

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:1.5MB/1.0MB/768KB/512KB)

(データフラッシュ:32KB)

(FLASH15MHD32-A)

Revision 2.2

---

2022-07

東芝デバイス&ストレージ株式会社

## 目次

序章 .....	9
関連するドキュメント .....	9
表記規約 .....	10
用語・略語 .....	12
1. 概要 .....	13
1.1. メモリーマップ .....	15
2. 構成 .....	16
2.1. ブロック図 .....	16
2.2. コードフラッシュの構成 .....	17
2.2.1. コードフラッシュの構成単位 .....	17
2.2.2. コードフラッシュのブロック構成 .....	17
2.2.3. コードフラッシュのページ構成 .....	22
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 .....	25
2.2.5. コードフラッシュの書き込み、消去時間 .....	25
2.2.6. メモリー容量と構成 .....	26
2.3. データフラッシュの構成 .....	27
2.3.1. データフラッシュの構成単位 .....	27
2.3.2. データフラッシュのブロック構成 .....	27
2.3.3. データフラッシュのページ構成 .....	28
2.3.4. データフラッシュの書き込み、消去時間 .....	29
2.3.5. メモリー容量と構成 .....	29
3. 機能説明・動作説明 .....	30
3.1. コードフラッシュ .....	31
3.1.1. コードフラッシュのコマンドシーケンス .....	31
3.1.1.1. コードフラッシュのコマンドシーケンス一覧 .....	31
3.1.1.2. バスライトサイクル時のアドレスビット構成(コードフラッシュ) .....	33
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 .....	35
3.1.1.4. プロテクトビットの指定(PBA) : コードフラッシュ .....	35
3.1.1.5. ID-Read のコード(IA, ID) : コードフラッシュ .....	37
3.1.1.6. メモリースワップビットの指定(MSA) .....	37
3.2. データフラッシュ .....	38
3.2.1. データフラッシュのコマンドシーケンス .....	38
3.2.1.1. データフラッシュのコマンドシーケンス一覧 .....	38
3.2.1.2. バスライトサイクル時のアドレスビット構成(データフラッシュ) .....	39
3.2.1.3. エリアアドレス(AA)、ブロックアドレス(BA) .....	40
3.2.1.4. プロテクトビットの指定(PBA) .....	41
3.2.1.5. ID-Read のコード(IA, ID) : データフラッシュ .....	41
3.3. フローチャート .....	42

3.3.1. 自動プログラム.....	42
3.3.2. 自動消去.....	44
3.3.3. プロテクトビット.....	46
3.3.4. セキュリティービット.....	48
3.3.5. メモリースワップ.....	50
4. フラッシュメモリー詳細.....	52
4.1. 機能.....	52
4.1.1. フラッシュメモリーの動作モード.....	53
4.1.2. コマンド実行方法.....	53
4.1.3. コマンド説明.....	55
4.1.3.1. 自動プログラム.....	55
4.1.3.2. 自動チップ消去.....	56
4.1.3.3. 自動エリア消去.....	56
4.1.3.4. 自動ブロック消去.....	57
4.1.3.5. 自動ページ消去.....	57
4.1.3.6. 自動プロテクトビットプログラム.....	57
4.1.3.7. 自動プロテクトビット消去.....	58
4.1.3.8. 自動セキュリティービットプログラム.....	58
4.1.3.9. 自動セキュリティービット消去.....	58
4.1.3.10. ID-Read.....	59
4.1.3.11. Read/リセットコマンド.....	59
4.1.3.12. 自動メモリースワッププログラム.....	60
4.1.3.13. 自動メモリースワップ消去.....	60
4.1.4. 自動チップ消去動作の中止.....	61
4.1.5. 自動動作の完了検知.....	61
4.1.5.1. 手順.....	61
4.1.6. プロテクト機能.....	62
4.1.6.1. プロテクトの設定方法.....	62
4.1.6.2. プロテクトの解除方法.....	62
4.1.6.3. プロテクトの一時解除機能.....	62
4.1.7. セキュリティー機能.....	63
4.1.7.1. セキュリティーの設定方法.....	63
4.1.7.2. セキュリティーの解除方法.....	63
4.1.7.3. 動作.....	63
4.1.8. メモリースワップ機能.....	64
4.1.8.1. メモリースワップの設定方法.....	64
4.1.8.2. 操作方法.....	64
4.1.8.3. メモリースワップ情報の消去.....	65
4.1.9. ユーザーインフォメーションエリア.....	66
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.....	66
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.....	66

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.....	66
5. レジスター説明.....	67
5.1. レジスター一覧.....	67
5.2. レジスター詳細.....	68
5.2.1. [FCSBMR] (Flash Security Bit Mask Register) .....	68
5.2.2. [FCSSR] (Flash Security Status Register).....	68
5.2.3. [FCKCR] (Flash Key Code Register).....	68
5.2.4. [FCSR0] (Flash Status Register 0).....	69
5.2.5. [FCPSR0] (Flash Protect Status Register 0) .....	70
5.2.6. [FCPSR1] (Flash Protect Status Register 1) .....	71
5.2.7. [FCPSR3] (Flash Protect Status Register 3) .....	72
5.2.8. [FCPSR4] (Flash Protect Status Register 4) .....	72
5.2.9. [FCPSR6] (Flash Protect Status Register 6) .....	73
5.2.10. [FCPMR0] (Flash Protect Mask Register 0) .....	73
5.2.11. [FCPMR1] (Flash Protect Mask Register 1) .....	74
5.2.12. [FCPMR3] (Flash Protect Mask Register 3) .....	75
5.2.13. [FCPMR4] (Flash Protect Mask Register 4) .....	76
5.2.14. [FCPMR6] (Flash Protect Mask Register 6) .....	77
5.2.15. [FCSR1] (Flash Status Register 1) .....	77
5.2.16. [FCWPSR] (Flash Memory SWAP Status Register).....	78
5.2.17. [FCAREASEL] (Flash Area Selection Register).....	79
5.2.18. [FCCR] (Flash Control Register).....	80
5.2.19. [FCSTCLR] (Flash Status Clear Register) .....	80
5.2.20. [FCBNKCR] (Flash Bank Change Register) .....	81
5.2.21. [FCBUFDISCLR] (Flash Buffer Disable and Clear Register) .....	81
6. プログラミング方法.....	82
6.1. 初期化.....	82
6.2. モードの説明.....	82
6.3. モードの決定.....	83
6.4. モードごとのメモリーマップ .....	83
6.5. 書き替え方法.....	84
6.5.1. (1-A)書き替えルーチンをフラッシュメモリに内蔵する場合の手順例.....	84
6.5.1.1. Step-1 .....	84
6.5.1.2. Step-2 .....	85
6.5.1.3. Step-3 .....	85
6.5.1.4. Step-4 .....	86
6.5.1.5. Step-5 .....	86
6.5.1.6. Step-6 .....	87
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.....	87
6.5.2.1. Step-1 .....	87
6.5.2.2. Step-2 .....	88
6.5.2.3. Step-3 .....	88
6.5.2.4. Step-4 .....	89

6.5.2.5. Step-5 .....	89
6.5.2.6. Step-6 .....	90
6.6. シングルブートモードによる書き替え方法.....	91
6.6.1. 概要 .....	91
6.6.2. モード設定.....	92
6.6.3. インターフェース仕様.....	92
6.6.4. メモリーの制約について .....	93
6.6.5. 動作コマンド .....	93
6.6.5.1. RAM 転送.....	93
6.6.5.2. フラッシュメモリ消去 .....	93
6.6.6. コマンドによらず共通の動作 .....	94
6.6.6.1. シリアル動作モード判定 .....	94
6.6.6.2. ACK 応答データ .....	96
6.6.6.3. パスワード .....	97
6.6.6.4. パスワード判定 .....	100
6.6.6.5. CHECKSUM の計算方法 .....	100
6.6.7. シリアル動作モード判定の通信ルール.....	101
6.6.8. RAM 転送コマンドの通信ルール.....	102
6.6.9. フラッシュメモリ消去の通信ルール.....	104
6.6.10. 内蔵ブートプログラム全体フローチャート.....	105
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 .....	106
6.6.11.1. Step-1 .....	106
6.6.11.2. Step-2 .....	106
6.6.11.3. Step-3 .....	107
6.6.11.4. Step-4 .....	107
6.6.11.5. Step-5 .....	108
6.6.11.6. Step-6 .....	108
6.7. デュアルモードによる書き替え方法 .....	109
6.7.1. フラッシュ書き替えの手順例 .....	109
6.7.1.1. Step-1 .....	109
6.7.1.2. Step-2 .....	110
6.7.1.3. Step-3 .....	110
6.7.1.4. Step-4 .....	111
6.7.1.5. Step-5 .....	111
6.8. ユーザーブートプログラムの書き替え方法.....	112
6.8.1. フラッシュ書き替えの手順例 .....	112
6.8.1.1. Step-1 .....	112
6.8.1.2. Step-2 .....	113
6.8.1.3. Step-3 .....	113
6.8.1.4. Step-4 .....	114
6.8.1.5. Step-5 .....	114
6.8.1.6. Step-6 .....	115
6.8.1.7. Step-7 .....	115

---

6.8.1.8. Step-8 .....	116
6.8.1.9. Step-9 .....	116
6.8.1.10. Step-10 .....	117
7. 使用上のご注意およびお願い事項.....	118
8. 改訂履歴.....	119
製品取り扱い上のお願い.....	122

## 目次

図 1.1	メモリーマップ例	15
図 2.1	フラッシュメモリーのブロック図	16
図 3.1	自動プログラムのフロー(1)	42
図 3.2	自動プログラムのフロー(2)	43
図 3.3	自動消去のフロー(1)	44
図 3.4	自動消去のフロー(2)	45
図 3.5	プロテクトのフロー(1)	46
図 3.6	プロテクトのフロー(2)	47
図 3.7	セキュリティーのフロー(1)	48
図 3.8	セキュリティーのフロー(2)	49
図 3.9	メモリースワップのフロー(1)	50
図 3.10	メモリースワップのフロー(2)	51
図 4.1	メモリースワップ実施手順の例	65
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)	84
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)	85
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)	85
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)	86
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)	86
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)	87
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)	87
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)	88
図 6.9	書き替えルーチンを外部から転送場合の手順(3)	88
図 6.10	書き替えルーチンを外部から転送場合の手順(4)	89
図 6.11	書き替えルーチンを外部から転送場合の手順(5)	89
図 6.12	書き替えルーチンを外部から転送場合の手順(6)	90
図 6.13	シリアル動作モード判定データ	94
図 6.14	シリアル動作モード受信フローチャート	95
図 6.15	シリアル動作モード判定フローチャート	96
図 6.16	パスワードの構成(送信例)	98
図 6.17	パスワードチェックフローチャート	100
図 6.18	内蔵ブートプログラム全体フローチャート	105
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)	106
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)	106
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)	107
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)	107
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)	108
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)	108
図 6.25	デュアルモードでフラッシュ書き替えの手順(1)	109
図 6.26	デュアルモードでフラッシュ書き替えの手順(2)	110
図 6.27	デュアルモードでフラッシュ書き替えの手順(3)	110
図 6.28	デュアルモードでフラッシュ書き替えの手順(4)	111
図 6.29	デュアルモードでフラッシュ書き替えの手順(5)	111
図 6.30	ユーザーブートプログラムの書き替え(1)	112
図 6.31	ユーザーブートプログラムの書き替え(2)	113
図 6.32	ユーザーブートプログラムの書き替え(3)	113
図 6.33	ユーザーブートプログラムの書き替え(4)	114
図 6.34	ユーザーブートプログラムの書き替え(5)	114
図 6.35	ユーザーブートプログラムの書き替え(6)	115
図 6.36	ユーザーブートプログラムの書き替え(7)	115
図 6.37	ユーザーブートプログラムの書き替え(8)	116
図 6.38	ユーザーブートプログラムの書き替え(9)	116

図 6.39 ユーザーブートプログラムの書き替え(10)..... 117

## 表目次

表 1.1	機能概要(コードフラッシュ).....	13
表 1.2	機能概要(ユーザーインフォメーションエリア).....	14
表 1.3	機能概要(データフラッシュ).....	14
表 2.1	信号一覧.....	16
表 2.2	1536KB コードフラッシュのブロック構成.....	17
表 2.3	1024KB コードフラッシュのブロック構成.....	19
表 2.4	768KB コードフラッシュのブロック構成.....	20
表 2.5	512KB コードフラッシュのブロック構成.....	21
表 2.6	1536KB コードフラッシュのページ構成.....	22
表 2.7	1024KB コードフラッシュのページ構成.....	23
表 2.8	768KB コードフラッシュのページ構成.....	24
表 2.9	512KB コードフラッシュのページ構成.....	25
表 2.10	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.....	25
表 2.11	コードフラッシュのメモリー容量と構成.....	26
表 2.12	32KB データフラッシュのブロック構成.....	27
表 2.13	32KB データフラッシュのページ構成.....	28
表 2.14	データフラッシュのメモリー容量と構成.....	29
表 3.1	JEDEC 準拠機能.....	30
表 3.2	内部 CPU によるフラッシュメモリーアクセス (コードフラッシュ).....	31
表 3.3	バスライトサイクル時のアドレスビット構成(コードフラッシュ).....	33
表 3.4	プロテクトビットプログラムアドレス表.....	35
表 3.5	ID-Read コマンドのコード指定とコードの内容(コードフラッシュ).....	37
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.....	37
表 3.7	コマンドシーケンス(データフラッシュ).....	38
表 3.8	バスライトサイクル時のアドレスビット構成(データフラッシュ).....	39
表 3.9	プロテクトビットプログラムアドレス表(データフラッシュ).....	41
表 3.10	ID-Read コマンドのコード指定とコードの内容 (データフラッシュ).....	41
表 4.1	フラッシュメモリーの機能.....	52
表 4.2	フラッシュの書き込み/消去動作などの完了検知.....	61
表 4.3	セキュリティ有効時の動作.....	63
表 6.1	モードと動作.....	82
表 6.2	動作モード設定表.....	83
表 6.3	機能とコマンド.....	91
表 6.4	使用端子例.....	92
表 6.5	シングルブート時のメモリーの制約.....	93
表 6.6	シングルブート時コマンド.....	93
表 6.7	転送可能なボーレートの設定例(fc=10MHz、誤差含まず).....	94
表 6.8	シリアル動作判定データに対する ACK 応答データ.....	96
表 6.9	動作コマンドデータに対する ACK 応答データ.....	96
表 6.10	CHECKSUM データに対する ACK 応答データ.....	96
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.....	97
表 6.12	パスワードの設定値、設定範囲.....	99
表 6.13	シリアル動作モード判定の通信ルール.....	101
表 6.14	RAM 転送コマンドの通信ルール.....	102
表 6.15	フラッシュメモリー消去の通信ルール.....	104
表 8.1	改訂履歴.....	119



## 序章

### 関連するドキュメント

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
POR	Power On Reset
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とデータを保存するデータ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。

データフラッシュはデータを格納し、電源が遮断されてもデータを保持します。

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 1.5MB 1.0MB 768KB 512KB	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的全領域の消去を自動で行います。 対象: コードフラッシュ データフラッシュ	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	
	異なる FLASH I/F に対する書き込み/消去	コードフラッシュ/データフラッシュに対する書き込み/消去	コードフラッシュ/データフラッシュに対する基本動作ができます。	デュアルモード

注) FLASH I/F ごとの先頭 32KB はページ単位でプロテクトします。

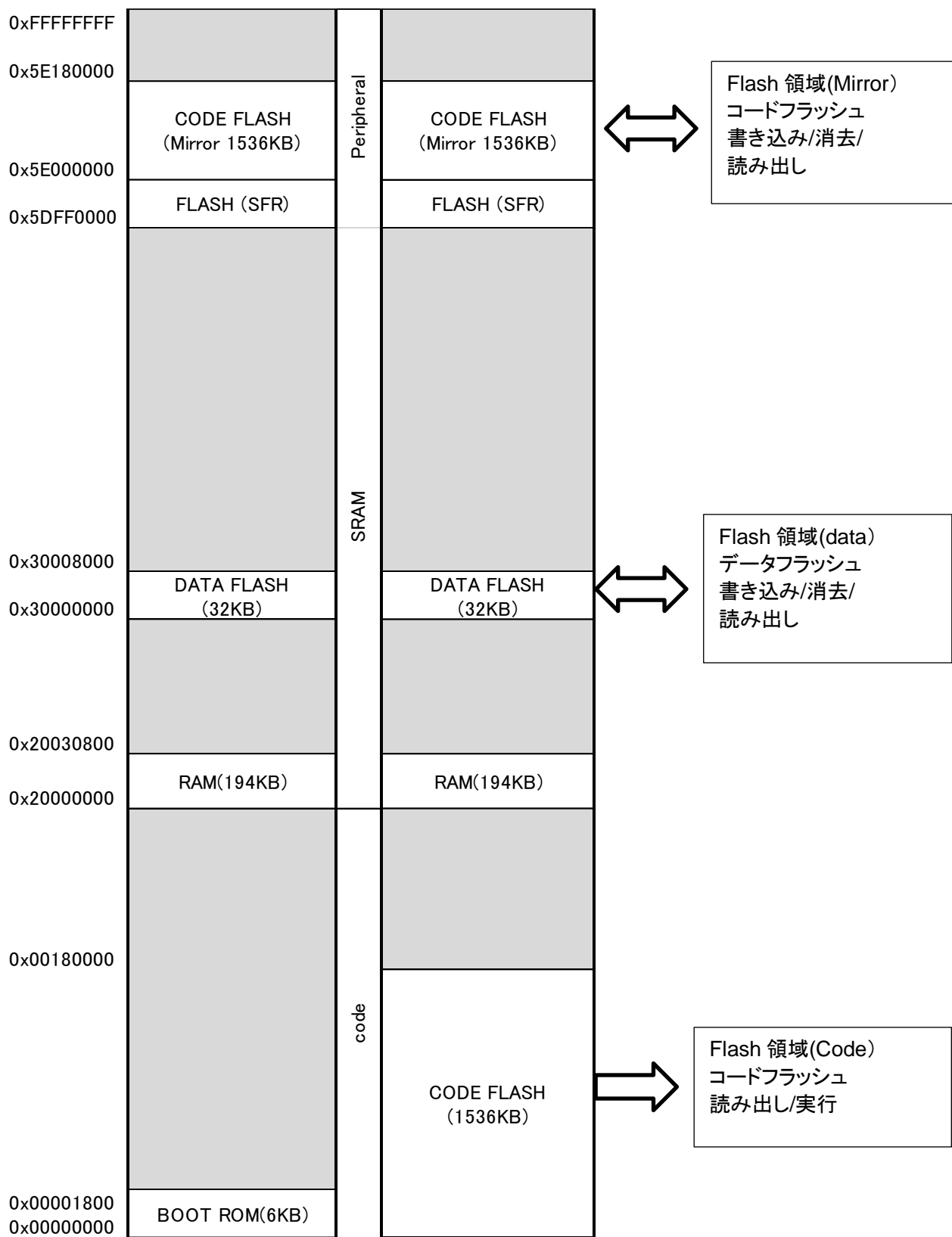
表 1.2 機能概要(ユーザーインフォメーションエリア)

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。

表 1.3 機能概要(データフラッシュ)

領域	機能	基本動作	動作	備考
データフラッシュ 32KB	フラッシュプログラム/消去動作	自動プログラム	1ワード(4バイト)単位でデータ書き込みを自動で行います。	
		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。	
	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	命令の実行が可能です。	プリフェッチバッファはありませんので、ウエイトがかかります。
	異なる FLASH I/F に対する書き込み/消去	コードフラッシュに対する書き込み/消去	コードフラッシュに対する基本動作ができます。	デュアルモード

## 1.1. メモリーマップ



Single Boot Mode                      Single Chip Mode

図 1.1 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。

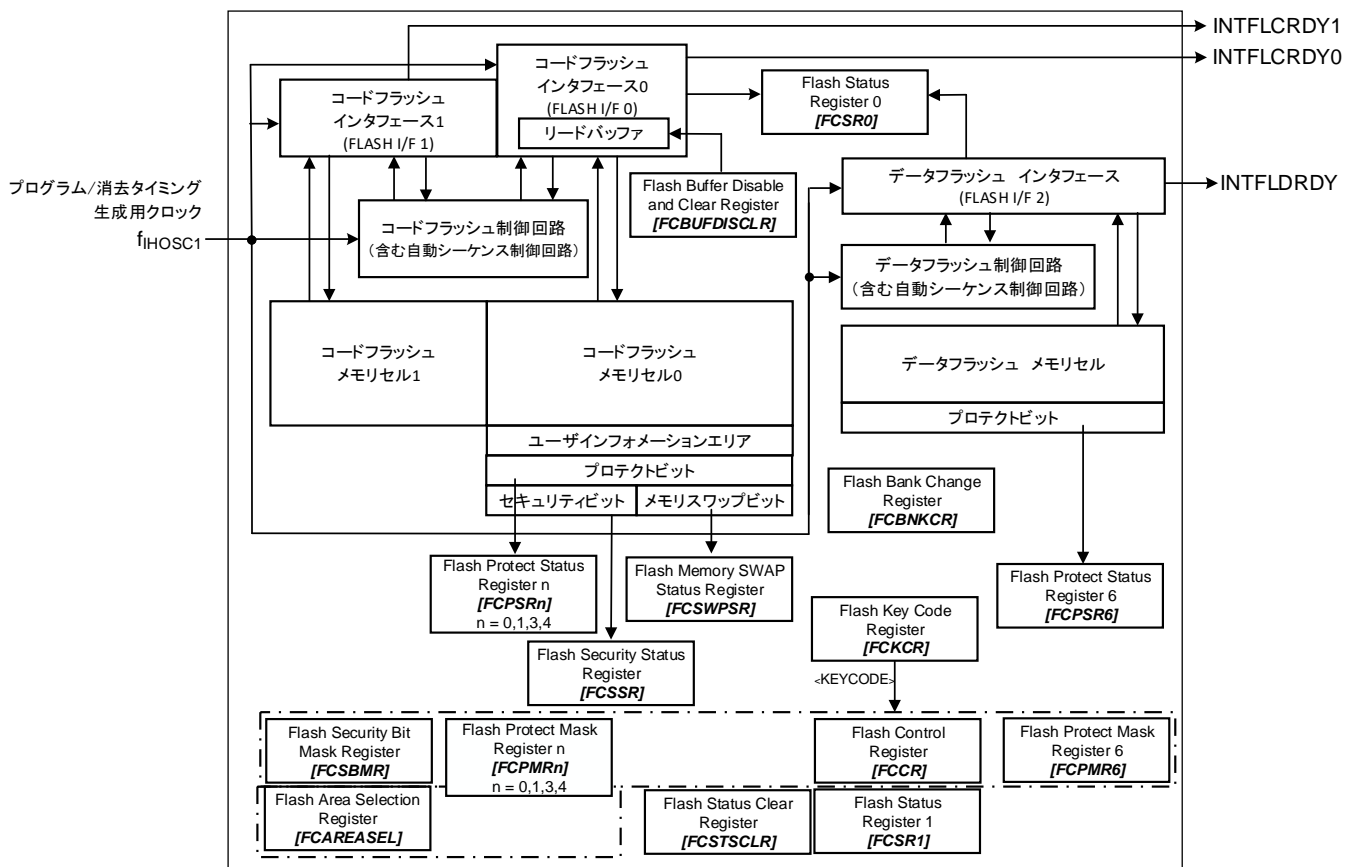


図 2.1 フラッシュメモリのブロック図

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	fIHOSC1	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	INTFLCRDY0	FLASH I/F0 Code FLASH Ready 割り込み 0	出力	例外
3	INTFLCRDY1	FLASH I/F1 Code FLASH Ready 割り込み 1	出力	例外
4	INTFLDRDY	Data 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 512KB 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 1536KBコードフラッシュのブロック構成

FLASH I/F	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 (K バイト)	
0	0	Block0	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	
		Block8	0x00040000-0x00047FFF	0x5E040000-0x5E047FFF	32	
Block9	0x00048000-0x0004FFFF	0x5E048000-0x5E04FFFF	32			
Block10	0x00050000-0x00057FFF	0x5E050000-0x5E057FFF	32			
Block11	0x00058000-0x0005FFFF	0x5E058000-0x5E05FFFF	32			
Block12	0x00060000-0x00067FFF	0x5E060000-0x5E067FFF	32			
Block13	0x00068000-0x0006FFFF	0x5E068000-0x5E06FFFF	32			
Block14	0x00070000-0x00077FFF	0x5E070000-0x5E077FFF	32			
Block15	0x00078000-0x0007FFFF	0x5E078000-0x5E07FFFF	32			

1	1	Block16	0x00080000-0x00087FFF	0x5E080000-0x5E087FFF	32	
		Block17	0x00088000-0x0008FFFF	0x5E088000-0x5E08FFFF	32	
		Block18	0x00090000-0x00097FFF	0x5E090000-0x5E097FFF	32	
		Block19	0x00098000-0x0009FFFF	0x5E098000-0x5E09FFFF	32	
		Block20	0x000A0000-0x000A7FFF	0x5E0A0000-0x5E0A7FFF	32	
		Block21	0x000A8000-0x000AFFFF	0x5E0A8000-0x5E0AFFFF	32	
		Block22	0x000B0000-0x000B7FFF	0x5E0B0000-0x5E0B7FFF	32	
		Block23	0x000B8000-0x000BFFFF	0x5E0B8000-0x5E0BFFFF	32	
		Block24	0x000C0000-0x000C7FFF	0x5E0C0000-0x5E0C7FFF	32	
		Block25	0x000C8000-0x000CFFFF	0x5E0C8000-0x5E0CFFFF	32	
		Block26	0x000D0000-0x000D7FFF	0x5E0D0000-0x5E0D7FFF	32	
		Block27	0x000D8000-0x000DFFFF	0x5E0D8000-0x5E0DFFFF	32	
		Block28	0x000E0000-0x000E7FFF	0x5E0E0000-0x5E0E7FFF	32	
		Block29	0x000E8000-0x000EFFFF	0x5E0E8000-0x5E0EFFFF	32	
		Block30	0x000F0000-0x000F7FFF	0x5E0F0000-0x5E0F7FFF	32	
		Block31	0x000F8000-0x000FFFFF	0x5E0F8000-0x5E0FFFFF	32	
1	2	Block0	PG0	0x00100000-0x00100FFF	0x5E100000-0x5E100FFF	4
			PG1	0x00101000-0x00101FFF	0x5E101000-0x5E101FFF	4
			PG2	0x00102000-0x00102FFF	0x5E102000-0x5E102FFF	4
			PG3	0x00103000-0x00103FFF	0x5E103000-0x5E103FFF	4
			PG4	0x00104000-0x00104FFF	0x5E104000-0x5E104FFF	4
			PG5	0x00105000-0x00105FFF	0x5E105000-0x5E105FFF	4
			PG6	0x00106000-0x00106FFF	0x5E106000-0x5E106FFF	4
			PG7	0x00107000-0x00107FFF	0x5E107000-0x5E107FFF	4
	Block1	0x00108000-0x0010FFFF	0x5E108000-0x5E10FFFF	32		
	Block2	0x00110000-0x00117FFF	0x5E110000-0x5E117FFF	32		
	Block3	0x00118000-0x0011FFFF	0x5E118000-0x5E11FFFF	32		
	Block4	0x00120000-0x00127FFF	0x5E120000-0x5E127FFF	32		
	Block5	0x00128000-0x0012FFFF	0x5E128000-0x5E12FFFF	32		
	Block6	0x00130000-0x00137FFF	0x5E130000-0x5E137FFF	32		
	Block7	0x00138000-0x0013FFFF	0x5E138000-0x5E13FFFF	32		
	Block8	0x00140000-0x00147FFF	0x5E140000-0x5E147FFF	32		
	Block9	0x00148000-0x0014FFFF	0x5E148000-0x5E14FFFF	32		
	Block10	0x00150000-0x00157FFF	0x5E150000-0x5E157FFF	32		
	Block11	0x00158000-0x0015FFFF	0x5E158000-0x5E15FFFF	32		
	Block12	0x00160000-0x00167FFF	0x5E160000-0x5E167FFF	32		
Block13	0x00168000-0x0016FFFF	0x5E168000-0x5E16FFFF	32			
Block14	0x00170000-0x00177FFF	0x5E170000-0x5E177FFF	32			
Block15	0x00178000-0x0017FFFF	0x5E178000-0x5E17FFFF	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 1024KBコードフラッシュのブロック構成

FLASH I/F	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 (K バイト)	
0	0	Block0	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	
		Block8	0x00040000-0x00047FFF	0x5E040000-0x5E047FFF	32	
	Block9	0x00048000-0x0004FFFF	0x5E048000-0x5E04FFFF	32		
	Block10	0x00050000-0x00057FFF	0x5E050000-0x5E057FFF	32		
	Block11	0x00058000-0x0005FFFF	0x5E058000-0x5E05FFFF	32		
	Block12	0x00060000-0x00067FFF	0x5E060000-0x5E067FFF	32		
	Block13	0x00068000-0x0006FFFF	0x5E068000-0x5E06FFFF	32		
	Block14	0x00070000-0x00077FFF	0x5E070000-0x5E077FFF	32		
	Block15	0x00078000-0x0007FFFF	0x5E078000-0x5E07FFFF	32		
	1	Block16	0x00080000-0x00087FFF	0x5E080000-0x5E087FFF	32	
		Block17	0x00088000-0x0008FFFF	0x5E088000-0x5E08FFFF	32	
		Block18	0x00090000-0x00097FFF	0x5E090000-0x5E097FFF	32	
		Block19	0x00098000-0x0009FFFF	0x5E098000-0x5E09FFFF	32	
		Block20	0x000A0000-0x000A7FFF	0x5E0A0000-0x5E0A7FFF	32	
		Block21	0x000A8000-0x000AFFFF	0x5E0A8000-0x5E0AFFFF	32	
		Block22	0x000B0000-0x000B7FFF	0x5E0B0000-0x5E0B7FFF	32	
		Block23	0x000B8000-0x000BFFFF	0x5E0B8000-0x5E0BFFFF	32	
		Block24	0x000C0000-0x000C7FFF	0x5E0C0000-0x5E0C7FFF	32	
		Block25	0x000C8000-0x000CFFFF	0x5E0C8000-0x5E0CFFFF	32	
		Block26	0x000D0000-0x000D7FFF	0x5E0D0000-0x5E0D7FFF	32	
Block27		0x000D8000-0x000DFFFF	0x5E0D8000-0x5E0DFFFF	32		
Block28		0x000E0000-0x000E7FFF	0x5E0E0000-0x5E0E7FFF	32		
Block29		0x000E8000-0x000EFFFF	0x5E0E8000-0x5E0EFFFF	32		
Block30		0x000F0000-0x000F7FFF	0x5E0F0000-0x5E0F7FFF	32		
Block31		0x000F8000-0x000FFFFF	0x5E0F8000-0x5E0FFFFF	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 768KBコードフラッシュのブロック構成

FLASH I/F	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 (K バイト)	
0	0	Block0	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	
	Block8	0x00040000-0x00047FFF	0x5E040000-0x5E047FFF	32		
	Block9	0x00048000-0x0004FFFF	0x5E048000-0x5E04FFFF	32		
	Block10	0x00050000-0x00057FFF	0x5E050000-0x5E057FFF	32		
	Block11	0x00058000-0x0005FFFF	0x5E058000-0x5E05FFFF	32		
	Block12	0x00060000-0x00067FFF	0x5E060000-0x5E067FFF	32		
	Block13	0x00068000-0x0006FFFF	0x5E068000-0x5E06FFFF	32		
	Block14	0x00070000-0x00077FFF	0x5E070000-0x5E077FFF	32		
	Block15	0x00078000-0x0007FFFF	0x5E078000-0x5E07FFFF	32		
1	Block16	0x00080000-0x00087FFF	0x5E080000-0x5E087FFF	32		
	Block17	0x00088000-0x0008FFFF	0x5E088000-0x5E08FFFF	32		
	Block18	0x00090000-0x00097FFF	0x5E090000-0x5E097FFF	32		
	Block19	0x00098000-0x0009FFFF	0x5E098000-0x5E09FFFF	32		
	Block20	0x000A0000-0x000A7FFF	0x5E0A0000-0x5E0A7FFF	32		
	Block21	0x000A8000-0x000AFFFF	0x5E0A8000-0x5E0AFFFF	32		
	Block22	0x000B0000-0x000B7FFF	0x5E0B0000-0x5E0B7FFF	32		
	Block23	0x000B8000-0x000BFFFF	0x5E0B8000-0x5E0BFFFF	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 512KBコードフラッシュのブロック構成

FLASH I/F	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 (K バイト)	
0	0	Block0	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	
		Block8	0x00040000-0x00047FFF	0x5E040000-0x5E047FFF	32	
Block9	0x00048000-0x0004FFFF	0x5E048000-0x5E04FFFF	32			
Block10	0x00050000-0x00057FFF	0x5E050000-0x5E057FFF	32			
Block11	0x00058000-0x0005FFFF	0x5E058000-0x5E05FFFF	32			
Block12	0x00060000-0x00067FFF	0x5E060000-0x5E067FFF	32			
Block13	0x00068000-0x0006FFFF	0x5E068000-0x5E06FFFF	32			
Block14	0x00070000-0x00077FFF	0x5E070000-0x5E077FFF	32			
Block15	0x00078000-0x0007FFFF	0x5E078000-0x5E07FFFF	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

コードフラッシュの例を以下に示します。

表 2.6 1536KBコードフラッシュのページ構成

FLASH I/F	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	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	
		⋮	⋮	⋮	⋮	⋮
		Page124	0x0007C000-0x0007CFFF	0x5E07C000-0x5E07CFFF	4	
		Page125	0x0007D000-0x0007DFFF	0x5E07D000-0x5E07DFFF	4	
		Page126	0x0007E000-0x0007EFFF	0x5E07E000-0x5E07EFFF	4	
		Page127	0x0007F000-0x0007FFFF	0x5E07F000-0x5E07FFFF	4	
		1	Page128	0x00080000-0x00080FFF	0x5E080000-0x5E080FFF	4
	Page129		0x00081000-0x00081FFF	0x5E081000-0x5E081FFF	4	
	Page130		0x00082000-0x00082FFF	0x5E082000-0x5E082FFF	4	
	Page131		0x00083000-0x00083FFF	0x5E083000-0x5E083FFF	4	
	Page132		0x00084000-0x00084FFF	0x5E084000-0x5E084FFF	4	
	Page133		0x00085000-0x00085FFF	0x5E085000-0x5E085FFF	4	
	Page134		0x00086000-0x00086FFF	0x5E086000-0x5E086FFF	4	
	Page135		0x00087000-0x00087FFF	0x5E087000-0x5E087FFF	4	
	⋮		⋮	⋮	⋮	⋮
	Page252		0x000FC000-0x000FCFFF	0x5E0FC000-0x5E0FCFFF	4	
	Page253		0x000FD000-0x000FDFFF	0x5E0FD000-0x5E0FDFFF	4	
	Page254		0x000FE000-0x000FEFFF	0x5E0FE000-0x5E0FEFFF	4	
	Page255		0x000FF000-0x000FFFFF	0x5E0FF000-0x5E0FFFFF	4	
	1		2	Page0	0x00100000-0x00100FFF	0x5E100000-0x5E100FFF
		Page1		0x00101000-0x00101FFF	0x5E101000-0x5E101FFF	4
Page2		0x00102000-0x00102FFF		0x5E102000-0x5E102FFF	4	
Page3		0x00103000-0x00103FFF		0x5E103000-0x5E103FFF	4	
Page4		0x00104000-0x00104FFF		0x5E104000-0x5E104FFF	4	
Page5		0x00105000-0x00105FFF		0x5E105000-0x5E105FFF	4	
Page6		0x00106000-0x00106FFF		0x5E106000-0x5E106FFF	4	
Page7		0x00107000-0x00107FFF		0x5E107000-0x5E107FFF	4	
⋮		⋮		⋮	⋮	⋮
Page124		0x0017C000-0x0017CFFF		0x5E17C000-0x5E17CFFF	4	
Page125		0x0017D000-0x0017DFFF		0x5E17D000-0x5E17DFFF	4	
Page126		0x0017E000-0x0017EFFF		0x5E17E000-0x5E17EFFF	4	
Page127		0x0017F000-0x0017FFFF		0x5E17F000-0x5E17FFFF	4	

表 2.7 1024KBコードフラッシュのページ構成

FLASH I/F	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	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	
		⋮	⋮	⋮	⋮	⋮
		Page124	0x0007C000-0x0007CFFF	0x5E07C000-0x5E07CFFF	4	
		Page125	0x0007D000-0x0007DFFF	0x5E07D000-0x5E07DFFF	4	
		Page126	0x0007E000-0x0007EFFF	0x5E07E000-0x5E07EFFF	4	
		Page127	0x0007F000-0x0007FFFF	0x5E07F000-0x5E07FFFF	4	
		Page128	0x00080000-0x00080FFF	0x5E080000-0x5E080FFF	4	
	Page129	0x00081000-0x00081FFF	0x5E081000-0x5E081FFF	4		
	Page130	0x00082000-0x00082FFF	0x5E082000-0x5E082FFF	4		
	Page131	0x00083000-0x00083FFF	0x5E083000-0x5E083FFF	4		
	Page132	0x00084000-0x00084FFF	0x5E084000-0x5E084FFF	4		
	Page133	0x00085000-0x00085FFF	0x5E085000-0x5E085FFF	4		
	Page134	0x00086000-0x00086FFF	0x5E086000-0x5E086FFF	4		
	Page135	0x00087000-0x00087FFF	0x5E087000-0x5E087FFF	4		
	⋮	⋮	⋮	⋮	⋮	
	Page252	0x000FC000-0x000FCFFF	0x5E0FC000-0x5E0FCFFF	4		
	Page253	0x000FD000-0x000FDFFF	0x5E0FD000-0x5E0FDFFF	4		
	Page254	0x000FE000-0x000FEFFF	0x5E0FE000-0x5E0FEFFF	4		
	Page255	0x000FF000-0x000FFFFF	0x5E0FF000-0x5E0FFFFF	4		

表 2.8 768KBコードフラッシュのページ構成

FLASH I/F	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	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
		⋮	⋮	⋮	⋮
		⋮	⋮	⋮	⋮
		Page124	0x0007C000-0x0007CFFF	0x5E07C000-0x5E07CFFF	4
		Page125	0x0007D000-0x0007DFFF	0x5E07D000-0x5E07DFFF	4
		Page126	0x0007E000-0x0007EFFF	0x5E07E000-0x5E07EFFF	4
		Page127	0x0007F000-0x0007FFFF	0x5E07F000-0x5E07FFFF	4
	1	Page128	0x00080000-0x00080FFF	0x5E080000-0x5E080FFF	4
		Page129	0x00081000-0x00081FFF	0x5E081000-0x5E081FFF	4
		Page130	0x00082000-0x00082FFF	0x5E082000-0x5E082FFF	4
		Page131	0x00083000-0x00083FFF	0x5E083000-0x5E083FFF	4
		Page132	0x00084000-0x00084FFF	0x5E084000-0x5E084FFF	4
		Page133	0x00085000-0x00085FFF	0x5E085000-0x5E085FFF	4
		Page134	0x00086000-0x00086FFF	0x5E086000-0x5E086FFF	4
		Page135	0x00087000-0x00087FFF	0x5E087000-0x5E087FFF	4
		⋮	⋮	⋮	⋮
		⋮	⋮	⋮	⋮
		Page188	0x000BC000-0x000BCFFF	0x5E0BC000-0x5E0BCFFF	4
		Page189	0x000BD000-0x000BDFFF	0x5E0BD000-0x5E0BDFFF	4
		Page190	0x000BE000-0x000BEFFF	0x5E0BE000-0x5E0BEFFF	4
		Page191	0x000BF000-0x000BFFFF	0x5E0BF000-0x5E0BFFFF	4



表 2.9 512KBコードフラッシュのページ構成

FLASH I/F	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	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
		⋮	⋮	⋮	⋮
		Page124	0x0007C000-0x0007CFFF	0x5E07C000-0x5E07CFFF	4
		Page125	0x0007D000-0x0007DFFF	0x5E07D000-0x5E07DFFF	4
		Page126	0x0007E000-0x0007EFFF	0x5E07E000-0x5E07EFFF	4
		Page127	0x0007F000-0x0007FFFF	0x5E07F000-0x5E07FFFF	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 2.10 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

FLASH I/F	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	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 2.11 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 (注3)
1536	512	3	32	48	4	384	29.5 $\mu$ s	11.59s	18.1ms	144.2ms	18.1ms	100.1ms
1024	512	2	32	32	4	256		7.73s				82.0ms
768	512	2(注4)	32	24	4	192		5.80s				
512	512	1	32	16	4	128		3.86s				

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。コードフラッシュ、データフラッシュ、プロテクトビット、セキュリティービットの消去時間を含みます。

注4) エリア1のサイズは256KBです。

## 2.3. データフラッシュの構成

### 2.3.1. データフラッシュの構成単位

データ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 32KB 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 4KB 固定です。
- ページ  
消去機能で使します。  
1 ページは 256 バイト固定です。

### 2.3.2. データフラッシュのブロック構成

表 2.12 32KBデータフラッシュのブロック構成

FLASH I/F	エリア	ブロック名称	書き込み/消去/読み出しアドレス	ブロックサイズ (KB)
2	4	Block0	0x30000000-0x30000FFF	4
		Block1	0x30001000-0x30001FFF	4
		Block2	0x30002000-0x30002FFF	4
		Block3	0x30003000-0x30003FFF	4
		Block4	0x30004000-0x30004FFF	4
		Block5	0x30005000-0x30005FFF	4
		Block6	0x30006000-0x30006FFF	4
		Block7	0x30007000-0x30007FFF	4

### 2.3.3. データフラッシュのページ構成

32KB データフラッシュの例を表 2.13 に示します。

表 2.13 32KBデータフラッシュのページ構成

FLASH I/F	エリア	ページ名称	書き込み/消去/読み出しアドレス	ページサイズ (バイト)
2	4	Page0	0x30000000-0x300000FF	256
		Page1	0x30000100-0x300001FF	256
		Page2	0x30000200-0x300002FF	256
		Page3	0x30000300-0x300003FF	256
		Page4	0x30000400-0x300004FF	256
		Page5	0x30000500-0x300005FF	256
		Page6	0x30000600-0x300006FF	256
		Page7	0x30000700-0x300007FF	256
		Page8	0x30000800-0x300008FF	256
		Page9	0x30000900-0x300009FF	256
		Page10	0x30000A00-0x30000AFF	256
		Page11	0x30000B00-0x30000BFF	256
		Page12	0x30000C00-0x30000CFF	256
		Page13	0x30000D00-0x30000DFF	256
		Page14	0x30000E00-0x30000EFF	256
		Page15	0x30000F00-0x30000FFF	256
		:	:	:
		:	:	:
Page124	0x30007C00-0x30007CFF	256		
Page125	0x30007D00-0x30007DFF	256		
Page126	0x30007E00-0x30007EFF	256		
Page127	0x30007F00-0x30007FFF	256		

### 2.3.4. データフラッシュの書き込み、消去時間

書き込みは4バイト単位(1ワード)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「2.3.5 メモリー容量と構成」を参照してください。

### 2.3.5. メモリー容量と構成

表 2.14 データ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)		消去時間(注)		
	サイズ (KB)	個数	サイズ (KB)	個数	サイズ (Bytes)	個数	ワード	エリア	ページ	ブロック	エリア
32	32	1	4	8	256	128	64.7 $\mu$ s	531ms	1.0ms	15.4ms	9.2ms

注) 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。

### 3. 機能説明・動作説明

コードフラッシュおよびデータフラッシュは、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、外部メモリとしてフラッシュメモリ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"> <li>・自動プログラム</li> <li>・自動チップ消去</li> <li>・自動ブロック消去</li> </ul>	<p>&lt;追加&gt;自動エリア消去、自動ページ消去、自動メモリスワップ            &lt;変更&gt;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)            &lt;削除&gt;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, データフラッシュ, 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, セキュリティー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アクセス (コードフラッシュ) 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アクセス (コードフラッシュ)

シーケンス	第 1 バス	第 2 バス	第 3 バス	第 4 バス	第 5 バス	第 6 バス	第 7 バス
	サイクル	サイクル	サイクル	サイクル	サイクル	サイクル	サイクル
コマンド	アドレス	アドレス	アドレス	アドレス	アドレス	アドレス	アドレス
	データ	データ	データ	データ	データ	データ	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	—	—	—

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	アドレス	アドレス	アドレス	アドレス	アドレス	アドレス
	データ	データ	データ	データ	データ	データ	データ
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 ースワップ 消去	0xYYYYX55X	0xYYYYXAAX	0xYYYYX55X	0xYYYYX55X	0xYYYYXAAX	MSA(注)	—
	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(コードフラッシュ)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(コードフラッシュ)

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アクセス (コードフラッシュ) と併せてご使用願います。

第 1 バスサイクルから「通常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(コードフラッシュ)

[通常コマンド]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:12]	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	"000" 固定	エリア 0:00 1:01 2:10	"0" 推奨	コマンド	"0" 推奨

[Read/リセット、ID-Read]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:16]	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	"000" 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	"000" 固定	"00000" 固定	ID アドレス	"0" 推奨

[自動チップ消去]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:12]	Adr [11:4]	Adr [3:0]
チップ消去	チップ消去の第 1~6 バスライトサイクルアドレス設定				
	0x5E	"000" 固定	"0" 推奨	コマンド	"0" 推奨

[自動エリア消去]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:0]
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	"000" 固定	エリア 0:00 1:01 2:10	"0" 推奨

[自動ブロック消去]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:15]	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	"000" 固定	エリア 0:00 1:01 2:10	ブロック アドレス	"0" 推奨

[自動ページ消去]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:12]	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	"000" 固定	エリア 0:00 1:01 2:10	ページ アドレス	"0" 推奨

[自動プログラム]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:4]	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 4~7 バスライトサイクルアドレス設定)				
	0x5E	"000" 固定	エリア 0:00 1:01 2:10	プログラム アドレス	"0" 推奨

[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 [23:21]	Adr [20:19]	Adr [18:12]	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	"000" 固定	FLASH I/F 0:00 1:10	"0000010" 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	"000" 固定	FLASH I/F 0:00 1:10	"0000010" 固定	プロテクト ビット選択	"0" 推奨

[自動メモリスワッププログラム/消去]

アドレス	Adr [31:24]	Adr [23:21]	Adr [20:19]	Adr [18:12]	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	"000" 固定	FLASH I/F 0:00 1:10	"0000011" 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	"000" 固定	FLASH I/F 0:00 1:10	"0000011" 固定	メモリスワップ 選択	"0" 推奨

[自動セキュリティビットプログラム/消去]

アドレス	Adr [31:24]	Adr [23:19]	Adr [18:17]	Adr [16:12]	Adr [11:0]
セキュリティビット消去	SBA: セキュリティビット消去アドレス (セキュリティビット消去の第 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
セキュリティビットプログラム	SBA: セキュリティビットプログラムアドレス (セキュリティ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~表 2.5 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

FLASH I/F	エリア	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 [11:10]	Adr [9]	Adr [8]	Adr [7]	Adr [6]	Adr [5]	Adr [4]		
0	0	0 (注)	0	[FCPSR0]	<PG0>	00	0	0	0	0	0	0	0	0x5E002000
			1		<PG1>	00	0	0	0	0	0	0	1	0x5E002010
			2		<PG2>	00	0	0	0	0	0	1	0	0x5E002020
			3		<PG3>	00	0	0	0	0	0	1	1	0x5E002030
			4		<PG4>	00	0	0	0	0	1	0	0	0x5E002040
			5		<PG5>	00	0	0	0	0	1	0	1	0x5E002050
			6		<PG6>	00	0	0	0	0	1	1	0	0x5E002060
			7		<PG7>	00	0	0	0	0	1	1	1	0x5E002070
		1	8~15	[FCPSR1]	<BLK1>	00	0	0	1	0	0	0	0	0x5E002080
		2	16~23		<BLK2>	00	0	0	1	0	0	1	0x5E002090	
		3	24~31		<BLK3>	00	0	0	1	0	1	0	0x5E0020A0	
		4	32~39		<BLK4>	00	0	0	1	0	1	1	0x5E0020B0	
		5	40~47		<BLK5>	00	0	0	1	1	0	0	0x5E0020C0	
		6	48~55		<BLK6>	00	0	0	1	1	0	1	0x5E0020D0	
		7	56~63		<BLK7>	00	0	0	1	1	1	0	0x5E0020E0	
8	64~71	<BLK8>	00		0	0	1	1	1	1	0x5E0020F0			

1	1	9	72~79		<BLK9>	00	0	1	0	0	0	0	0x5E002100	
		10	80~87		<BLK10>	00	0	1	0	0	0	1	0x5E002110	
		11	88~95		<BLK11>	00	0	1	0	0	1	0	0x5E002120	
		12	96~103		<BLK12>	00	0	1	0	0	1	1	0x5E002130	
		13	104~111		<BLK13>	00	0	1	0	1	0	0	0x5E002140	
		14	112~119		<BLK14>	00	0	1	0	1	0	1	0x5E002150	
		15	120~127		<BLK15>	00	0	1	0	1	1	0	0x5E002160	
		16	128~135		<BLK16>	00	0	1	0	1	1	1	0x5E002170	
		17	136~143		<BLK17>	00	0	1	1	0	0	0	0x5E002180	
		18	144~151		<BLK18>	00	0	1	1	0	0	1	0x5E002190	
		19	152~159		<BLK19>	00	0	1	1	0	1	0	0x5E0021A0	
		20	160~167		<BLK20>	00	0	1	1	0	1	1	0x5E0021B0	
		21	168~175		<BLK21>	00	0	1	1	1	0	0	0x5E0021C0	
		22	176~183		<BLK22>	00	0	1	1	1	0	1	0x5E0021D0	
		23	184~191		<BLK23>	00	0	1	1	1	1	0	0x5E0021E0	
	24	192~199	<BLK24>	00	0	1	1	1	1	1	0x5E0021F0			
	25	200~207	<BLK25>	00	1	0	0	0	0	0	0x5E002200			
	26	208~215	<BLK26>	00	1	0	0	0	0	1	0x5E002210			
	27	216~223	<BLK27>	00	1	0	0	0	1	0	0x5E002220			
	28	224~231	<BLK28>	00	1	0	0	0	1	1	0x5E002230			
	29	232~239	<BLK29>	00	1	0	0	1	0	0	0x5E002240			
	30	240~247	<BLK30>	00	1	0	0	1	0	1	0x5E002250			
	31	248~255	<BLK31>	00	1	0	0	1	1	0	0x5E002260			
	1	2	0 (注)	0	[FCPSR3]	<PG0>	00	0	0	0	0	0	0	0x5E102000
				1		<PG1>	00	0	0	0	0	0	1	0x5E102010
				2		<PG2>	00	0	0	0	0	1	0	0x5E102020
				3		<PG3>	00	0	0	0	0	1	1	0x5E102030
				4		<PG4>	00	0	0	0	1	0	0	0x5E102040
				5		<PG5>	00	0	0	0	1	0	1	0x5E102050
				6		<PG6>	00	0	0	0	1	1	0	0x5E102060
		7	<PG7>	00	0	0	0	1	1	1	0x5E102070			
1		8~15	[FCPSR4]	<BLK1>	00	0	0	1	0	0	0	0x5E102080		
2		16~23		<BLK2>	00	0	0	1	0	0	1	0x5E102090		
3		24~31		<BLK3>	00	0	0	1	0	1	0	0x5E1020A0		
4		32~39		<BLK4>	00	0	0	1	0	1	1	0x5E1020B0		
5		40~47		<BLK5>	00	0	0	1	1	0	0	0x5E1020C0		
6		48~55		<BLK6>	00	0	0	1	1	0	1	0x5E1020D0		
7		56~63		<BLK7>	00	0	0	1	1	1	0	0x5E1020E0		
8		64~71		<BLK8>	00	0	0	1	1	1	1	0x5E1020F0		
9		72~79		<BLK9>	00	0	1	0	0	0	0	0x5E102100		

	10	80~87		<BLK10>	00	0	1	0	0	0	1	0x5E102110
	11	88~95		<BLK11>	00	0	1	0	0	1	0	0x5E102120
	12	96~103		<BLK12>	00	0	1	0	0	1	1	0x5E102130
	13	104~111		<BLK13>	00	0	1	0	1	0	0	0x5E102140
	14	112~119		<BLK14>	00	0	1	0	1	0	1	0x5E102150
	15	120~127		<BLK15>	00	0	1	0	1	1	0	0x5E102160

(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(コードフラッシュ)

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	0x022F	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[FCWPSR]<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[FCWPSR]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[FCWPSR]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]	
[FCWPSR]	<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. データフラッシュ

### 3.2.1. データフラッシュのコマンドシーケンス

#### 3.2.1.1. データフラッシュのコマンドシーケンス一覧

データ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 3.7 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 3.8 バスライトサイクル時のアドレスビット構成(データフラッシュ)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(data) に設定してください

表 3.7 コマンドシーケンス(データフラッシュ)

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	アドレス	アドレス	アドレス	アドレス	アドレス	アドレス
	データ	データ	データ	データ	データ	データ	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	—	—	—
	0xAA	0x55	0xC0	PD0	—	—	—
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	—
	0xAA	0x55	0x80	0xAA	0x55	0x60	—

## 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)  
 PBA: プロテクトビットアドレス

### 3.2.1.2. バスライトサイクル時のアドレスビット構成(データフラッシュ)

表 3.8 バスライトサイクル時のアドレスビット構成(データフラッシュ)は、表 3.7 コマンドシーケンスと併せてご使用願います。

第1バスサイクルから「通常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 3.8 バスライトサイクル時のアドレスビット構成(データフラッシュ)

[通常のコマンド]

アドレス	Adr [31:24]	Adr [23:16]	Adr [15]	Adr [14:12]	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					
	0x30	"00000000" 固定	エリア (注)	"0" 推奨	コマンド	"0" 推奨

注) エリアは"0"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 [23:16]	Adr [15]	Adr [14:13]	Adr [12:0]
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				
	0x30	"00000000" 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)				
	0x30	"00000000" 固定	"0" 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 [23:16]	Adr [15]	Adr [14:0]
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)			
	0x30	"00000000" 固定	エリア (注)	"0" 推奨

注) エリアは"0"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 [23:16]	Adr [15]	Adr [14:12]	Adr [11:0]
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)				
	0x30	"00000000" 固定	エリア (注)	ブロック アドレス	"0" 推奨

注) エリアは"0"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 [23:16]	Adr [15]	Adr [14:8]	Adr [7:0]
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)				
	0x30	"00000000" 固定	エリア (注)	ページ アドレス	"0" 推奨

注) エリアは"0"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 [23:16]	Adr [15]	Adr [14:2]	Adr [1:0]
プログラム	PA:プログラムアドレス(プログラムの第 4 バスライトサイクルアドレス設定)				
	0x30	"00000000" 固定	エリア (注)	プログラム アドレス	"0" 推奨

注) エリアは"0"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 [23:16]	Adr [15]	Adr [14:8]	Adr [7:2]	Adr [1:0]
プロテクトビット 消去	PBA: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					
	0x30	"00000000" 固定	"0"固定	"0000001" 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)					
	0x30	"00000000" 固定	"0"固定	"0000001" 固定	プロテクト ビット選択	"0" 推奨

### 3.2.1.3. エリアアドレス(AA)、ブロックアドレス(BA)

“表 2.12 32KB データ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。



### 3.2.1.4. プロテクトビットの指定(PBA)

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 3.9 プロテクトビットプログラムアドレス表(データフラッシュ)」に示します。

表 3.9 プロテクトビットプログラムアドレス表(データフラッシュ)

エリア	ブロック	レジスター	プロテクトビット	PBA[7:2]					アドレス例 [31:0]
				Adr [7:6]	Adr [5]	Adr [4]	Adr [3]	Adr [2]	
4	0	[FCPSR6]	<DBLK0>	00	0	0	0	0	0x30000100
	1		<DBLK1>	00	0	0	0	1	0x30000104
	2		<DBLK2>	00	0	0	1	0	0x30000108
	3		<DBLK3>	00	0	0	1	1	0x3000010C
	4		<DBLK4>	00	0	1	0	0	0x30000110
	5		<DBLK5>	00	0	1	0	1	0x30000114
	6		<DBLK6>	00	0	1	1	0	0x30000118
	7		<DBLK7>	00	0	1	1	1	0x3000011C

### 3.2.1.5. ID-Read のコード(IA, ID) : データフラッシュ

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 3.10 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 3.10 ID-Readコマンドのコード指定とコードの内容 (データフラッシュ)

Code	ID[15:0]	IA[14:13]	アドレス例[31:0]
メーカーコード	0x0098	00	0x30000000
デバイスコード	0x005A	01	0x30002000
—	Reserved	10	N/A
マクロコード	0x0240	11	0x30006000

## 3.3. 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。

### 3.3.1. 自動プログラム

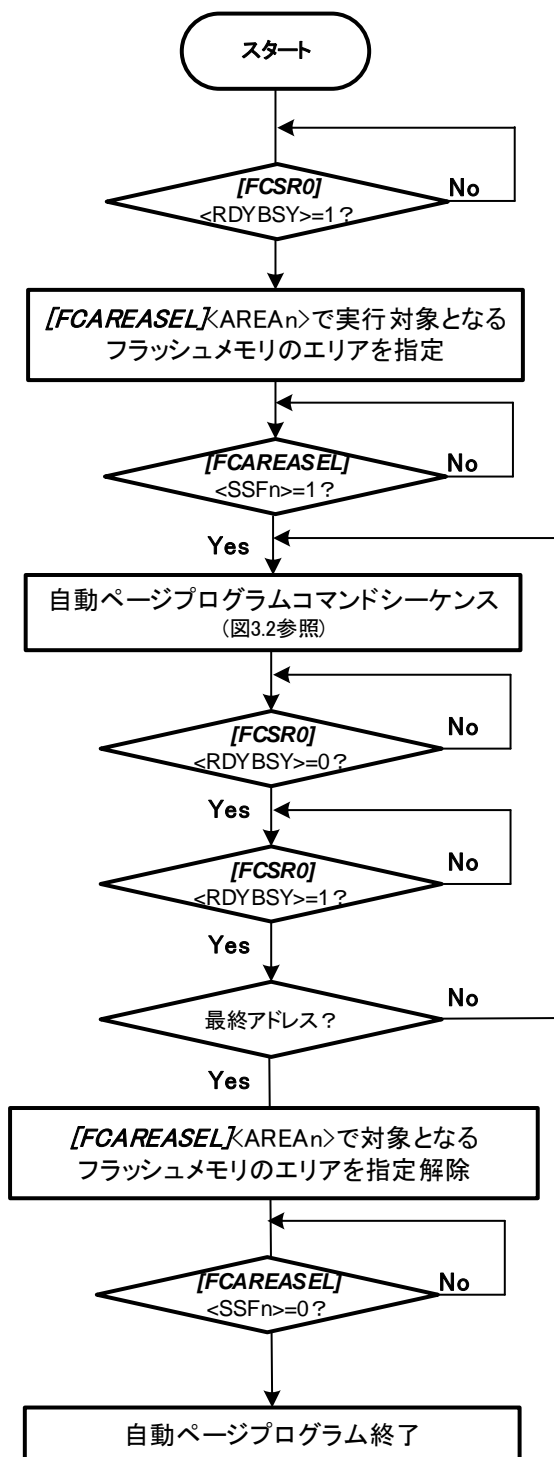


図 3.1 自動プログラムのフロー(1)

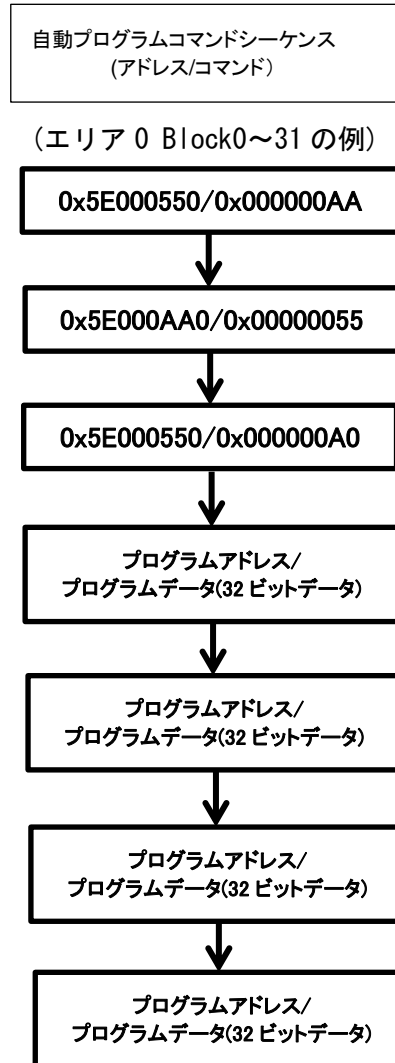


図 3.2 自動プログラムのフロー(2)

## 3.3.2. 自動消去

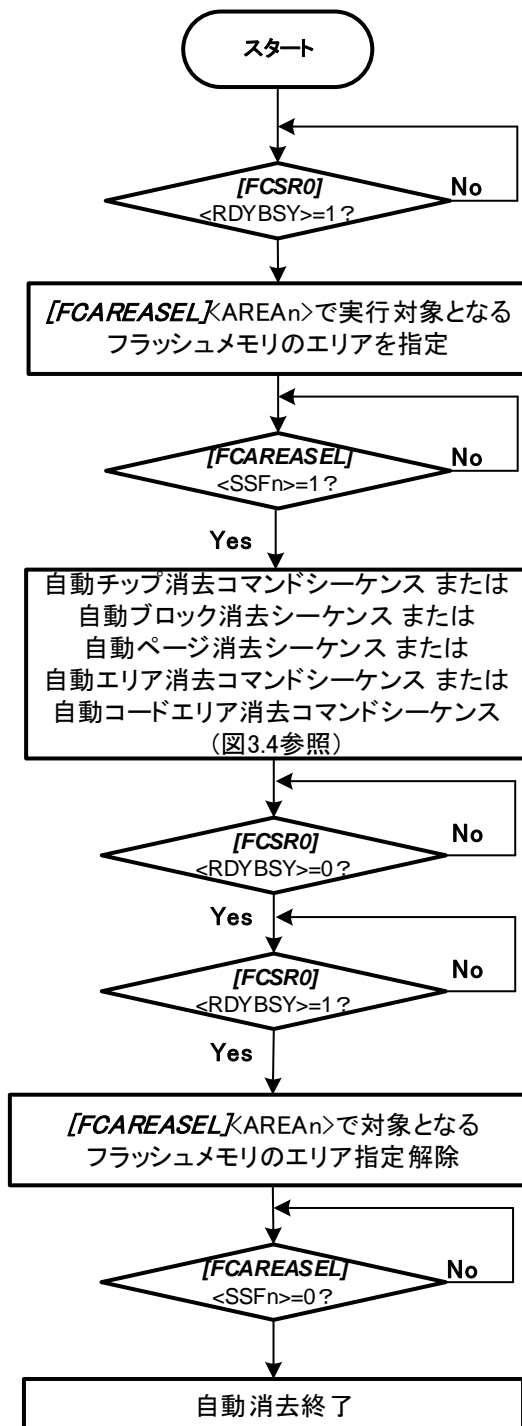


図 3.3 自動消去のフロー(1)

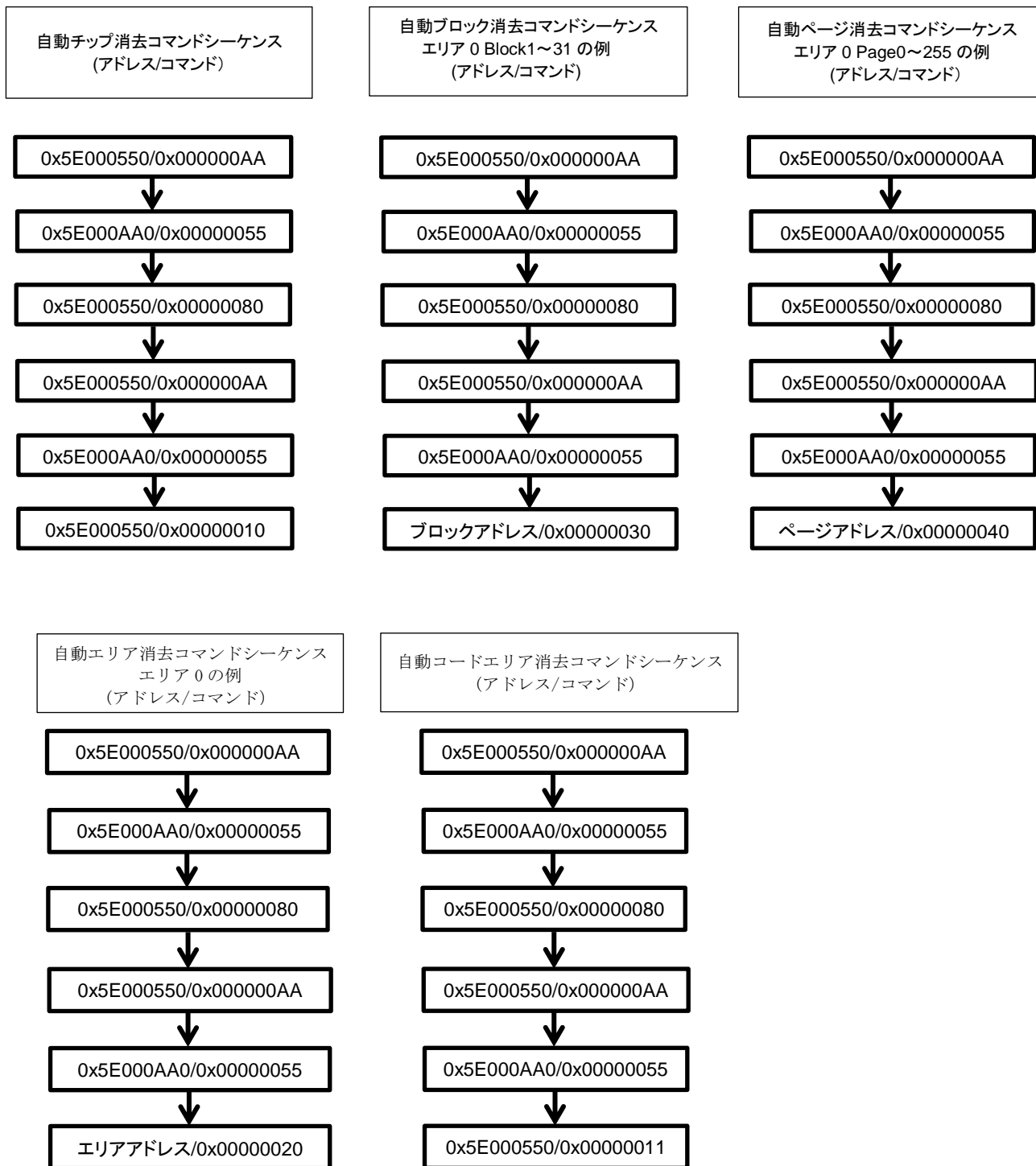


図 3.4 自動消去のフロー(2)

## 3.3.3. プロテクトビット

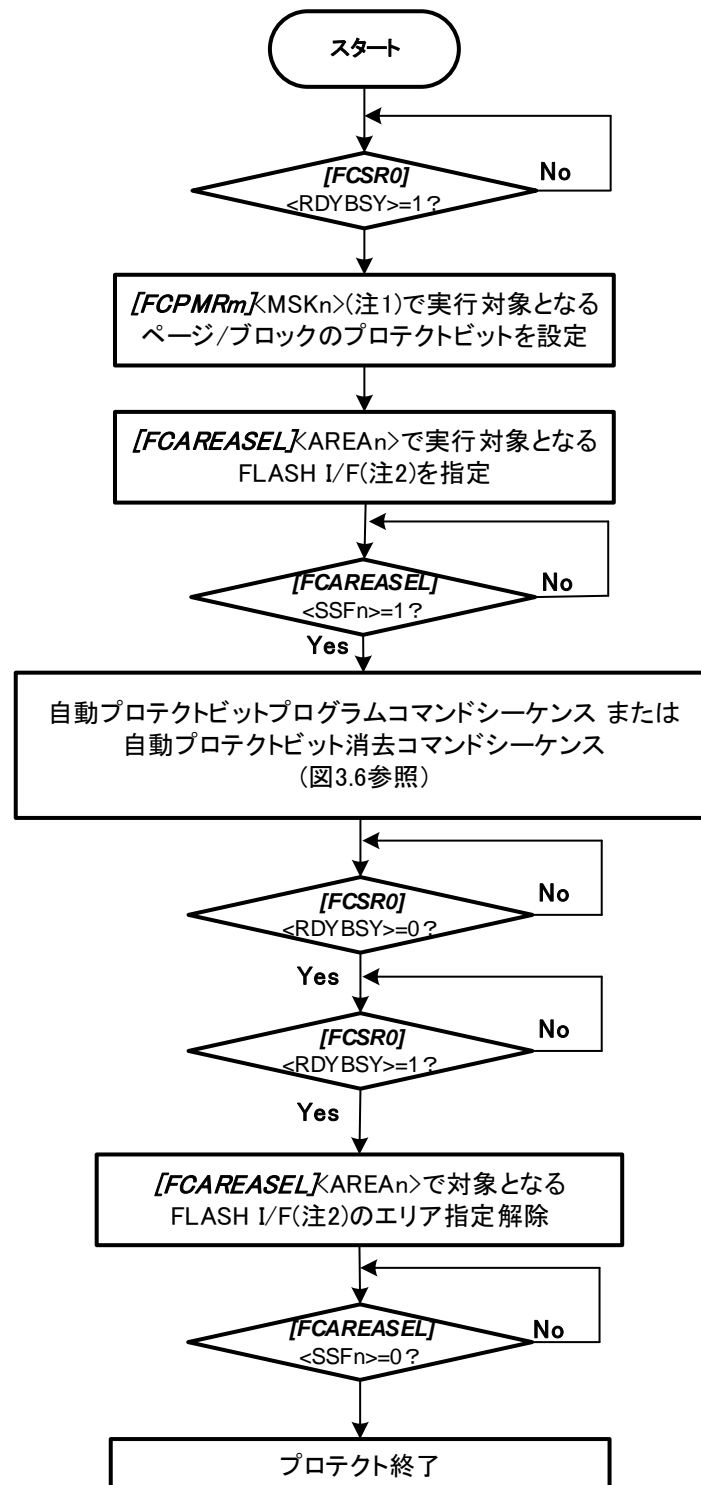


図 3.5 プロテクトのフロー(1)

注1) <PMn>、<MSKn>、<DMSKn>は代表して<MSKn>で表します。

注2) FLASH I/F0 の場合はエリア 0、FLASH I/F1 の場合はエリア 2、FLASH I/F2 の場合はエリア 4 を指定します。

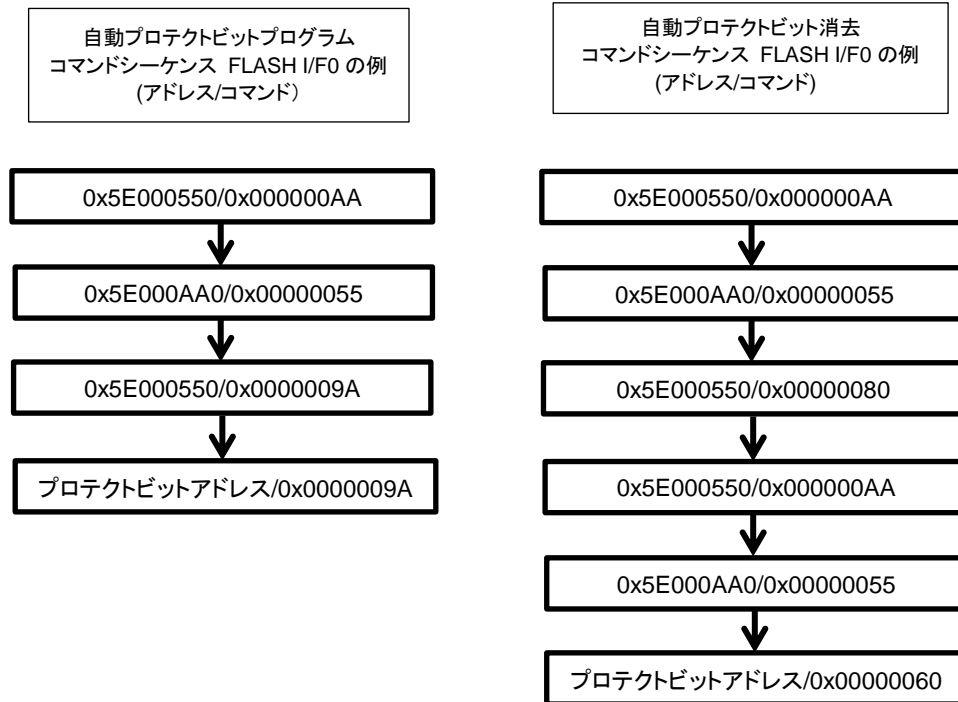


図 3.6 プロテクトのフロー(2)

## 3.3.4. セキュリティビット

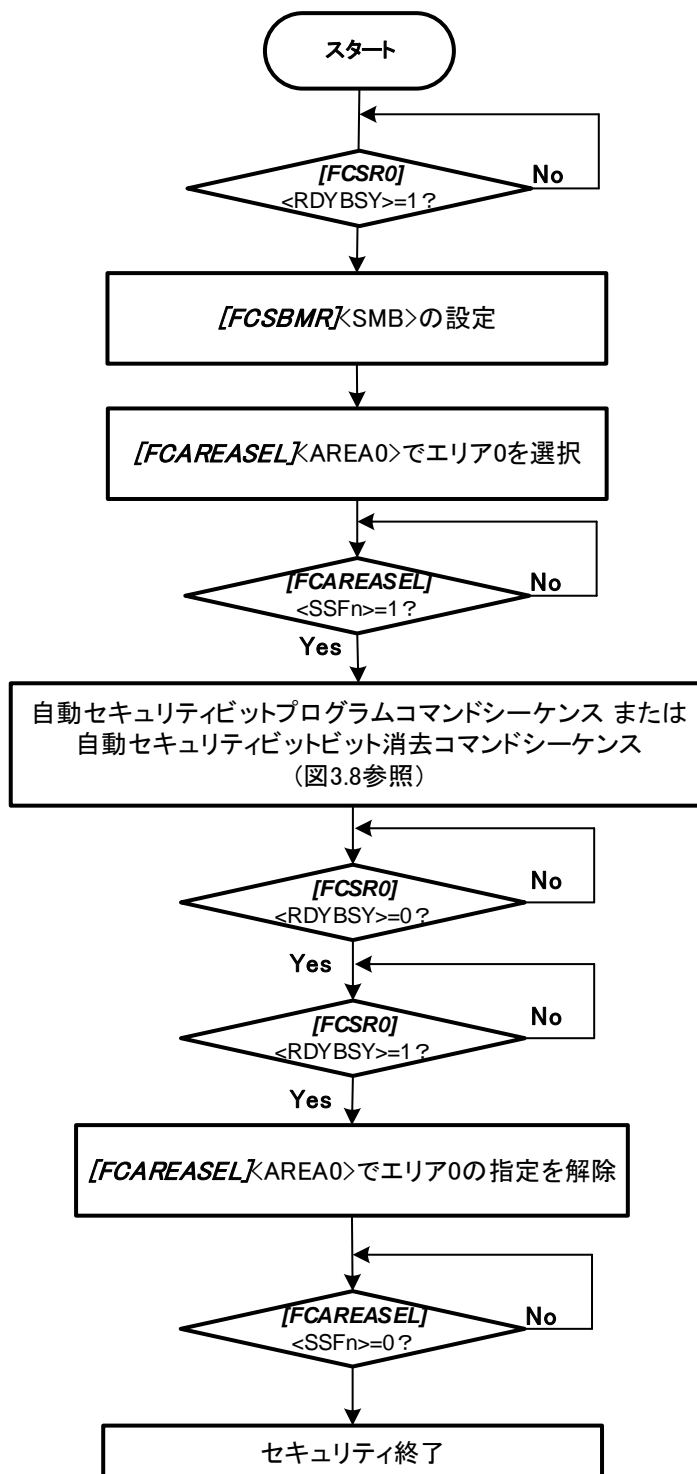


図 3.7 セキュリティのフロー(1)



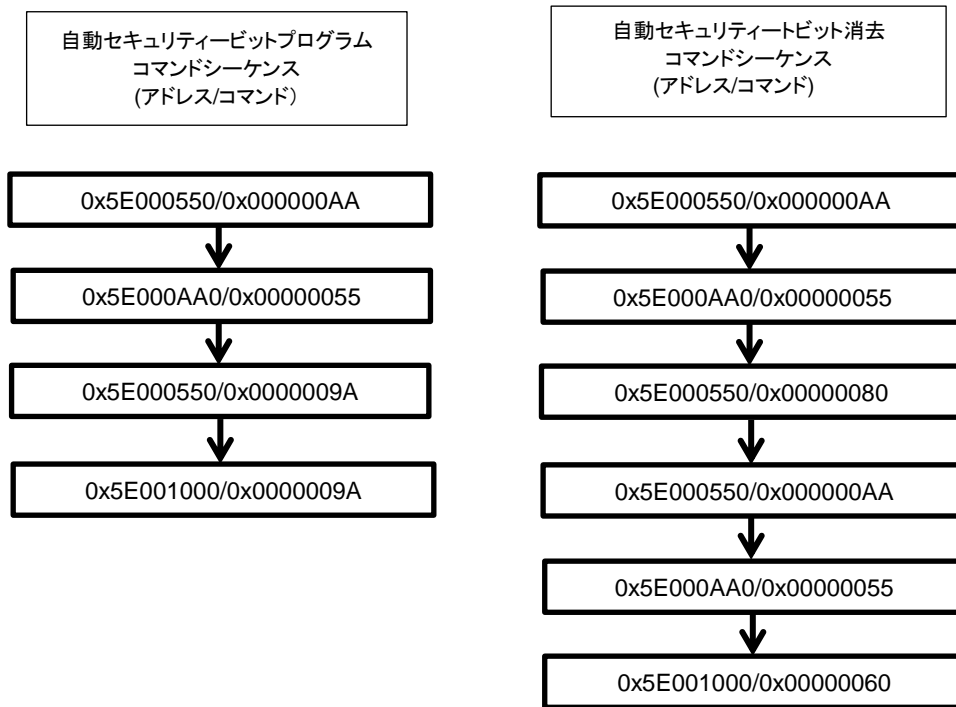


図 3.8 セキュリティーのフロー(2)

## 3.3.5. メモリスワップ

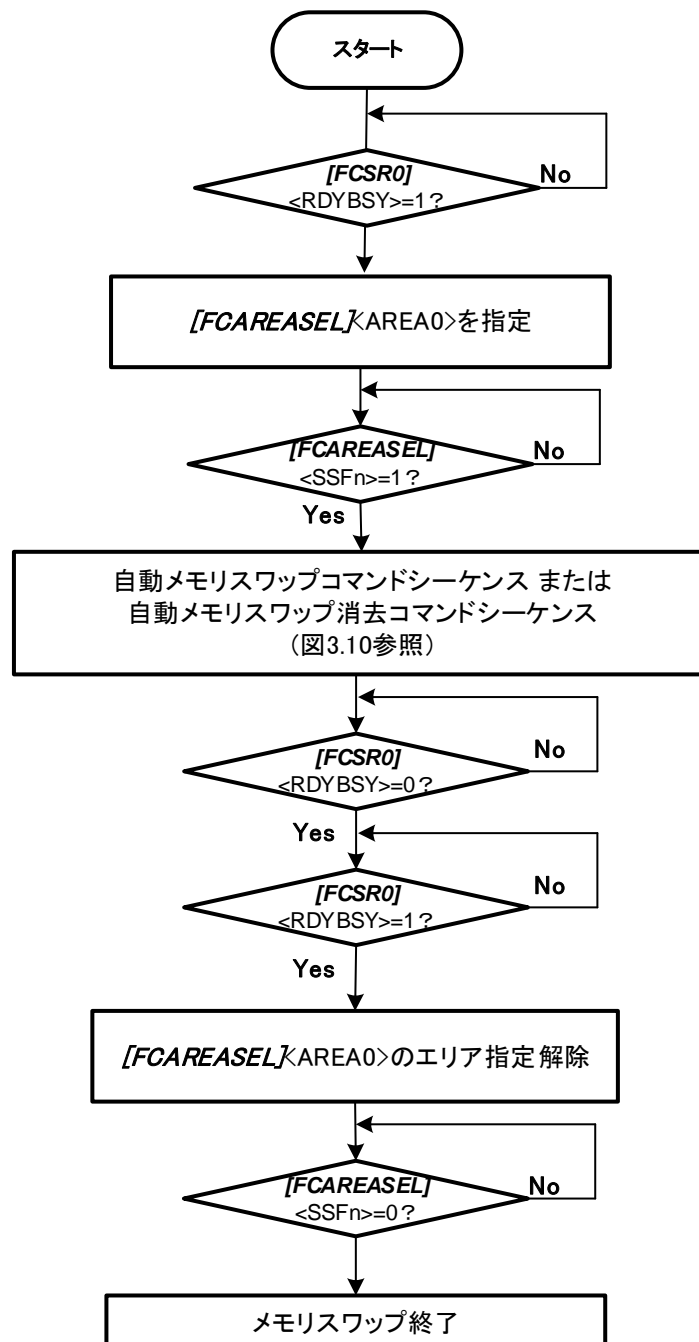


図 3.9 メモリスワップのフロー(1)

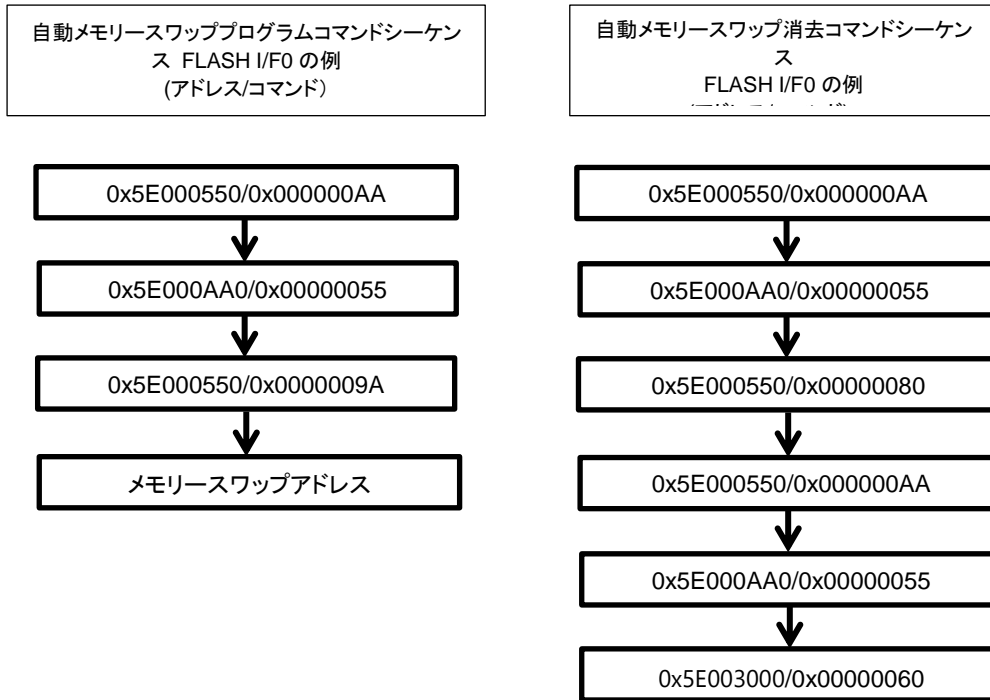


図 3.10 メモリスワップのフロー(2)

## 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。

また FLASH I/F0 のフラッシュメモリー上でプログラムを実行中に、命令実行を行っていない他の FLASH I/F(例えば FLASH I/F2<エリア 4>: データフラッシュ)のフラッシュメモリーを書き込み/消去できます(逆も可能です)。このような使い方を本ドキュメント内ではデュアルモードと呼びます。

### 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。 データフラッシュ:1ワード(4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。
自動セキュリティビットプログラム/消去	フラッシュメモリーへのセキュリティ設定および解除することができます。
自動メモリスワッププログラム/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 **ID-Read** コマンドを除き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際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
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。

FLASH I/F が 2 つ以上ある場合、上記以外のコマンドシーケンスは、デュアルモードによる書き替えが可能です。例えば FLASH I/F0 と FLASH I/F2 がある場合、書き込み/消去を行う対象(FLASH I/F2)のフラッシュメモリ以外のエリア(FLASH I/F0)のフラッシュメモリ上でプログラムを実行できます(逆も可能です)。

デュアルモードの場合は、FLASH I/F2(データフラッシュ)へのプログラムの書き替えに限り割り込みの使用が可能です。

### 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」および「3.2.1 データ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。データフラッシュは1ワード(4 バイト)単位で書き込みができます。

フラッシュメモリへの書き込みは、"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ありません。  
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以降は4ワードの内の残りのデータを書き込みます。データフラッシュは1ワード(32ビット)単位で書き込んでください。

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。

データフラッシュの4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4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、コードフラッシュとデータ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。データフラッシュ、コードフラッシュの順番で消去します。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、データフラッシュ

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、フラッシュメモリ内部ではページ単位で消去動作を繰り返します。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、フラッシュメモリ内部ではページ単位で消去動作を繰り返します。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「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  
コードフラッシュとデータフラッシュ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、セキュリティービットは解除されます。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、コードフラッシュとデータ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、セキュリティービットは解除されます。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プログラム

##### (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。正常に書き込みできたかどうか、**[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

項目	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	INTFLCRDY0	FLASH I/F0 の Code FLASH Ready 割り込み
	INTFLCRDY1	FLASH I/F1 の Code FLASH Ready 割り込み
データフラッシュの書き込み/消去動作の完了	INTFLDRDY	Data FLASH Ready 割り込み

自動チップ消去コマンドシーケンスを実行した場合は、まずデータフラッシュへの書き込み/消去終了時にINTFLDRDYが発生、次にコードフラッシュ(FLASH I/F1)への書き込み/消去終了時にINTFLCRDY1が、最後にコードフラッシュ(FLASH I/F0)への書き込み/終了後にINTFLCRDY0が発生します。

##### 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(データフラッシュの場合)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. INTFLDRDY 割り込みの許可の設定をします。
2. データフラッシュに対し書き込み/消去コマンドを発行後、**[FCSR0]<RDYBSY>**で自動動作中(BUSY 状態)を確認します。
3. フラッシュの自動動作終了後、INTFLDRDY 割り込みが発生します。

4. 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、INTFLDRDY 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、INTFLDRDY 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。
5. 書き込みを継続する場合は、メイン処理を行いながら 3~4 を繰り返します。

#### 4.1.6. プロテクト機能

プロテクト機能は、フラッシュメモリへの書き込み、消去をブロック単位で禁止することができます。プロテクトの設定は、コードフラッシュとデータフラッシュで個別に設定します。

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。データフラッシュでは、全て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

プロテクト機能を有効にするには、プロテクトビットプログラムコマンドでプロテクトビットを“1”にします。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)  $\langle PMn \rangle$ 、 $\langle MSKn \rangle$ 、 $\langle DMSKn \rangle$  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 レジスター詳細」の  $[FCPMRm]$  を参照してください。

注)  $\langle PMn \rangle$ 、 $\langle MSKn \rangle$ 、 $\langle DMSKn \rangle$  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.  $[FCSBMR]\langle SMB \rangle = 1$
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、 $[FCSBMR]\langle SMB \rangle = 1$  にして  $[FCSSR]\langle SEC \rangle$  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.  $[FCSBMR]\langle SMB \rangle = 0$  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。

$[FCSBMR]\langle SMB \rangle = 1$ 、 $[FCSSR]\langle SEC \rangle = 1$  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、データ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。

##### 4.1.7.3. 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「6.8 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	Page0 旧オリジナル	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	新オリジナル	旧オリジナルのコピー

消去ルーチン: フラッシュメモリ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 2.10 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—説明

### 5.1. レジスタ—一覧

フラッシュメモリ—関連のレジスタ—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 Type 1
フラッシュメモリ—	FC	-	0x5DFF0000

レジスタ—名		アドレス(Base+)
Flash Security Bit Mask Register	<b>[FCSBMR]</b>	0x0010
Flash Security Status Register	<b>[FCSSR]</b>	0x0014
Flash Key Code Register	<b>[FCKCR]</b>	0x0018
Flash Status Register 0	<b>[FCSR0]</b>	0x0020
Flash Protect Status Register 0	<b>[FCPSR0]</b>	0x0030
Flash Protect Status Register 1	<b>[FCPSR1]</b>	0x0034
Flash Protect Status Register 3	<b>[FCPSR3]</b>	0x003C
Flash Protect Status Register 4	<b>[FCPSR4]</b>	0x0040
Flash Protect Status Register 6	<b>[FCPSR6]</b>	0x0048
Flash Protect Mask Register 0	<b>[FCPMR0]</b>	0x0050
Flash Protect Mask Register 1	<b>[FCPMR1]</b>	0x0054
Flash Protect Mask Register 3	<b>[FCPMR3]</b>	0x005C
Flash Protect Mask Register 4	<b>[FCPMR4]</b>	0x0060
Flash Protect Mask Register 6	<b>[FCPMR6]</b>	0x0068
Flash Status Register 1	<b>[FCSR1]</b>	0x0100
Flash Memory SWAP Status Register	<b>[FCSWPSR]</b>	0x0104
Flash Area Selection Register	<b>[FCAREASEL]</b>	0x0140
Flash Control Register	<b>[FCCR]</b>	0x0148
Flash Status Clear Register	<b>[FCSTSCLR]</b>	0x014C
Flash Bank Change Register	<b>[FCBNKCR]</b>	0x0150
Flash Buffer Disable and Clear Register	<b>[FCBUFDISCLR]</b>	0x0158

注)レジスタ—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(電源投入時またはSTOP2(電源遮断ありから)の復帰時)。

注) 本レジスタ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[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	0x00000000	W	レジスタ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	RDYBSY2	1	R	エリア 4 の ReadyBusy 0:自動動作中 1:自動動作終了
9	RDYBSY1	1	R	エリア 2 の ReadyBusy 0:自動動作中 1:自動動作終了
8	RDYBSY0	1	R	エリア 0 と 1 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	BLK31	0/1	R	<p>コードフラッシュ(エリア 0,1)のプロテクトステータス</p> <p>1:プロテクト状態 0:プロテクト状態ではない</p> <p>Block1 から Block31 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。</p>
30	BLK30	0/1	R	
29	BLK29	0/1	R	
28	BLK28	0/1	R	
27	BLK27	0/1	R	
26	BLK26	0/1	R	
25	BLK25	0/1	R	
24	BLK24	0/1	R	
23	BLK23	0/1	R	
22	BLK22	0/1	R	
21	BLK21	0/1	R	
20	BLK20	0/1	R	
19	BLK19	0/1	R	
18	BLK18	0/1	R	
17	BLK17	0/1	R	
16	BLK16	0/1	R	
15	BLK15	0/1	R	
14	BLK14	0/1	R	
13	BLK13	0/1	R	
12	BLK12	0/1	R	
11	BLK11	0/1	R	
10	BLK10	0/1	R	
9	BLK9	0/1	R	
8	BLK8	0/1	R	
7	BLK7	0/1	R	
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. [FCPSR3] (Flash Protect Status Register 3)

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 2)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	

## 5.2.8. [FCPSR4] (Flash Protect Status Register 4)

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	BLK15	0/1	R	コードフラッシュ(エリア 2)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 Block15 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。
14	BLK14	0/1	R	
13	BLK13	0/1	R	
12	BLK12	0/1	R	
11	BLK11	0/1	R	
10	BLK10	0/1	R	
9	BLK9	0/1	R	
8	BLK8	0/1	R	
7	BLK7	0/1	R	
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	リードすると"0"が読めます。



## 5.2.9. [FCPSR6] (Flash Protect Status Register 6)

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	DBLK7	0/1	R	データフラッシュ(エリア 4)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  データフラッシュ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はできません。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	DBLK6	0/1	R	
5	DBLK5	0/1	R	
4	DBLK4	0/1	R	
3	DBLK3	0/1	R	
2	DBLK2	0/1	R	
1	DBLK1	0/1	R	
0	DBLK0	0/1	R	

## 5.2.10. [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[FCPMR0]<PMn>のデータを書き換える。

## 5.2.11. [FCPMR1] (Flash Protect Mask Register 1)

Bit	Bit Symbol	リセット後	Type	機能
31	MSK31	1	R/W	コードフラッシュ(エリア 0,1)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 Block31 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。
30	MSK30	1	R/W	
29	MSK29	1	R/W	
28	MSK28	1	R/W	
27	MSK27	1	R/W	
26	MSK26	1	R/W	
25	MSK25	1	R/W	
24	MSK24	1	R/W	
23	MSK23	1	R/W	
22	MSK22	1	R/W	
21	MSK21	1	R/W	
20	MSK20	1	R/W	
19	MSK19	1	R/W	
18	MSK18	1	R/W	
17	MSK17	1	R/W	
16	MSK16	1	R/W	
15	MSK15	1	R/W	
14	MSK14	1	R/W	
13	MSK13	1	R/W	
12	MSK12	1	R/W	
11	MSK11	1	R/W	
10	MSK10	1	R/W	
9	MSK9	1	R/W	
8	MSK8	1	R/W	
7	MSK7	1	R/W	
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。

## 5.2.12. [FCPMR3] (Flash Protect Mask Register 3)

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 2)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。

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[FCPMR3]<PMn>のデータを書き換える。

## 5.2.13. [FCPMR4] (Flash Protect Mask Register 4)

Bit	Bit Symbol	リセット後	Type	機能
31:16	-	1	R/W	"1"を書いてください
15	MSK15	1	R/W	コードフラッシュ(エリア 2)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 Block15 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。
14	MSK14	1	R/W	
13	MSK13	1	R/W	
12	MSK12	1	R/W	
11	MSK11	1	R/W	
10	MSK10	1	R/W	
9	MSK9	1	R/W	
8	MSK8	1	R/W	
7	MSK7	1	R/W	
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。

## 5.2.14. [FCPMR6] (Flash Protect Mask Register 6)

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。
15:8	-	1	R/W	"1"を書いてください。
7	DMSK7	1	R/W	データフラッシュ(エリア 4)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) データフラッシュメモリー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	DMSK6	1	R/W	
5	DMSK5	1	R/W	
4	DMSK4	1	R/W	
3	DMSK3	1	R/W	
2	DMSK2	1	R/W	
1	DMSK1	1	R/W	
0	DMSK0	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。

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[FCPMR6]<DMSKn>のデータを書き換える。

## 5.2.15. [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。

## 5.2.16. [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) 010000: エリア(エリア 0 ↔ エリア 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。

## 5.2.17. [FCAREASEL] (Flash Area Selection Register)

Bit	Bit Symbol	リセット後	Type	機能
31	-	0	R	リードすると"0"が読めます。
30	SSF4	0	R	エリア 4 の選択状態 1: エリア 4 を選択 (書き込みモード) 0: エリア 4 を非選択 (リードモード)
29	-	0	R	リードすると"0"が読めます。
28	SSF2	0	R	エリア 2 の選択状態 1: エリア 2 を選択 (書き込みモード) 0: エリア 2 を非選択 (リードモード)
27	SSF1	0	R	エリア 1 の選択状態 1: エリア 1 を選択 (書き込みモード) 0: エリア 1 を非選択 (リードモード)
26	SSF0	0	R	エリア 0 の選択状態 1: エリア 0 を選択 (書き込みモード) 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	AREA4[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) となるデータフラッシュのエリア 4 を指定します。 111: エリア 4 を選択 上記以外: エリア 4 を非選択
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	AREA2[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 2 を指定します。 111: エリア 2 を選択 上記以外: エリア 2 を非選択
7	-	0	R	リードすると"0"が読めます。
6:4	AREA1[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 1 を指定します。 111: エリア 1 を選択 上記以外: エリア 1 を非選択
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。  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]>, <AREA1[2:0]>, <AREA2[2:0]>, <AREA4[2:0]>を書き換えた場合は、<SSF0>, <SSF1>, <SSF2>, <SSF4>をリードして書き換えた結果が反映されるまで待つてから、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。

## 5.2.18. **[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RAM上のプログラムで行ってください。

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**[FCCR]**<WEABORT>のデータを書き換える。

## 5.2.19. **[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	<b>[FCSR1]</b> <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。



## 5.2.20. [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。"5.2.21[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。

## 5.2.21. [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	コードフラッシュのバッファを停止してクリアします。 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  [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する場合には、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。

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

コードフラッシュまたはデータ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、シングルチップモードにはノーマルモード、デュアルモードがあります。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	ノーマルモード	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。FLASH I/F を 1 つだけ内蔵している場合はこのモードだけ使用可能です。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。
	デュアルモード	ユーザーのアプリケーションプログラムを実行しながら、FLASH I/F の異なる内蔵フラッシュメモリーを書き換えることができます。2 つ以上の FLASH I/F(例えばコードフラッシュのエリア、またはデータフラッシュ)を内蔵している場合に使用可能です。フラッシュメモリーの書き替え方法は「6.7 デュアルモードによる書き替え方法」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 端子またはパワーオンリセット(POR)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(注)	0→1	0
	-	0

注) パワーオンリセット(POR)からのリセット解除によるモードの決定は、一部製品のみ対応しています。詳細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。

### 6.4. モードごとのメモリーマップ

図 1.1 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) コピールーチン: 下記(c)を内蔵 RAM にコピーするためのプログラム
- (c) 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

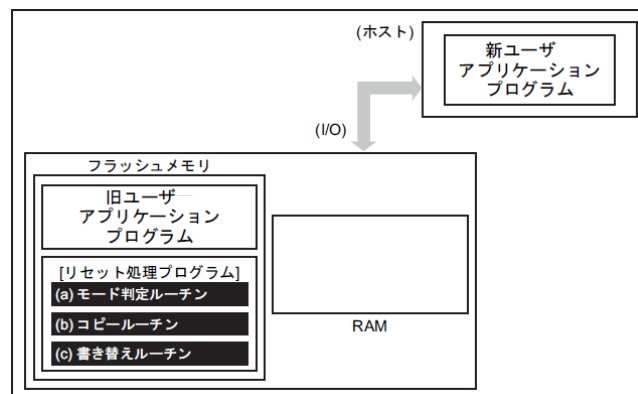


図 6.1 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(1)

## 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)

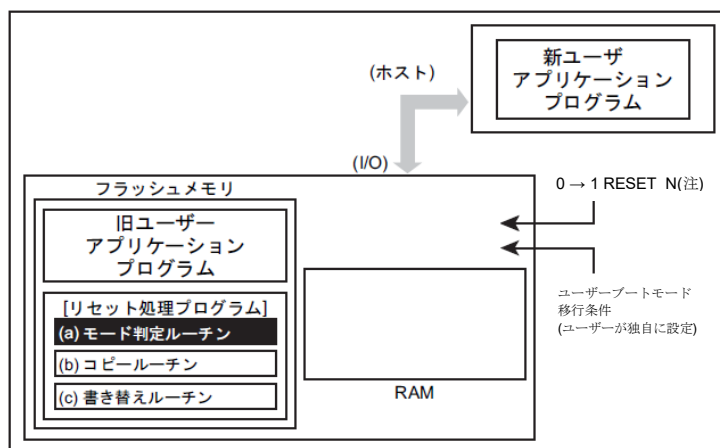


図 6.2 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(2)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

## 6.5.1.3. Step-3

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

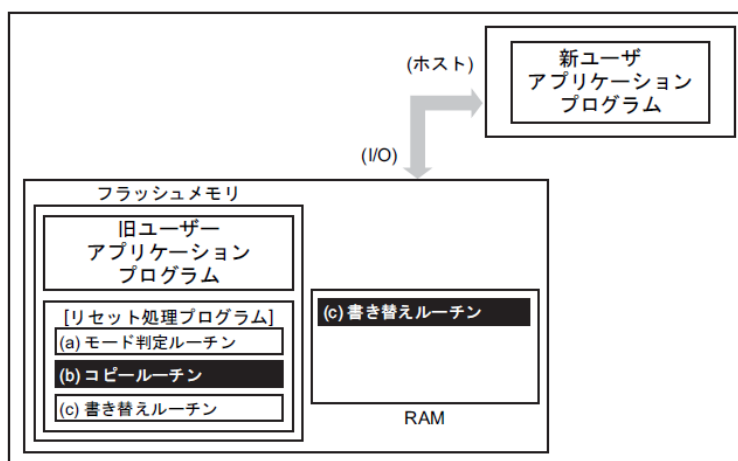


図 6.3 書き替えルーチンをフラッシュメモリーに内蔵場合の手順(3)

## 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。

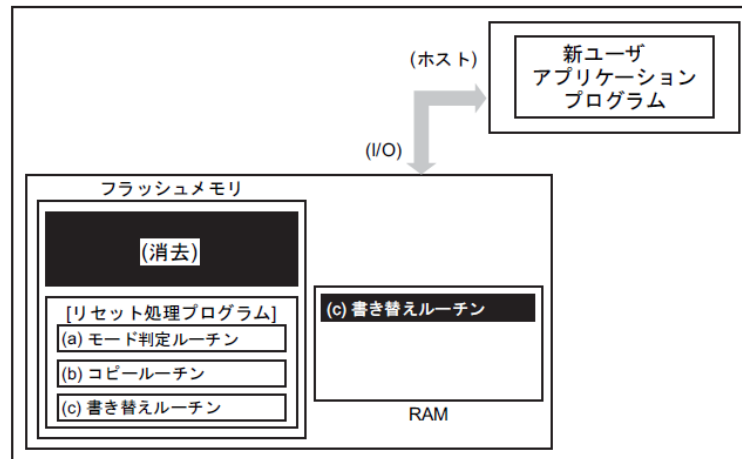


図 6.4 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(4)

## 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。

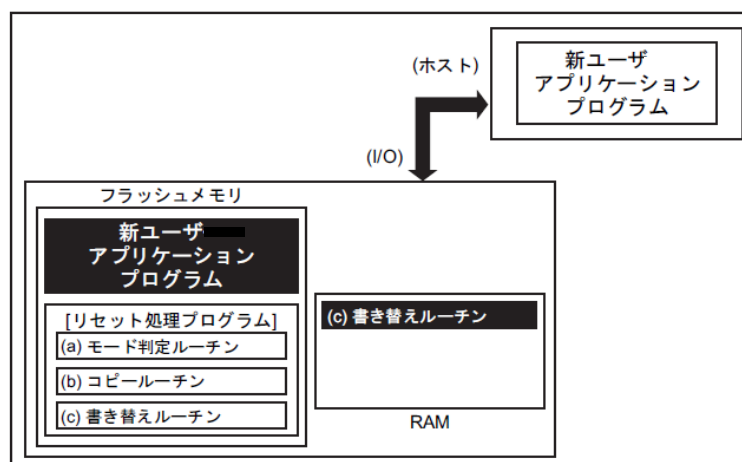


図 6.5 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(5)

## 6.5.1.6. Step-6

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。

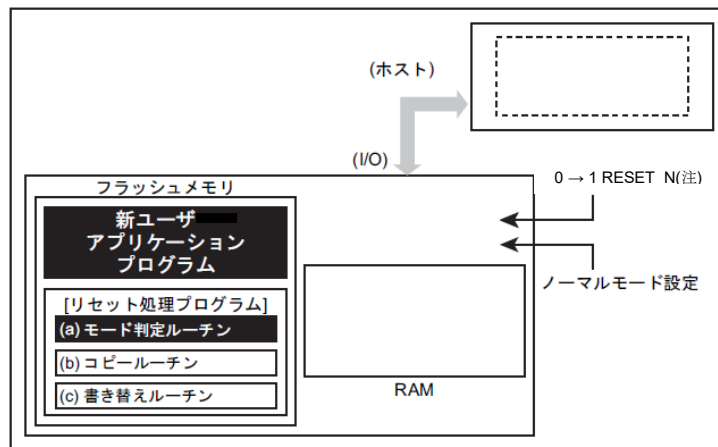


図 6.6 書き替えルーチンをフラッシュメモリーに内蔵する場合の手順(6)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

## 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

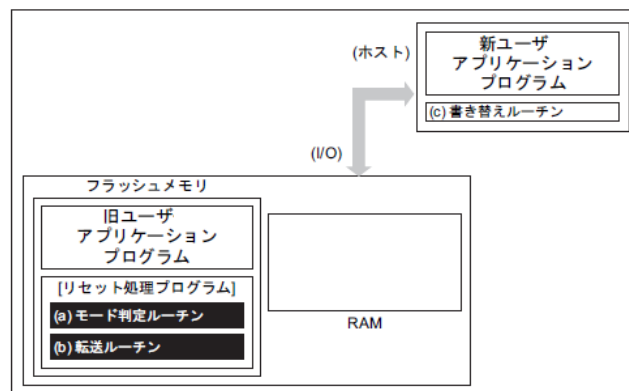


図 6.7 書き替えルーチンを外部から転送する手順(1)

## 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)

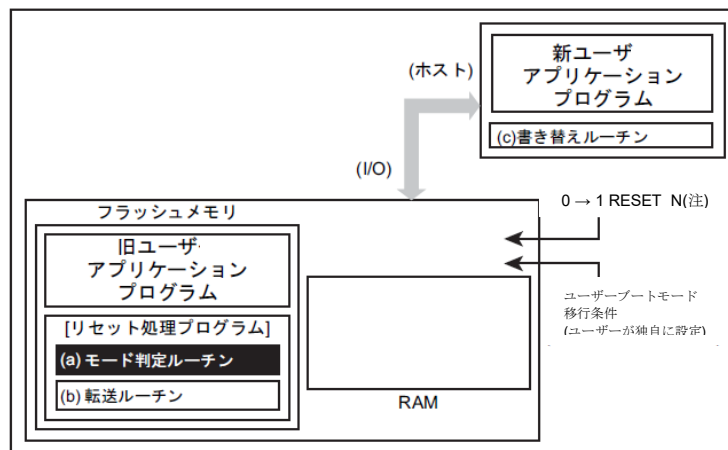


図 6.8 書き替えルーチンを外部から転送する手順(2)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、リファレンスマニュアル「製品固別情報」を参照してください。

## 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。

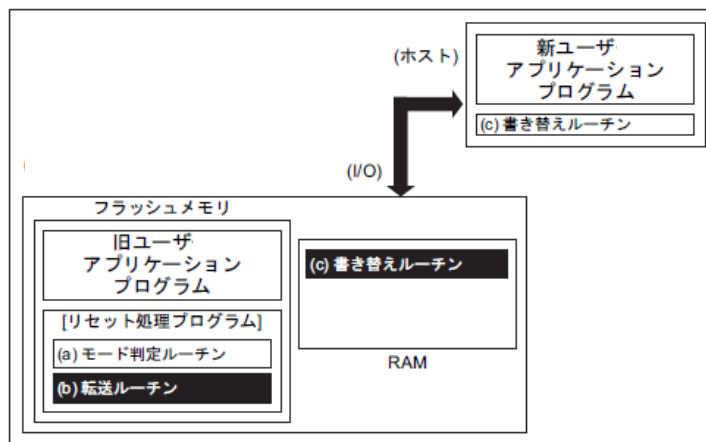


図 6.9 書き替えルーチンを外部から転送する手順(3)



## 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。

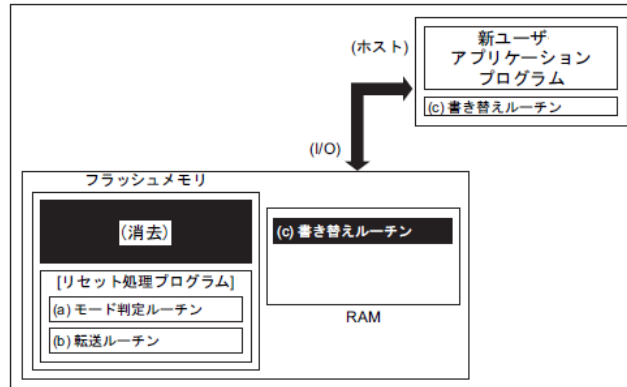


図 6.10 書き替えルーチンを外部から転送する手順(4)

## 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。

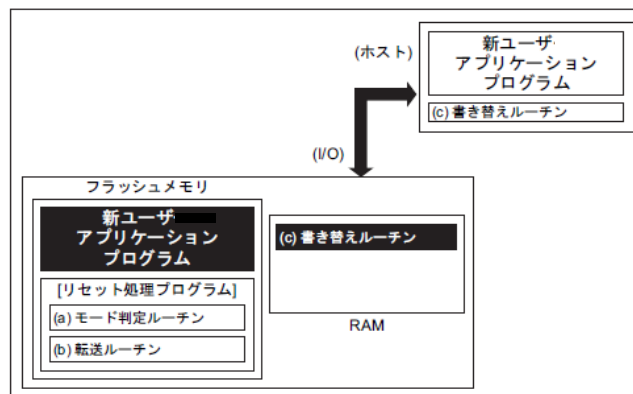


図 6.11 書き替えルーチンを外部から転送する手順(5)

## 6.5.2.6. Step-6

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。

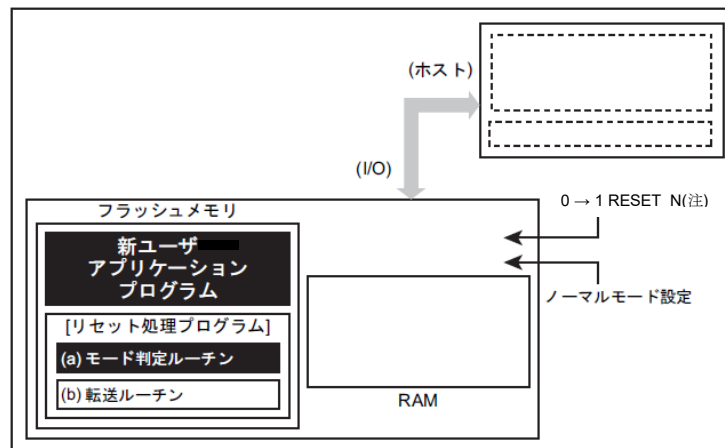


図 6.12 書き替えルーチンを外部から転送する手順(6)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

## 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。
フラッシュメモリ消去コマンド	フラッシュメモリ消去	フラッシュメモリ消去コマンドは、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を消去します。	対象: データフラッシュ、 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。

通信チャンネル: UART チャンネル 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。

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

注 1)使用される UART のチャンネルは製品によって異なります。詳細はリファレンスマニュアルの「製品個別情報」を確認してください。

注 2)同一チャンネルの UART が 2 系統ある場合は、シングルブートモード起動時にどちらにホストが接続されているのか自動で検出して接続されます(製品によって異なります)。使用しないチャンネルの RXD 端子はオープンまたは High 固定にしてください。また、同時に両方の端子にホストを接続しないでください。

#### 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	コードフラッシュの“0x5E001000”～最大容量-N×3-1 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) データフラッシュはパスワード領域としては使えません。

#### 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. フラッシュメモリ消去

フラッシュメモリ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、パスワード無しでデータフラッシュ、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

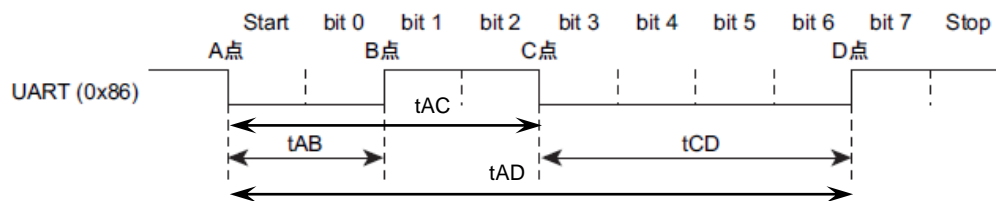


図 6.13 シリアル動作モード判定データ

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。

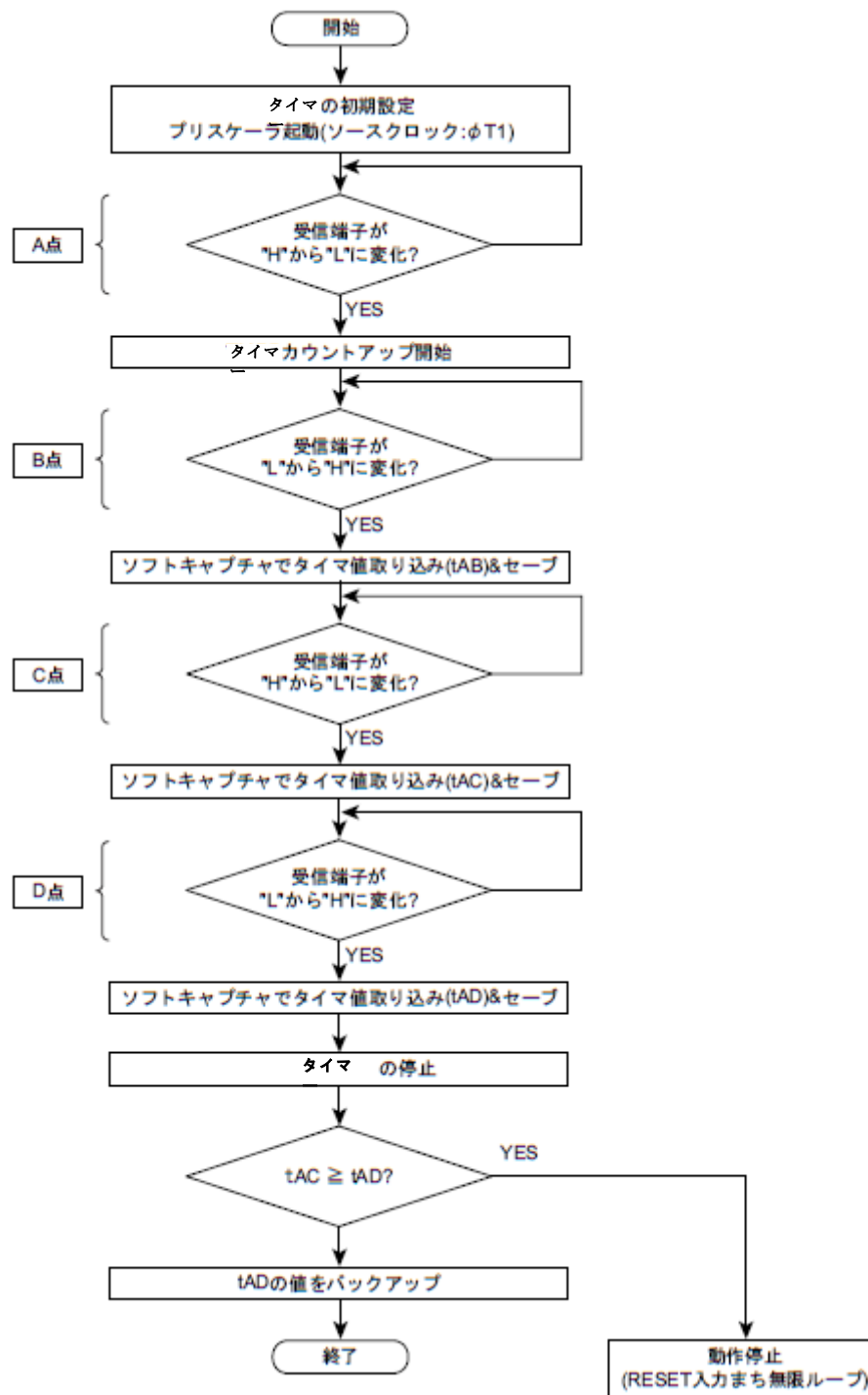


図 6.14 シリアル動作モード受信フローチャート

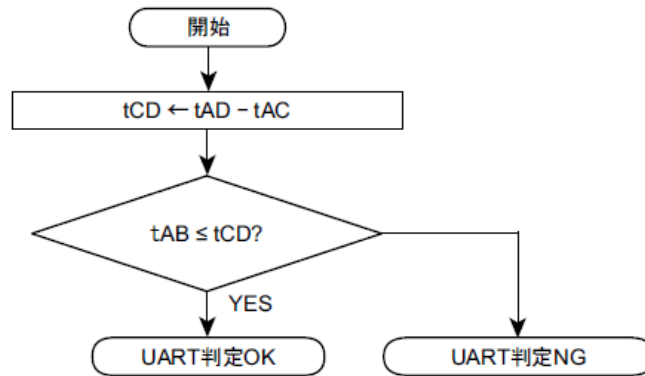


図 6.15 シリアル動作モード判定フローチャート

### 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
0x?8(注)	動作コマンドデータに受信エラーが発生した。
0x?1(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	フラッシュメモリ消去コマンド終了
0x4C	フラッシュメモリ消去コマンドが不正に終了した
0x47	フラッシュメモリ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×4)+1)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)またはパワーオンリセット(POR)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。

注) パワーオンリセット(POR)からのリセット解除によるシングルブートモードの起動は、一部製品のみ対応しています。詳細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。

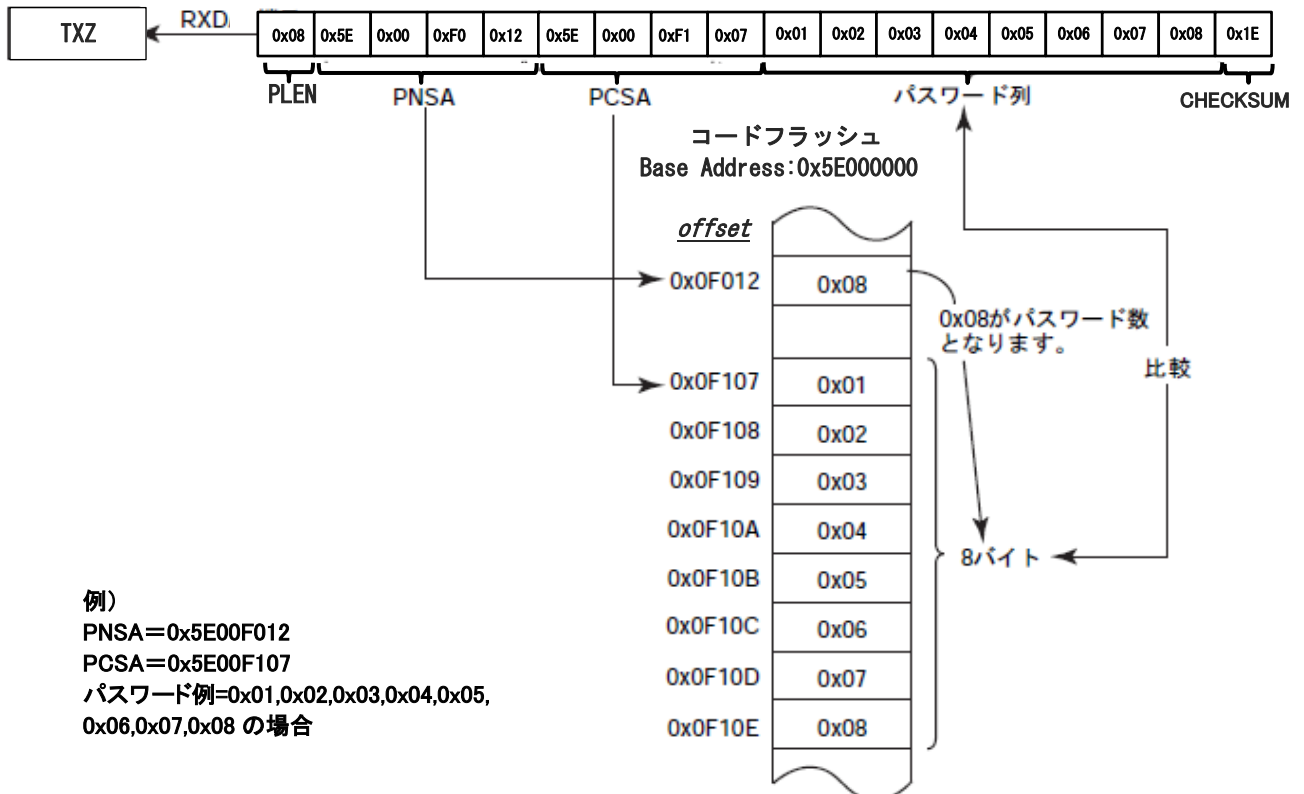


図 6.16 パスワードの構成(送信例)

### (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

コードフラッシュおよびデータ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。

コードフラッシュは全領域“0xFF”となっているが、データフラッシュにデータが残っているような場合はパスワードエラーとなりますので、チップ消去を実行してください。

#### (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)	$0x5E001000 \leq \text{PNSA} \leq \text{最大メモリーアドレス}$	$0x5E001000 \leq \text{PNSA} \leq \text{最大メモリーアドレス}$
PCSA (パスワード比較開始アドレス)	$0x5E001000 \leq \text{PCSA} \leq \text{最大メモリーアドレス}$	$0x5E001000 \leq \text{PCSA} \leq \text{最大メモリーアドレス} - (N \times 4) + 1$
N (パスワード数)	必要 (注 2)	$8 \leq N$
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。

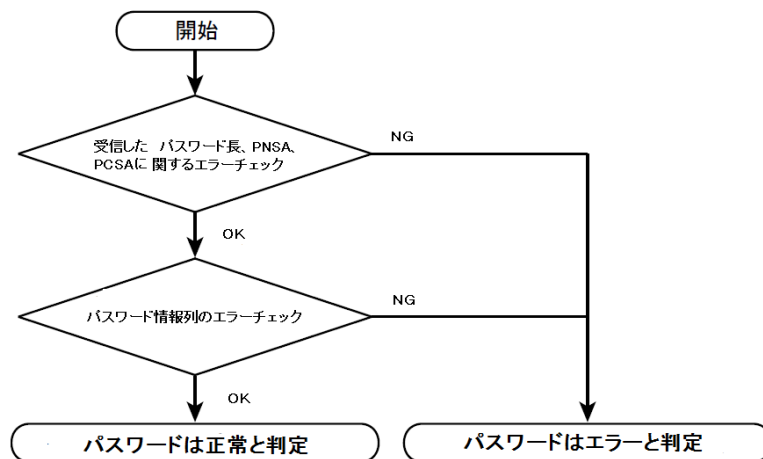


図 6.17 パスワードチェックフローチャート

#### 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、ポーレート設定	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、ポ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 (PNSA) (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 (PCSA) (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「6.6.6.5 CHECKSUM 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。
10	C→T	RAM 格納開始アドレス 23 ~ 16	
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	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	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。
14	C→T	RAM 格納バイト数 7~0	
15	C→T	送信データ (No.9 ~ 14) の CHECKSUM 値	コントローラー側は送信データ(No.9~14)の CHECKSUM 値を送信してください。
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	CHECKSUM 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。

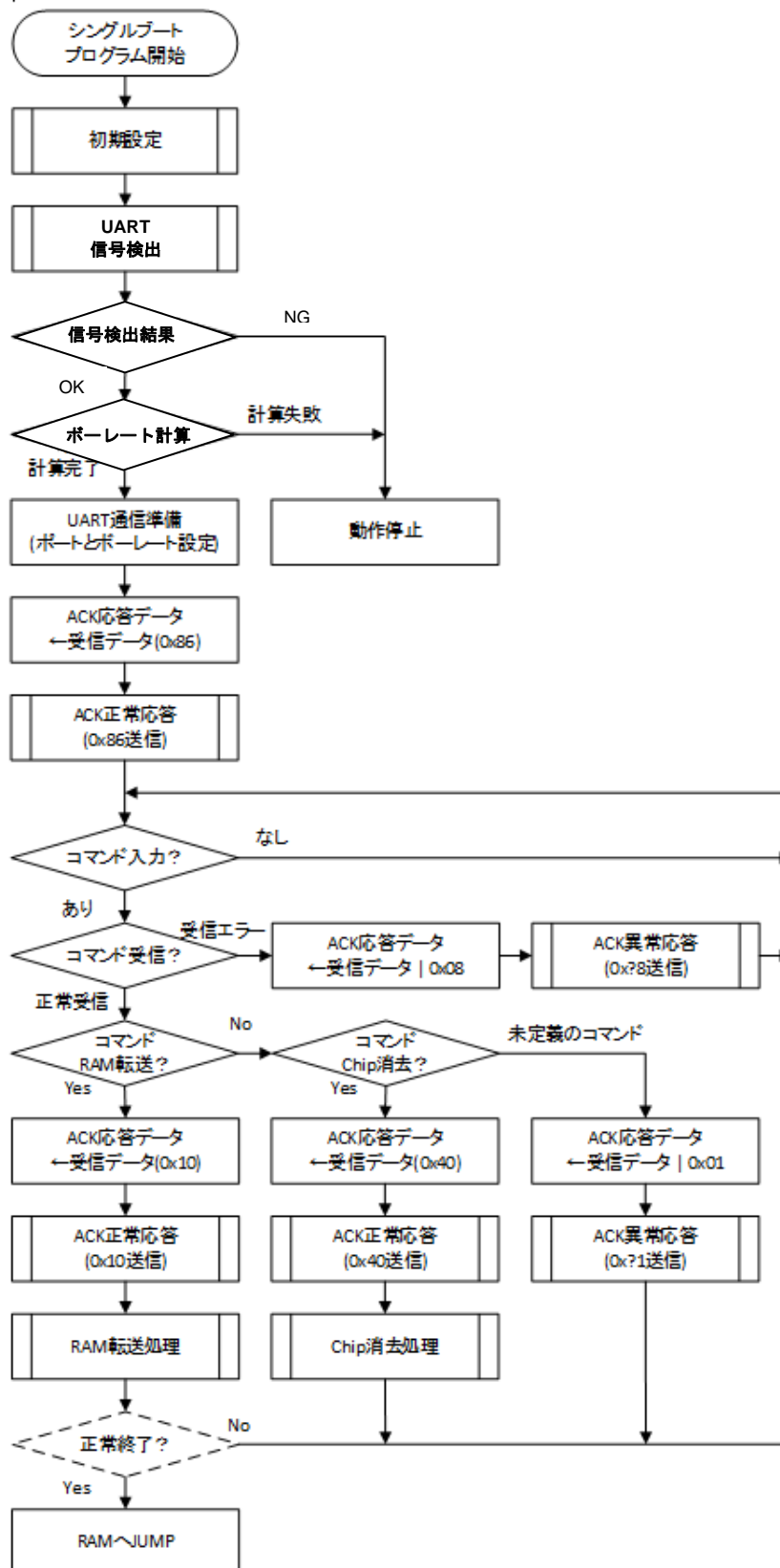


図 6.18 内蔵ブートプログラム全体フローチャート

## 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。

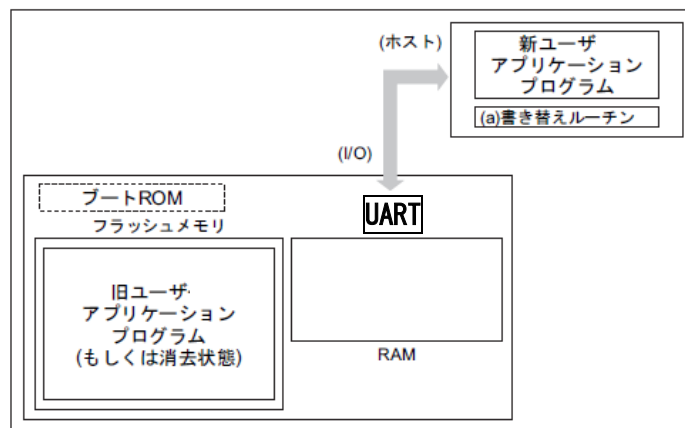


図 6.19 内蔵ブートプログラムを利用した書き替え手順(1)

### 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。

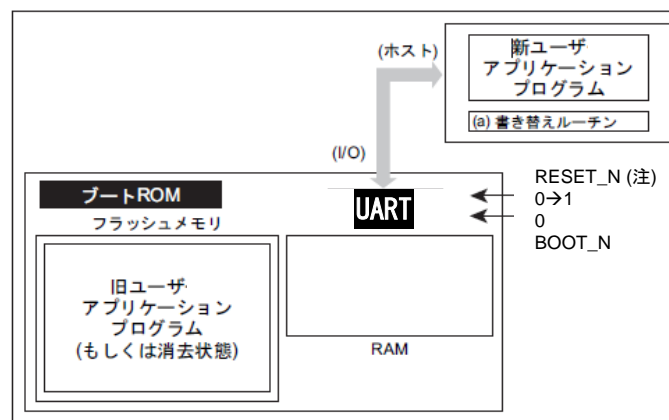


図 6.20 内蔵ブートプログラムを利用した書き替え手順(2)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

### 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。

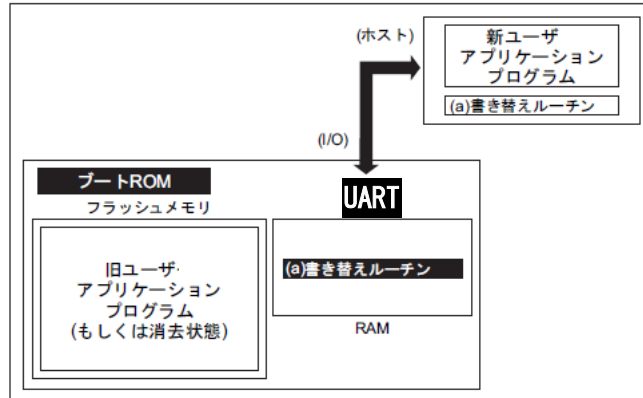


図 6.21 内蔵ブートプログラムを利用した書き替え手順(3)

### 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)

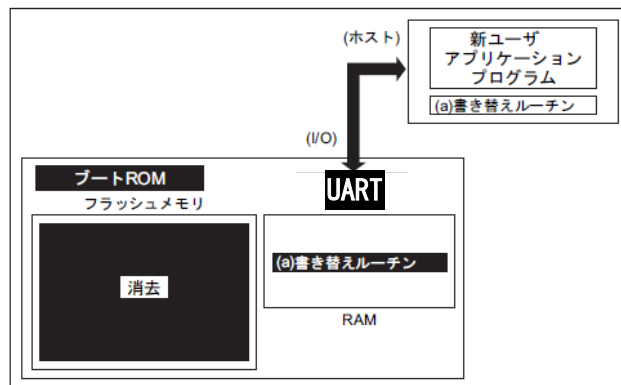


図 6.22 内蔵ブートプログラムを利用した書き替え手順(4)

## 6.6.11.5. Step-5

さらに、RAM 上の(a)書き換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、書き換えルーチンを転送したときと同じホストおよび UART 経由で書き換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換えルーチンを組み立ててください。

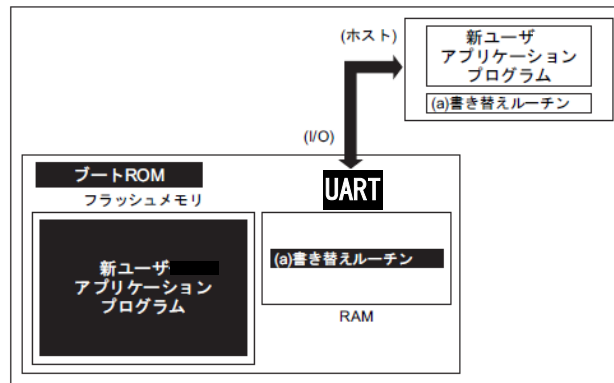


図 6.23 内蔵ブートプログラムを利用した書き換え手順(5)

## 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。

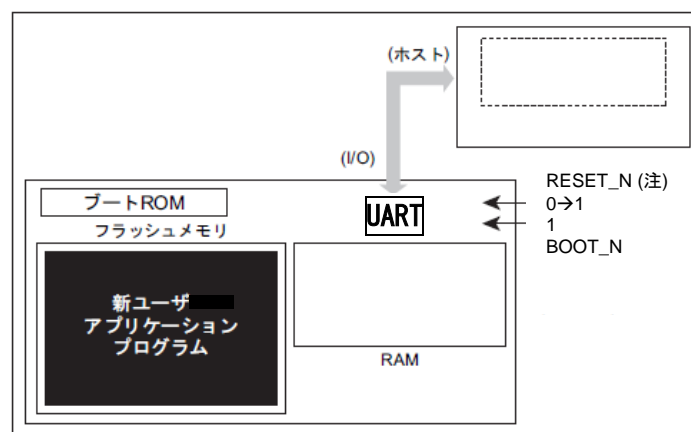


図 6.24 内蔵ブートプログラムを利用した書き換え手順(6)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

## 6.7. デュアルモードによる書き替え方法

ユーザーのセット上で、特定のブロックに置くフラッシュメモリー書き替えルーチンでフラッシュメモリーの書き替えを実行する方法です。

FLASH I/F0(エリア0)のフラッシュメモリー上でプログラムを実行中に、命令実行を行っていない他のFLASH I/F(例えばFLASH I/F2のエリア4:データフラッシュ)のフラッシュメモリーを書き込み/消去できません(条件によっては逆も可能です)。フラッシュメモリーの書き込み/消去とプログラム実行を同じFLASH I/Fのフラッシュメモリー上で行えませんので、フラッシュメモリーの書き込み/消去とプログラム実行は別々のFLASH I/Fで行ってください。

デュアルモード中に例外を使用する場合は、フラッシュメモリーの書き込み/消去を行うエリアを誤って実行しないよう留意ください。

### 6.7.1. フラッシュ書き替えの手順例

#### 6.7.1.1. 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)に設定されたらオンボードプログラミングに移行するか、どのFLASH I/Fのフラッシュメモリーを書き換えるかを決め、それに合った回路の設計、プログラムの作成を行います。

- (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

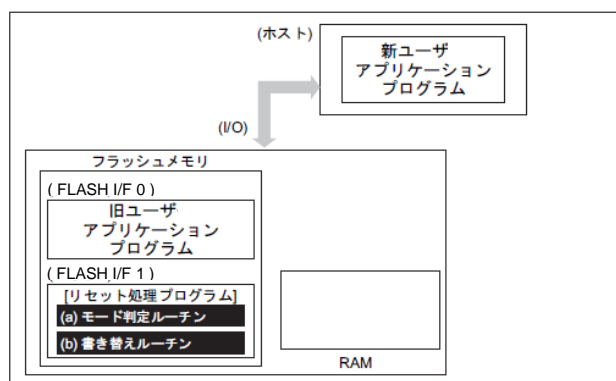


図 6.25 デュアルモードでフラッシュ書き替えの手順(1)

## 6.7.1.2. 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デュアルモードへの移行を判定します。移行条件が整っている場合、プログラムはフラッシュ書き替えルーチンへジャンプします(デュアルモードへの移行)。

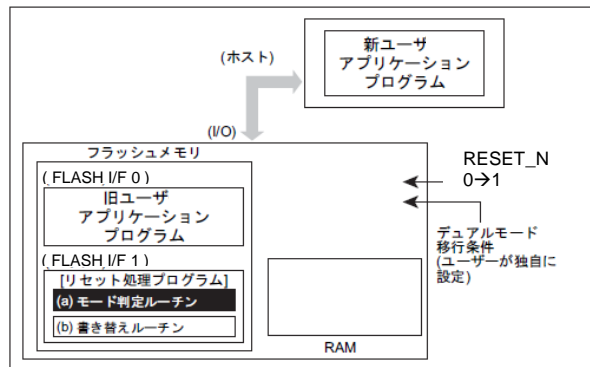


図 6.26 デュアルモードでフラッシュ書き替えの手順(2)

## 6.7.1.3. Step-3

フラッシュ書き替えルーチンへジャンプすると、まず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(エリア消去、ブロック単位、またはページ消去)を行います。

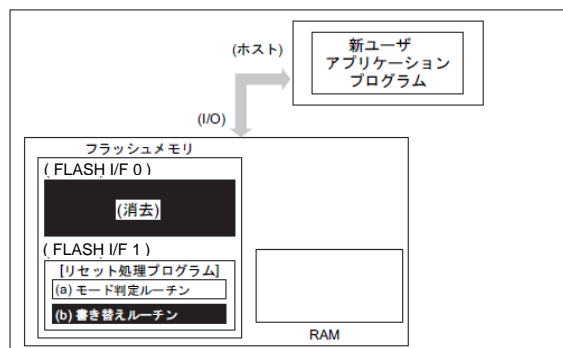


図 6.27 デュアルモードでフラッシュ書き替えの手順(3)

## 6.7.1.4. Step-4

次に、フラッシュメモリーの消去した領域がブランク状態であることを確認し、その後転送元(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、RAM上に展開します。

RAM上に展開したデータをフラッシュメモリーの消去した領域に書き込みます。全てのデータの書き込みが完了したら、フラッシュメモリーに書き込んだ領域のライト/消去プロテクトをオンにします。

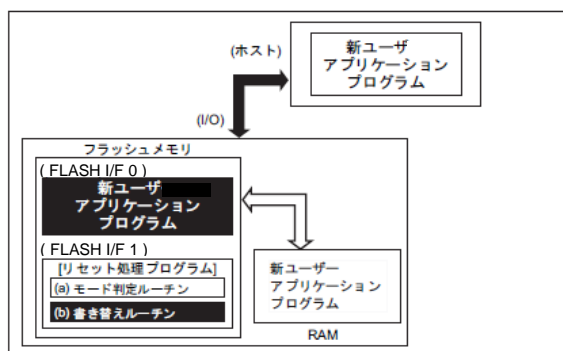


図 6.28 デュアルモードでフラッシュ書き替えの手順(4)

## 6.7.1.5. Step-5

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。

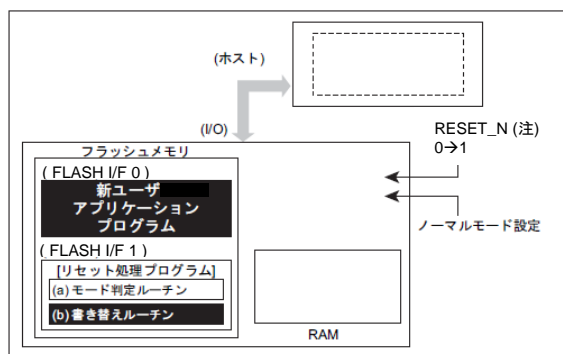


図 6.29 デュアルモードでフラッシュ書き替えの手順(5)

注) 製品によっては、RESET\_N 端子の代わりにパワーオンリセット(POR)を使用することができます。詳細は、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。

## 6.8. 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。

### 6.8.1. フラッシュ書き替えの手順例

#### 6.8.1.1. Step-1

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

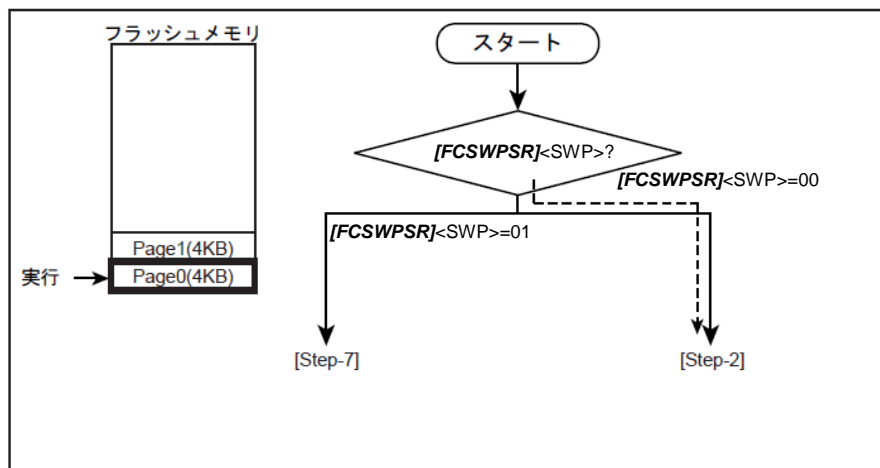


図 6.30 ユーザーブートプログラムの書き替え(1)



## 6.8.1.2. 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。

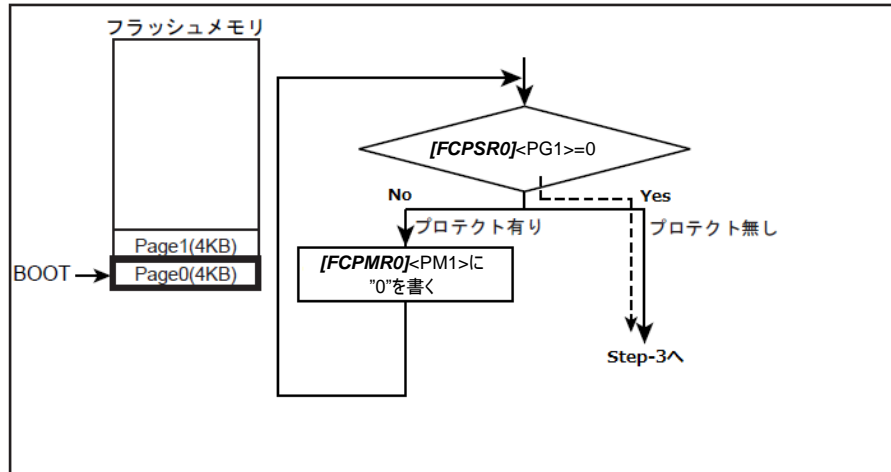


図 6.31 ユーザーブートプログラムの書き替え(2)

## 6.8.1.3. 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。

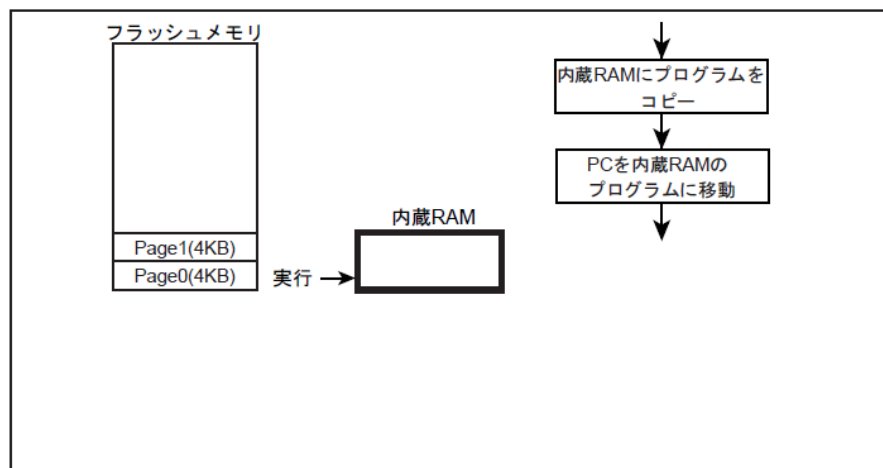


図 6.32 ユーザーブートプログラムの書き替え(3)

## 6.8.1.4. 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。

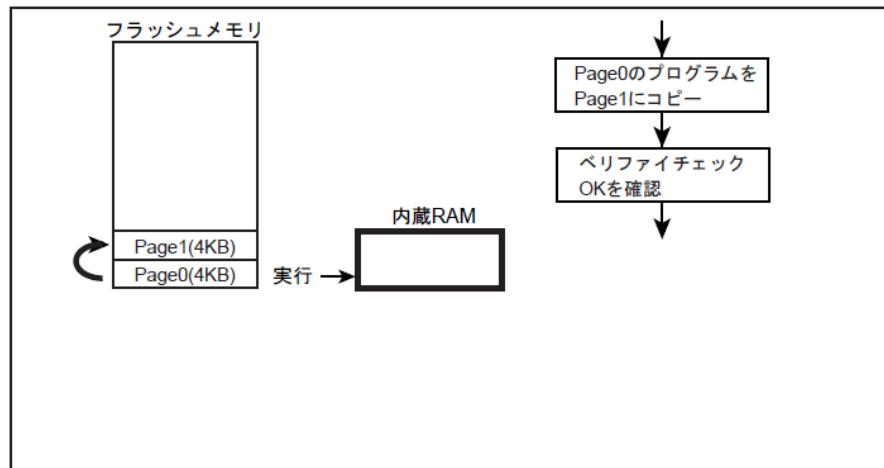


図 6.33 ユーザーブートプログラムの書き替え(4)

## 6.8.1.5. 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。

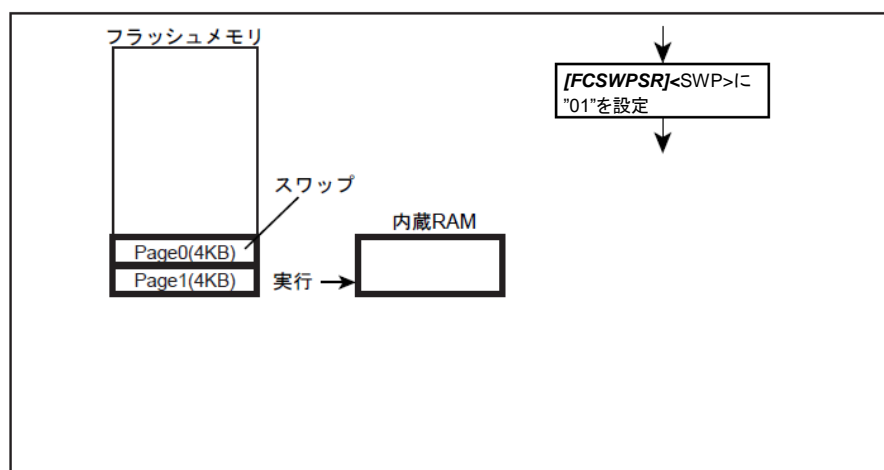


図 6.34 ユーザーブートプログラムの書き替え(5)

## 6.8.1.6. 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、 $[FCSWPSR]\langle SWP[1:0]\rangle$  に "01" の条件用ルーチンへ分岐します。 ([Step-7]へ)

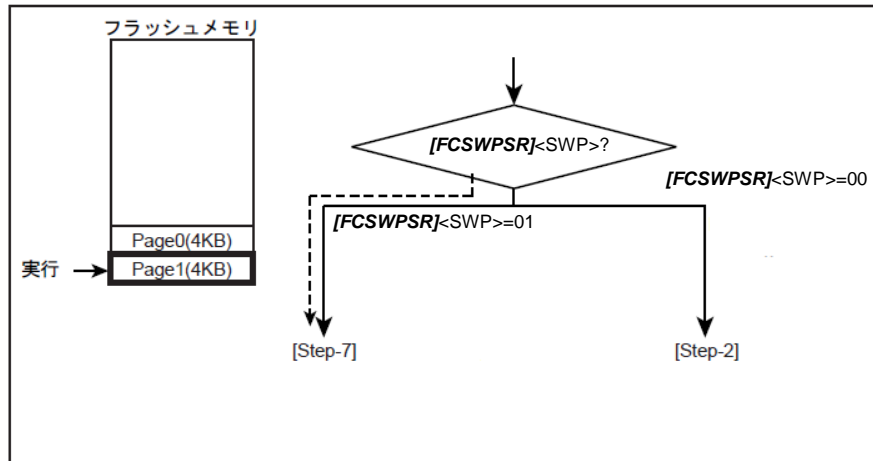


図 6.35 ユーザーブートプログラムの書き替え(6)

## 6.8.1.7. 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。

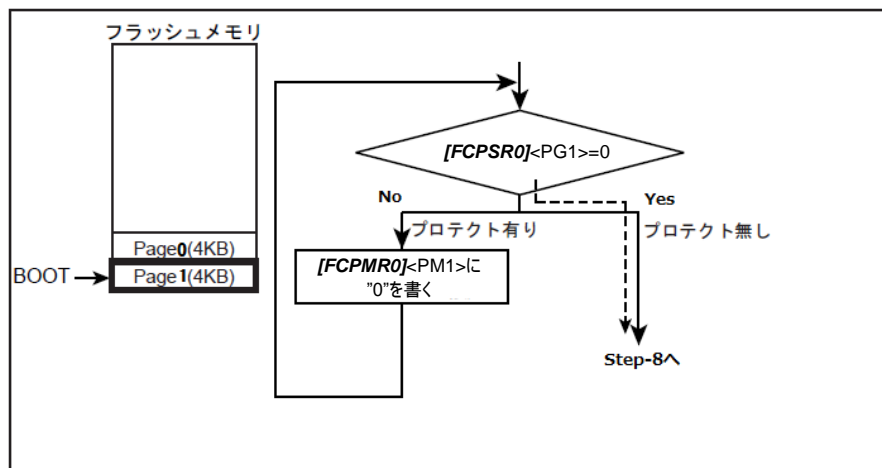


図 6.36 ユーザーブートプログラムの書き替え(7)

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。

## 6.8.1.8. 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。

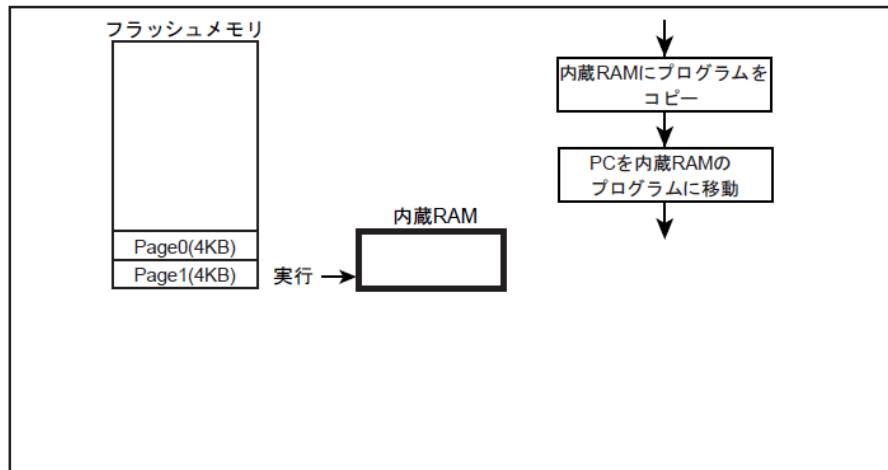


図 6.37 ユーザーブートプログラムの書き替え(8)

## 6.8.1.9. Step-9

新しいブートプログラムを Page0 に書き込みます。

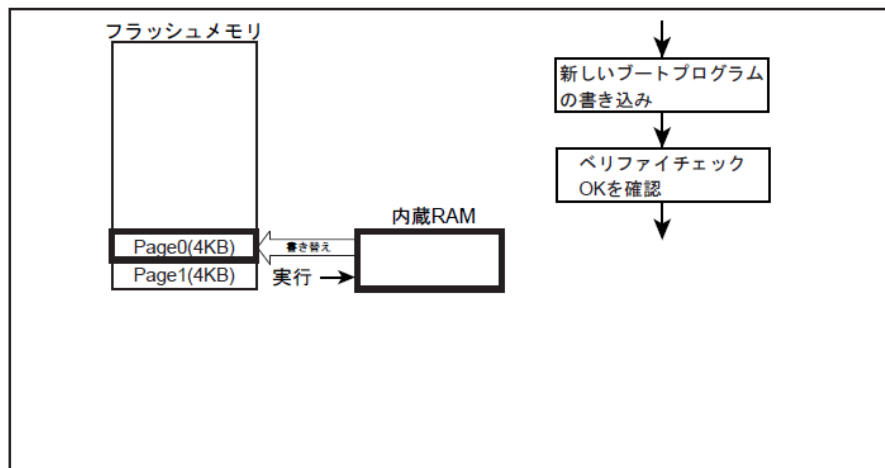


図 6.38 ユーザーブートプログラムの書き替え(9)

## 6.8.1.10. Step-10

自動メモリ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。

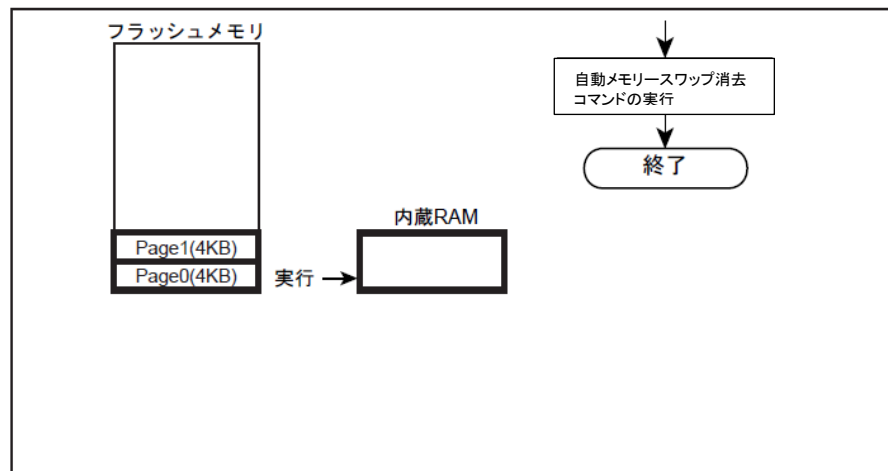


図 6.39 ユーザーブートプログラムの書き替え(10)

## 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。
- 正常にプログラム/消去ができたか、コマンド実行後に読み出して確認することを推奨します。

8. 改訂履歴

表 8.1 改訂履歴

Revision	Date	Description
1.0	2017-12-13	<p>新規</p> <ul style="list-style-type: none"> <li>・関連するドキュメント タイトルのリファレンスマニュアル→ドキュメントに修正</li> <li>・1. 概要 <ul style="list-style-type: none"> <li>表 1.1 の機能 リード/ライトプロテクト機能をリード/ライトプロテクト機能とセキュリティに分割し、表 1.1 下に注)を追加</li> <li>表 1.2 の領域 ユーザーインフォメーションエリアの下に 4KB を追加</li> <li>表 1.2 の機能 リード/ライトプロテクト機能の動作説明を見直し</li> </ul> </li> <li>・2.1 ブロック図 <ul style="list-style-type: none"> <li>図 2.1 の修正 FLASH I/F0(1)の分割、図 2.1 の修正による表 2.1 の見直し</li> </ul> </li> <li>・2.2.2 コードフラッシュのブロック構成 <ul style="list-style-type: none"> <li>表 2.2 の FLASH I/F1 の GP0 の終了アドレス修正</li> </ul> </li> <li>・2.2.3 コードフラッシュのページ構成 <ul style="list-style-type: none"> <li>FLASH I/F1 の Page252~255→Page124~127</li> <li>表 2.6,表 2.7,表 2.8 の Page128~135 のアドレス修正</li> </ul> </li> <li>・2.2.6 メモリー容量と構成 <ul style="list-style-type: none"> <li>表 2.11 に注 4)を追加</li> </ul> </li> <li>・3.1.1.2 バスライトサイクル時のアドレスビット構成 <ul style="list-style-type: none"> <li>表 3.3 の[通常コマンド][自動チップ消去]の推奨値の Adr 表記を"0"に修正</li> <li>PBA の説明 プロテクトビットアドレス→プロテクトビット消去に修正</li> <li>MSA の説明 メモリースワッププログラム→メモリースワップ消去に修正</li> </ul> </li> <li>・3.1.1.4 プロテクトビットの指定 <ul style="list-style-type: none"> <li>FLASH I/F0 の 25 ブロックのページ 107→207 に修正</li> </ul> </li> <li>・3.2.1.2 バスライトサイクル時のアドレスビット構成(データフラッシュ) <ul style="list-style-type: none"> <li>表 3.8 の注) 0 固定で使用してください。→ エリアは"0"固定で使用してください。に修正</li> </ul> </li> <li>・3.3.1 自動プログラム <ul style="list-style-type: none"> <li>図 3.1 の[FCAREASEL]&lt;AREAn&gt;にて実行対象 →[FCAREASEL]&lt;AREAn&gt;で実行対象に修正</li> <li>図 3.1 下の注)をフローチャート内に記載</li> </ul> </li> <li>・3.3.2 自動消去 <ul style="list-style-type: none"> <li>図 3.3 の[FCAREASEL]&lt;AREAn&gt;にて実行対象 →[FCAREASEL]&lt;AREAn&gt;で実行対象に修正</li> <li>図 3.4 自動ページ消去コマンドシーケンスエリア 0 Page0~255 の例にタイトル修正</li> <li>図 3.4 自動コードエリア消去コマンドシーケンスのエリア 0 の例を削除</li> </ul> </li> <li>・3.3.3 プロテクトビット 章の追加</li> <li>・3.3.4 セキュリティビット 章の追加</li> <li>・3.3.5 メモリースワップ 章の追加</li> <li>・4 フラッシュメモリー詳細 "他の FLASH I/F0"→"他の FLASH I/F"に修正</li> <li>・4.1.2 コマンド実行方法 "幾つか"→"いくつか"に修正</li> <li>・4.1.3.8 自動セキュリティビットプログラム (2)実行方法にシステムリセットで有効になる説明を追加</li> <li>・4.1.3.9 自動セキュリティビット消去 (2)実行方法にシステムリセットで有効になる説明を追加</li> <li>・4.1.5 自動動作の完了検知 割り込み発生タイミングの説明を見直し</li> <li>・4.1.7.1 セキュリティの設定方法 システムリセットで有効になる説明を追加</li> <li>・4.1.9.1 ユーザーインフォメーションエリアの切り替え手順 ユーザーインフォメーションメモリーをユーザーインフォメーションエリアに修正</li> </ul>
2.0	2018-06-05	<ul style="list-style-type: none"> <li>・3.3.1 自動プログラム 図 3.1 の[FCAREASEL]&lt;AREAn&gt;にて実行対象 →[FCAREASEL]&lt;AREAn&gt;で実行対象に修正</li> <li>図 3.1 下の注)をフローチャート内に記載</li> <li>・3.3.2 自動消去 図 3.3 の[FCAREASEL]&lt;AREAn&gt;にて実行対象 →[FCAREASEL]&lt;AREAn&gt;で実行対象に修正</li> <li>図 3.4 自動ページ消去コマンドシーケンスエリア 0 Page0~255 の例にタイトル修正</li> <li>図 3.4 自動コードエリア消去コマンドシーケンスのエリア 0 の例を削除</li> <li>・3.3.3 プロテクトビット 章の追加</li> <li>・3.3.4 セキュリティビット 章の追加</li> <li>・3.3.5 メモリースワップ 章の追加</li> <li>・4 フラッシュメモリー詳細 "他の FLASH I/F0"→"他の FLASH I/F"に修正</li> <li>・4.1.2 コマンド実行方法 "幾つか"→"いくつか"に修正</li> <li>・4.1.3.8 自動セキュリティビットプログラム (2)実行方法にシステムリセットで有効になる説明を追加</li> <li>・4.1.3.9 自動セキュリティビット消去 (2)実行方法にシステムリセットで有効になる説明を追加</li> <li>・4.1.5 自動動作の完了検知 割り込み発生タイミングの説明を見直し</li> <li>・4.1.7.1 セキュリティの設定方法 システムリセットで有効になる説明を追加</li> <li>・4.1.9.1 ユーザーインフォメーションエリアの切り替え手順 ユーザーインフォメーションメモリーをユーザーインフォメーションエリアに修正</li> </ul>

		<ul style="list-style-type: none"> <li>・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 ユーザーインフォメーションメモリをユーザーインフォメーションエリアに修正 説明見直し</li> <li>・4.1.9.3 ユーザーインフォメーションメモリの消去方法 ユーザーインフォメーションメモリをユーザーインフォメーションエリアに修正 説明見直し</li> <li>・5.2.4 [FCSR0] ビット[31:11]の説明見直し</li> <li>・5.2.13 [FCPMR4] ビット[31:16]のリセット後/Type/機能の内容修正</li> <li>・5.2.14 [FCPMR6] ビット[15:8]のリセット後/Type/機能の内容修正</li> <li>・5.2.17 [FCAREASEL] ビット[22:20]のリセット後の値 “-“ → ”000” に修正 ビット[15:12]の説明見直し</li> <li>・5.2.21 [FCBUFDISCLR] ユーザーインフォメーションメモリをユーザーインフォメーションエリアに修正</li> <li>・6.6.3 インターフェース仕様 表 6.4 の 4 行上の説明 “タイマーカウンタ” → “タイマーカウンタ(T32A)”に修正</li> <li>・6.6.6.3 パスワード (3) 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をユーザーインフォメーション エリアに修正</li> <li>・6.6.10 内蔵ブートプログラム全体フローチャート 上から 3 段目 ポート自動判別処理→ポーレート検出に修正 上から 4 段目 ポート判別→ポーレート検出結果に修正</li> <li>・6.6.11.6 Step-6 図 6.24 の”シングルチップモード”を削除</li> </ul>
<p>2.1</p>	<p>2019-07-03</p>	<ul style="list-style-type: none"> <li>・1.1 メモリーマップ 図 1.1 のメモリーマップ変更</li> <li>・2.2.6 メモリー容量と構成 チップ消去時間の修正 注 3)の見直し</li> <li>・3 機能説明・動作説明 使用上の注意事項(1)(2) 内蔵高速発振器→内蔵高速発振器 1 に修正</li> <li>・3.3.3 プロテクトビット 図 3.5 下から 3 つめの箱 (注)→(注 2)に修正</li> <li>・3.3.5 メモリースワップ 図 3.9 [FCAREASEL]&lt;AREAn&gt; →[FCAREASEL]&lt;AREA0&gt;に修正、注)を削除 図 3.10 メモリースワップコマンド→自動メモリースワッププログラムコマンド に修正 メモリースワップ消去→自動メモリースワップ消去 に修正 図 3.10 自動メモリースワップ消去コマンドの最終箱 メモリースワップアドレス→ 0x5E003000 に修正</li> <li>・4.1 機能 表 4.1 自動セキュリティー→自動セキュリティービットプログラム/消去 に修正 自動メモリースワップ→自動メモリースワッププログラム/消去 に修正</li> <li>・4.1.2 コマンド実行方法 4.の「全てのエリア選択ビット」の設定から、自動チップ消去コマンド以外を削除</li> <li>・4.1.3.12 自動メモリースワッププログラム タイトルを自動メモリースワップ→自動メモリースワッププログラム に修正 (2)の説明 消去が行われたか→書き込みきたかどうか に修正</li> <li>・4.1.6.1 プロテクトの設定方法 説明追加</li> <li>・4.1.7.1 セキュリティーの設定方法 説明追加</li> <li>・4.1.7.2 セキュリティーの解除方法 注)を追加</li> <li>・6.5.1.2 Step-2 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.5.1.6 Step-6 「製品固有情報」→「製品個別情報」に修正</li> </ul>



		<ul style="list-style-type: none"> <li>・6.5.2.2 Step-2 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.5.2.6 Step-6 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.6.3 インターフェース仕様 タイマーカウンター(T32A)→タイマーカウンター に修正 注 2) “H”固定 → High 固定に修正</li> <li>・6.6.4 メモリーの制約について 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.6.6.3 パスワード (2)PNSA 最大メモリアドレス内→(最大メモリアドレス)内に修正</li> <li>・6.5.11.2 Step-2 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.5.11.6 Step-6 「製品固有情報」→「製品個別情報」に修正</li> <li>・6.7.1.5 Step-5 「製品固有情報」→「製品個別情報」に修正</li> <li>・7.使用上のご注意およびお願い事項 お願い事項の追加</li> </ul>
2.2	2022-07-07	<ul style="list-style-type: none"> <li>・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</li> </ul>

## 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。