説明・動作説明 使用上の注意点(1)(2) 内蔵高速発振器→内蔵高速発振器 1 に修正 ・3.2.3 プロテクトビット 図 3.6 左側フローの見出し 自動プログラムビット → 自動プロテクトビットに修正 ・3.2.5 メモリースワップ 図 3.10 フローの見出し メモリースワップ→自動メモリースワッププログラム ・4.1.6.3 プロテクトの一時解除機能 レジスター設定の参照先を「5.2 レジスターの詳細」に変更 ・5.2.4 [FCSR0] Bit15 のリセット後の値を“不定”に修正 ・6.6.3 インターフェース仕様 表 6.4 の下の注)を削除 ・6.6.4 メモリーの制約について 表 6.5 参照リファレンスマニュアルを追記 ・6.6.10 内蔵ブートプログラム全体フローチャート 図 6.18 UART 判定の記述を修正

		<ul style="list-style-type: none">・6.6.11 内蔵ブート ROM の書き替えアルゴリズムを利用した書き替え手順 UART を使った例であることを明記・6.6.11.6 Step-6 図 6.24 不要な説明を削除
1.3	2022-07-07	<ul style="list-style-type: none">・表 1.2 機能概要(ユーザーインフォメーションエリア) プログラム/消去プロテクト機能の削除

製品取り扱い上のお願い

株式会社東芝およびその子会社ならびに関係会社を以下「当社」といいます。

本資料に掲載されているハードウェア、ソフトウェアおよびシステムを以下「本製品」といいます。

- 本製品に関する情報等、本資料の掲載内容は、技術の進歩などにより予告なしに変更されることがあります。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。また、文書による当社の事前の承諾を得て本資料を転載複製する場合でも、記載内容に一切変更を加えたり、削除したりしないでください。
- 当社は品質、信頼性の向上に努めていますが、半導体・ストレージ製品は一般に誤作動または故障する場合があります。本製品をご使用頂く場合は、本製品の誤作動や故障により生命・身体・財産が侵害されることのないように、お客様の責任において、お客様のハードウェア・ソフトウェア・システムに必要な安全設計を行うことをお願いします。なお、設計および使用に際しては、本製品に関する最新の情報（本資料、仕様書、データシート、アプリケーションノート、半導体信頼性ハンドブックなど）および本製品が使用される機器の取扱説明書、操作説明書などをご確認の上、これに従ってください。また、上記資料などに記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を使用する場合は、お客様の製品単独およびシステム全体で十分に評価し、お客様の責任において適用可否を判断してください。
- 本製品は、特別に高い品質・信頼性が要求され、またはその故障や誤作動が生命・身体に危害を及ぼす恐れ、膨大な財産損害を引き起こす恐れ、もしくは社会に深刻な影響を及ぼす恐れのある機器（以下"特定用途"という）に使用されることは意図されていませんし、保証もされていません。特定用途には原子力関連機器、航空・宇宙機器、医療機器（ヘルスケア除く）、車載・輸送機器、列車・船舶機器、交通信号機器、燃焼・爆発制御機器、各種安全関連機器、昇降機器、発電関連機器などが含まれますが、本資料に個別に記載する用途は除きます。特定用途に使用された場合には、当社は一切の責任を負いません。なお、詳細は当社営業窓口まで、または当社 Web サイトのお問い合わせフォームからお問い合わせください。
- 本製品を分解、解析、リバースエンジニアリング、改造、改変、翻案、複製等しないでください。
- 本製品を、国内外の法令、規則及び命令により、製造、使用、販売を禁止されている製品に使用することはできません。
- 本資料に掲載してある技術情報は、製品の代表的動作・応用を説明するためのもので、その使用に際して当社及び第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。
- 別途、書面による契約またはお客様と当社が合意した仕様書がない限り、当社は、本製品および技術情報に関して、明示的にも黙示的にも一切の保証（機能動作の保証、商品性の保証、特定目的への合致の保証、情報の正確性の保証、第三者の権利の非侵害保証を含むがこれに限らない。）をしておりません。
- 本製品、または本資料に掲載されている技術情報を、大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」、「米国輸出管理規則」等、適用ある輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
- 本製品の RoHS 適合性など、詳細につきましては製品個別に必ず当社営業窓口までお問い合わせください。本製品のご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用ある環境関連法令を十分調査の上、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は一切の責任を負いかねます。