

RH850/U2Bx

R01AN6438JJ0100

Rev.1.00

シングルマップモードによる GCFU を使用した OTA 動作例

要旨

本アプリケーションノートは、RH850/U2Bx のシングルマップモードによる OTA 動作例をまとめたものです。外部デバイスとのインタフェースとして、RS-CANFD を使用します。内蔵 Code Flash 書き換えプログラムはユーザマット上にあるものとします。

なお、本アプリケーションノートに掲載されているタスク例およびアプリケーション例は確認済みですが、実際にご使用になる場合には、必ず動作環境を確認の上ご使用下さいますようお願いいたします。

適用

この資料は、RH850/U2Bx に適用されます。

【注 1】セルフプログラミング機能有効化、CodeFlash メモリマッピング

本アプリケーションノートでは、OTA を行うために CS+上で下記設定を有効にしてください。

- (1)プロジェクト・ツリーから「***** (デバック・ツール)」を選択
- (2)「接続用設定」のタブを選択
- (3)「フラッシュ」の“Code Flash のセルフ・プログラミングを行う” = “はい” に設定
- (4)「メモリ」の“マップモード” = “シングルマップモード” に設定

目次

1. シングルマップモードによるOTA.....	3
1.1 アドレス変換機能	3
1.1.1 GCFUによるアドレス変換	3
1.1.2 アドレス変換後のFlashアクセス	4
1.1.3 GCFU使用時の制約事項	4
1.2 起動Bankによる書き込み動作	5
1.2.1 プログラムA起動.....	5
1.2.2 プログラムB起動.....	5
1.3 BGO機能	6
1.3.1 同時Prog/Erase機能.....	6
2. 仕様.....	7
2.1 全体仕様	7
2.2 RS-CANFD通信仕様	8
2.3 CANFDコマンド仕様	8
2.4 全体シーケンス	9
2.5 使用機能	11
2.6 動作モード	11
2.7 メモリマッピング	11
3. 外部デバイスを使用したOTA動作例	12
3.1 動作手順	12
3.1.1 ① 起動Bank切り替え	12
3.1.2 ② BGO書き込み	15
3.1.3 ③ ID認証	18
3.1.4 ④ Code Flash消去	19
3.1.5 ⑤ Code Flash書き込みデータダウンロード	22
3.1.6 ⑥ Code Flash書き込み	23
3.1.7 ⑦ 起動Bank情報書き込み関数	26
4. メモリ割り付け	27
4.1 アドレスマップ	27
4.1.1 アドレス配置図	27

1. シングルマップモードによる OTA

1.1 アドレス変換機能

1.1.1 GCFU によるアドレス変換

シングルマップモードでは GCFU によるアドレス変換機能を使用して、更新前・後のプログラムのマッピング切り替えを行います。部分的なプログラムの更新を行う場合に効率よく Code Flash を使用可能です。図 1-1 に GCFU によるアドレス変換を示します。

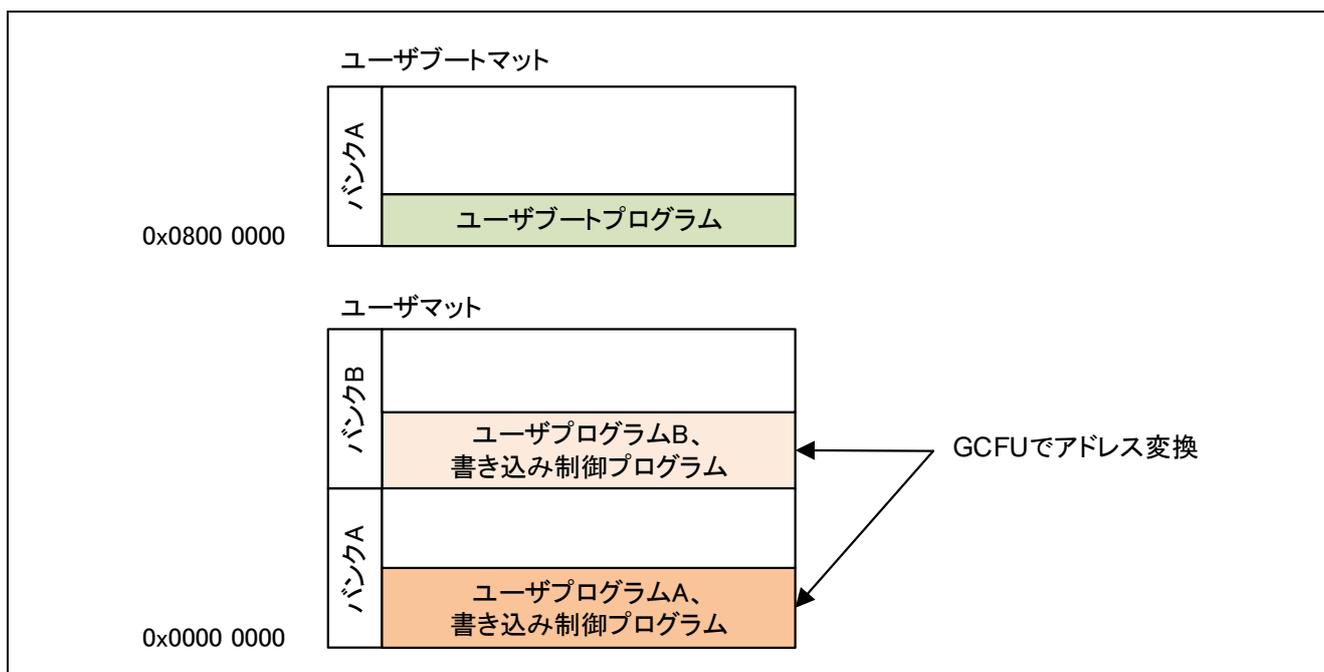


図 1-1 GCFU によるアドレス変換

1.1.2 アドレス変換後の Flash アクセス

GCFU でアドレス変換した場合、CPU からは Flash を論理アドレスとしてアクセスします。FACI は GCFU でのアドレス変換に依存しないため、物理アドレスとしてアクセスします。図 1-2 にメモリ関連バス構成図を示します。

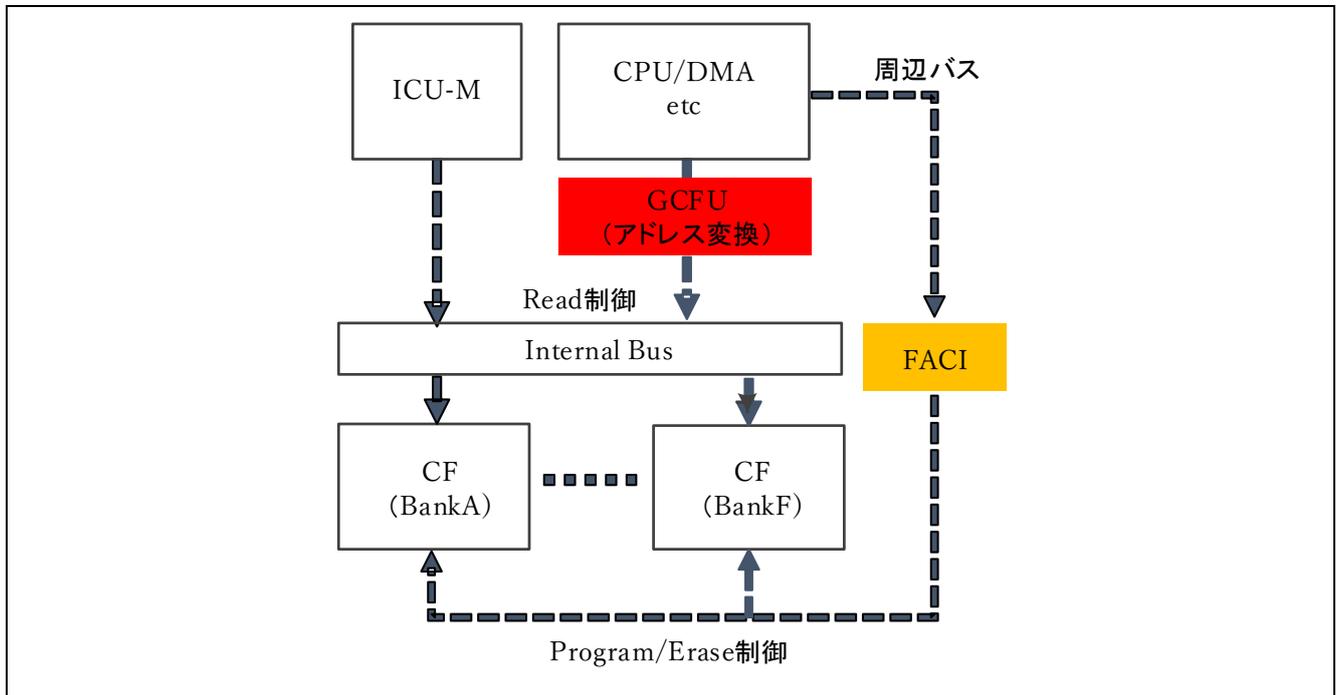


図 1-2 メモリ関連バス構成図

1.1.3 GCFU 使用時の制約事項

一部のブロックをアドレス変換する際、プログラム更新中にアドレス変換対象外のブロックにあるプログラム、及びデータをアクセスするとエラーが発生します。このため、プログラム更新中にアドレス変換対象外のブロックをアクセスする際は、消去書き込みをサスペンドして ROM リードモードにしてください。

また、ICU-M からの読み出しアクセスはアドレス変換対象外です。実行領域を意識したプログラミングが必要です。

1.2 起動 Bank による書き込み動作

次回、起動する Bank 情報を Data Flash に格納しておきます。起動する Bank は BankA の場合はアドレス変換せず、BankB の場合はアドレス変換します。起動 Bank による動作を以下に説明します。

1.2.1 プログラム A 起動

アドレス変換しません（物理アドレス＝論理アドレス）。書き込み対象は BankB になります。

図 1-3 にプログラム A 起動での書き込み動作を示します。

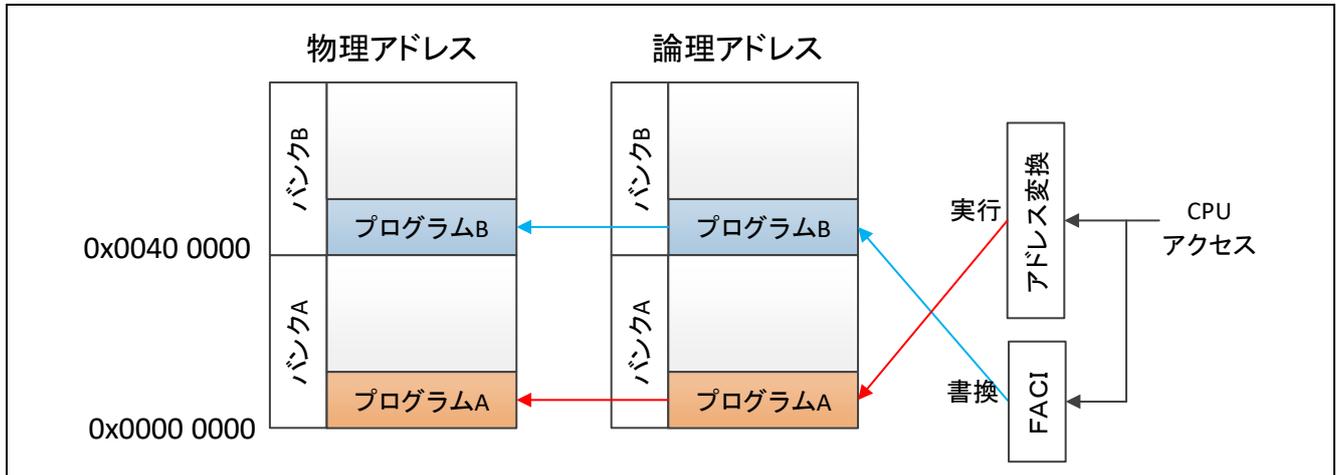


図 1-3 プログラム A 起動での書き込み動作

1.2.2 プログラム B 起動

アドレス変換します。（物理アドレス≠論理アドレス）、書き込み対象は BankA になります。

図 1-4 にプログラム B 起動での書き込み動作を示します。

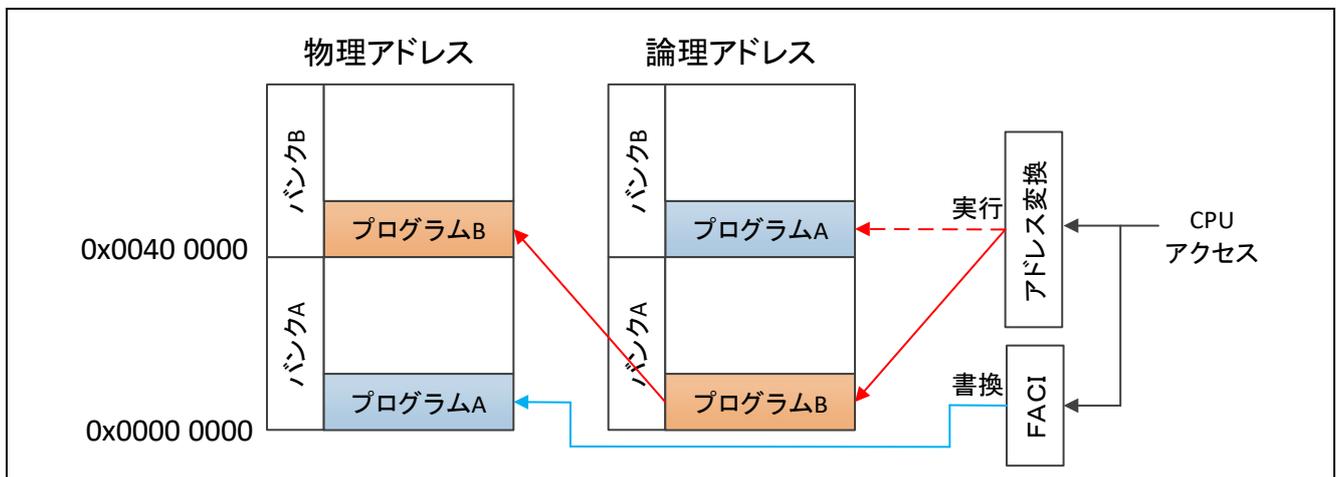


図 1-4 プログラム B 起動での書き込み動作

1.3 BGO 機能

1.3.1 同時 Prog/Erase 機能

BGO(Background operation)機能として、Code Flash の Program/Erase 中の Data Flash のリードを追加サポートします。また、異なるバンク間であれば、すべての Prog/Erase の組み合わせで、Prog/Erase のサスペンドが出来ます。本動作例では Code Flash 消去中に Data Flash の消去を優先して実行する同時 Prog/Erase を実現するため、Program/Erase サスペンド機能を使用しています。図 1-5に Flash メモリ関連モジュール構成図を示します。

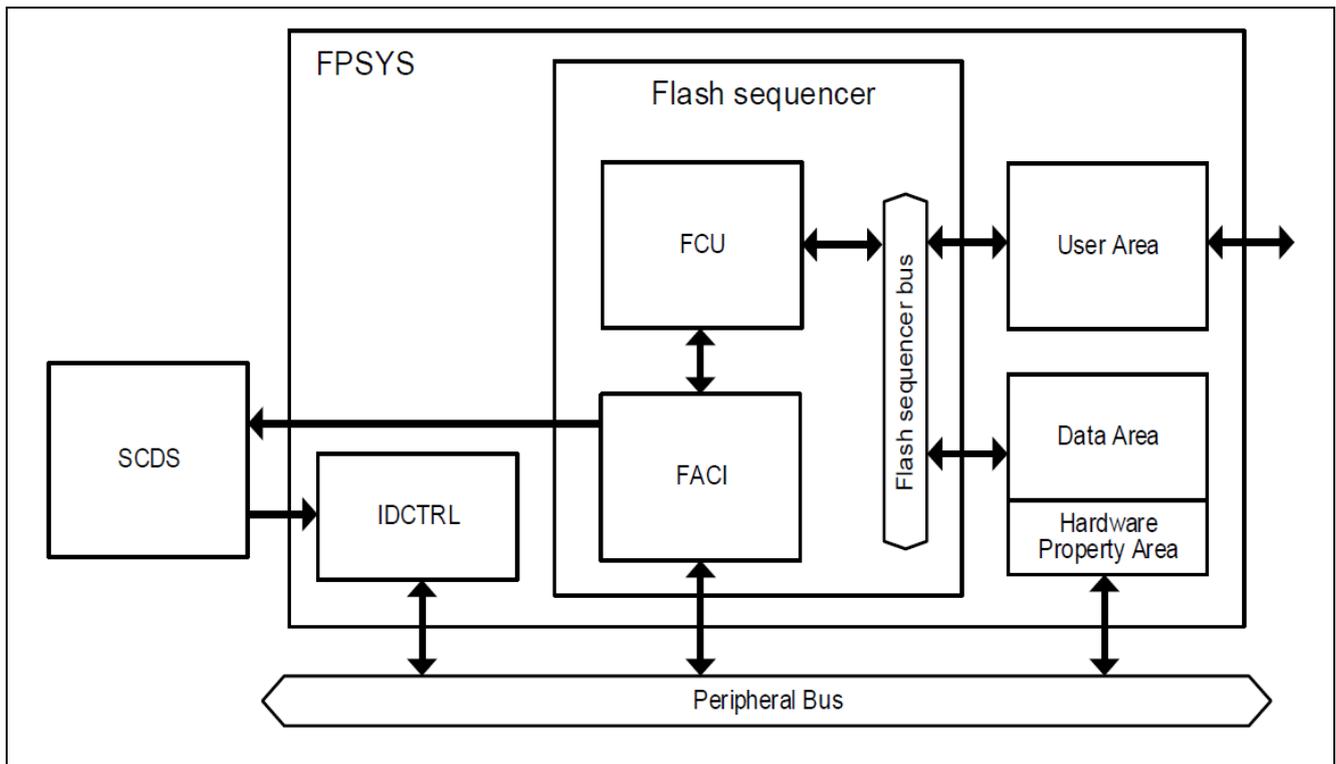


図 1-5 Flash メモリ関連モジュール構成図

2. 仕様

2.1 全体仕様

- 本アプリケーションノートは、シングルマップモードによる OTA 2Bank 構成で、ユーザマットの書き換えを行います。表 2-1にメモリ割り付けを示します。
- GCFU (Global Calibration Function Unit)機能を使用して、BankA 起動時は BankA を論理アドレス 0x00000000~0x0017FFFF として動作するように設定します、BankB 起動時には BankB を論理アドレス 0x00000000~0x0017FFFF として動作するように設定します。起動 Bank 情報は Data Flash に格納しており、起動 Bank を判別して GCFU 機能にてアドレス変換を行います。
- 書き換え領域はプログラムが動作する Bank とは異なる Bank に対して行います。
- Code Flash メモリマッピングモードはシングルマップモード、動作モードはノーマルオペレーションモード、起動マットはユーザブートマットです。
- Code Flash 書き込みに使用されるデータは RS-CANFD(ch1)を使用し、内蔵 RAM に格納されます。
- Code Flash 書き換え対象デバイスは、外部デバイスから RS-CANFD を使用して、特定の ID および、データを受信時に、対応した Code Flash 書き換え処理を行います。それら特定の ID および、データの組み合わせを本アプリケーションノートでは CANFD コマンドと呼びます。

図 2-1にシステム構成図を示します。

表 2-1 メモリ割り付け

領域	物理アドレス	ブロック	Bank	サイズ	OTA 対象
ユーザブートマット	0x0800 0000 – 0x0800 FFFF	ユーザブートエリア 0	BankA	64K バイト	非対象
ユーザマット (プログラム A)	0x0000 0000 – 0x0001 7FFF	ブロック 0-5	BankA	96K バイト	対象
ユーザマット (プログラム B)	0x0040 0000 – 0x0041 7FFF	ブロック 0-5	BankB	96K バイト	対象

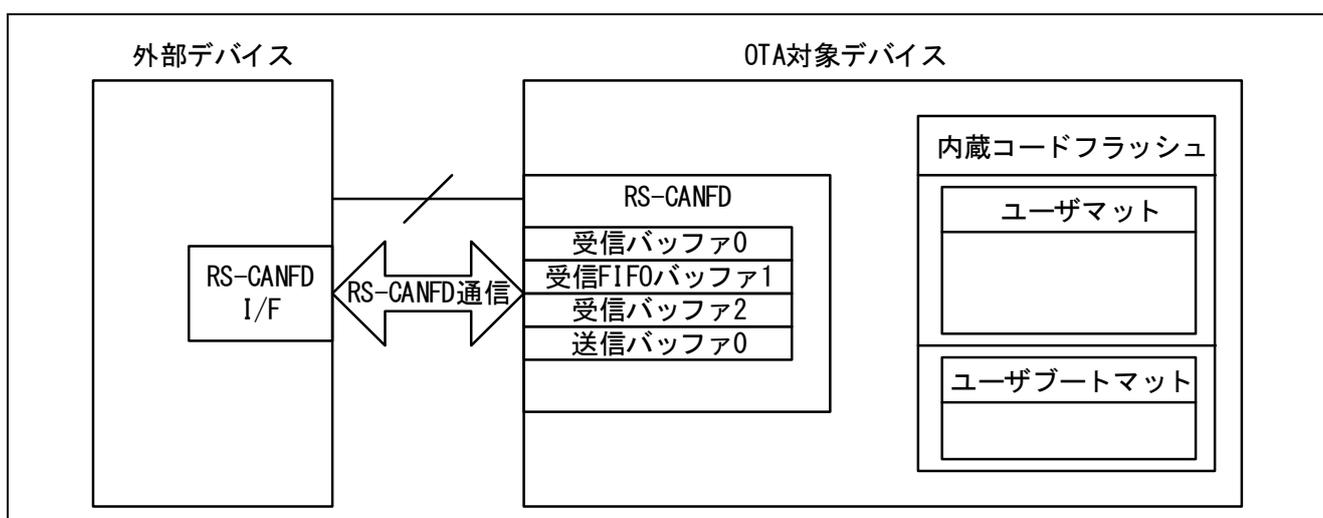


図 2-1 システム構成図

2.2 RS-CANFD 通信仕様

- チャンネルは 1 を使用します。
- 通信速度を通常ビットレート 1Mbps、データビットレート 2Mbps に設定します。
- 通信フレームを、CAN FD フレームに設定します。
- 外部デバイスから送信される各 CANFD コマンドを格納するために、チャンネル 1 の受信ルール数を 2 に設定します。

2.3 CANFD コマンド仕様

- 書き換え開始コマンドは、外部デバイスから Code Flash 書き換え対象デバイスに送信することにより Code Flash 書き換え処理が開始されます。
- 書き込みデータ要求コマンドは、Code Flash 書き換え対象デバイスから外部デバイスに送信することにより書き込みデータを要求します。
- 書き込みデータダウンロードコマンドは、外部デバイスから Code Flash 書き換え対象デバイスに書き込みデータを送信します。
- 書き込み終了コマンドは、Code Flash 書き換え対象デバイスから外部デバイスに送信することにより書き換え動作を終了します。

表 2-2に CANFD コマンド仕様を示します。

表 2-2 CANFD コマンド仕様

バッファ	チャンネル	コマンド名	送/受	標準 ID	データ長	データ
0	1	書き換え開始コマンド	受信	H'100	1Byte	H'00
1	1	書き込みデータダウンロード	受信	H'110	64Byte	Code Flash 書き込みデータ 512Byte をダウンロード(64Byte x 8)
1	1	書き込みデータ要求コマンド	送信	H'111	1Byte	H'11
2	1	書き込み終了コマンド	送信	H'121	1Byte	H'22

2.4 全体シーケンス

図 2-2~図 2-3に全体シーケンスを示します。

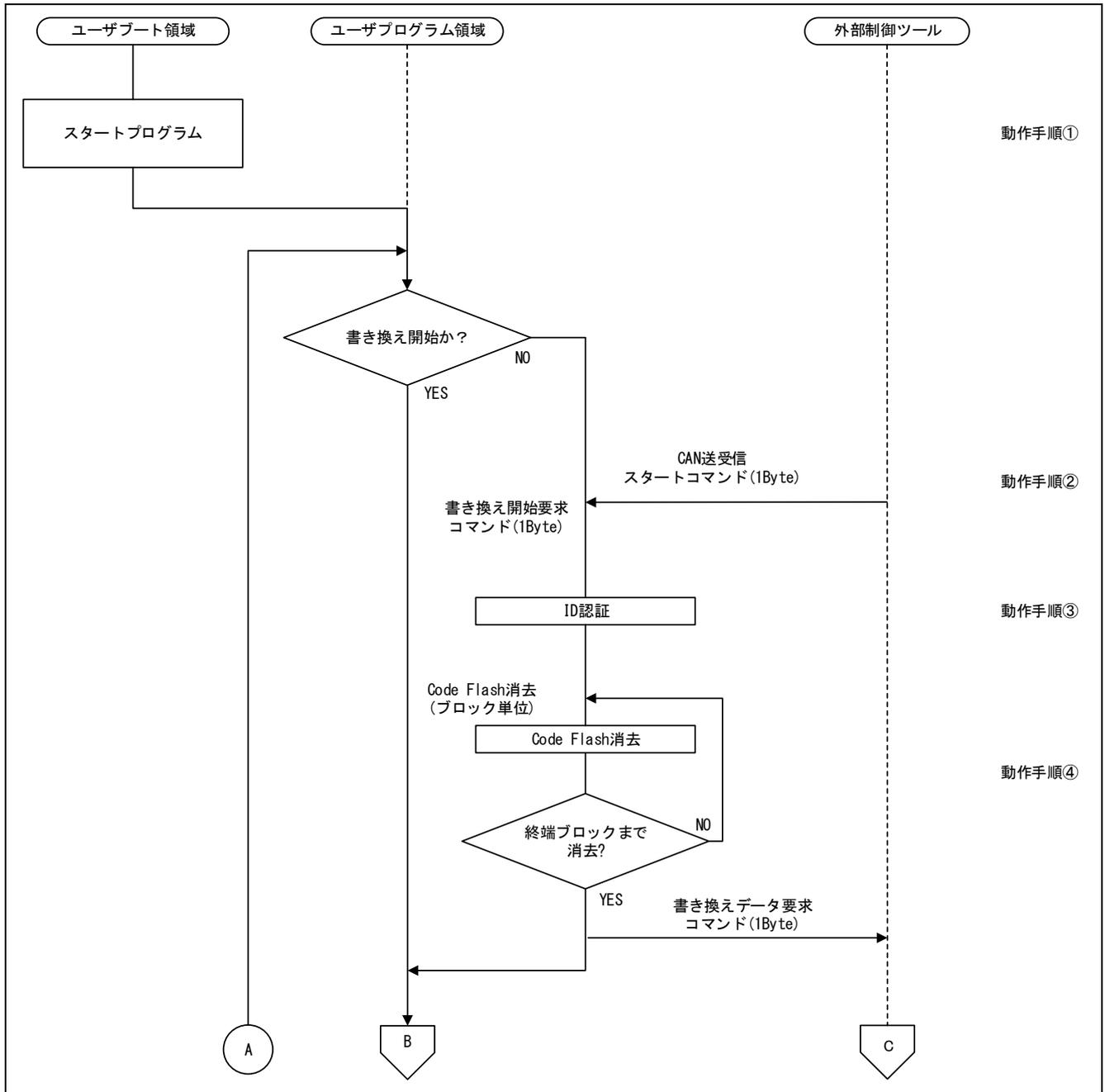


図 2-2 全体シーケンス図(1)

2.5 使用機能

- CANFD インタフェース(RS-CANFD)
- FACI
- GCFU
- 端子

2.6 動作モード

本アプリケーションノートでは、Code Flash 書き換え時のマイコンの動作モードはユーザーブートモードで行います。ユーザーブートモードでは起動マットはユーザーブートマットです。

動作モード選択方法は、モード端子で設定します。オプションバイトの設定は、RH850 ファミリー用 Renesas Flash Programmer を使用し設定します。

表 2-3に動作モード選択を示します。

表 2-3 動作モード選択

端子設定値			オプションバイト設定値		動作モード	起動マット
MD1	MD0	TRST	STMSEL1	STMSEL0		
0	0	0	0	1	ユーザーブートモード	ユーザーブートマット

2.7 メモリマッピング

本アプリケーションノートでは、Code Flash 書き換え時のメモリマッピングはシングルマップモードで行います。表 2-4にメモリマッピング選択を示します。

表 2-4 メモリマッピング選択

オプションバイト設定値		メモリマッピング
MAPMODE1	MAPMODE0	
0	1	シングルマップモード

3. 外部デバイスを使用した OTA 動作例

3.1 動作手順

動作手順①～⑦は、「2.4 全体シーケンス」に対応します。

3.1.1 ① 起動 Bank 切り替え

リセットスタート後、ユーザブートマップ上のスタートプログラムは Data Flash を読み出して、ユーザマップの起動 Bank を判定し、適切な Bank に切り替えてユーザプログラムにジャンプします。図 3-1に スタートプログラム動作を示します。

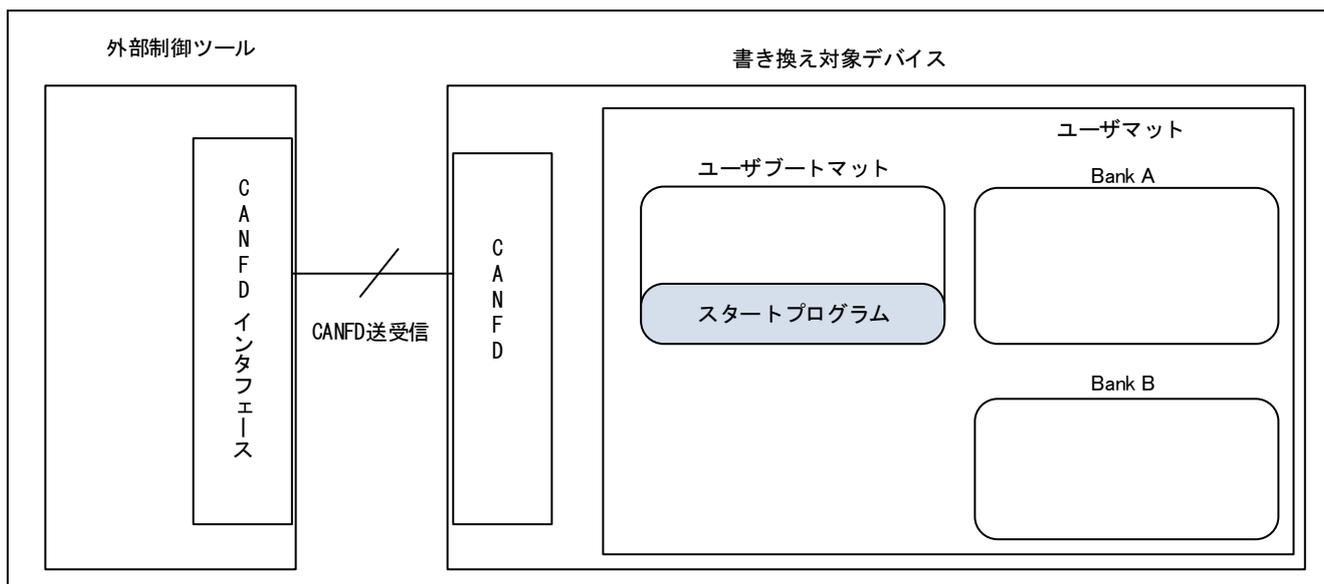


図 3-1 スタートアッププログラム動作

表 3-1 「main_pe0()関数」

関数名	概要
main_pe0()	プログラム開始。ユーザマップの起動 Bank 切り替え後、ユーザプログラムにジャンプします。

図 3-2に「main_pe0()関数」のフローチャートを示します。

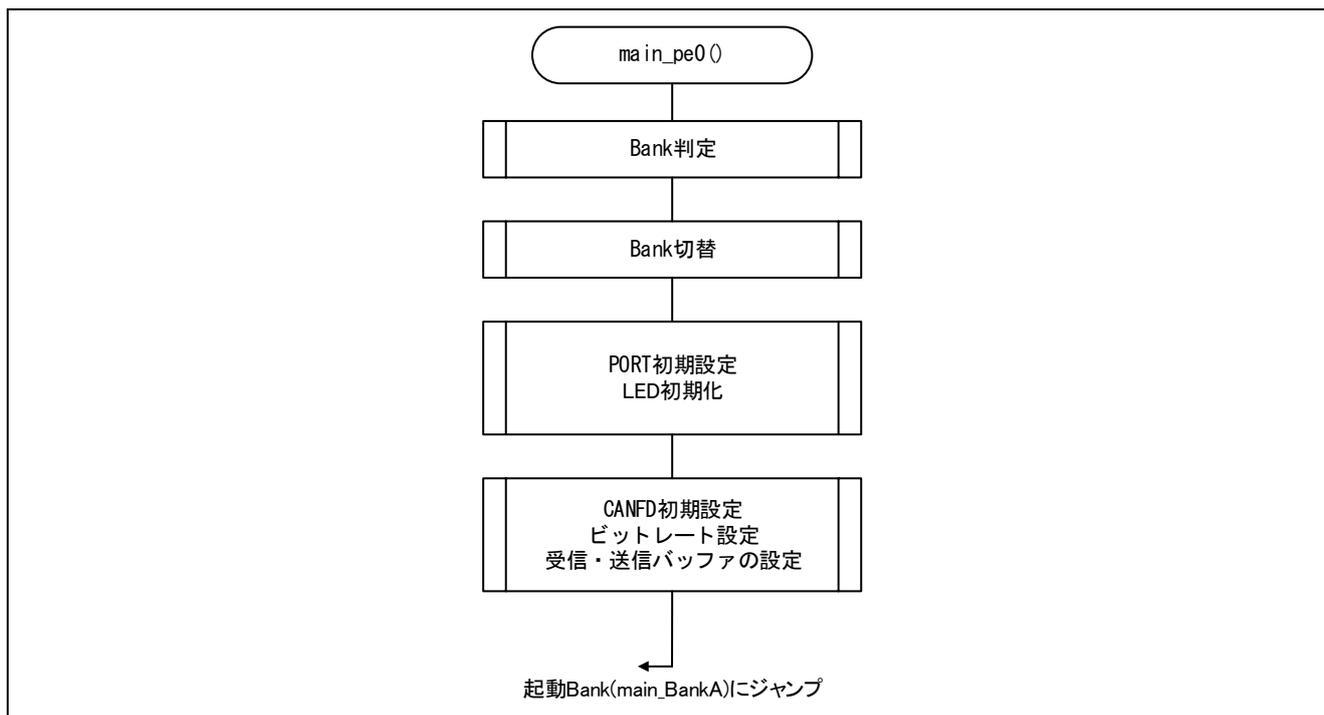


図 3-2 main_pe0()関数」のフローチャート
「動作手順①」

図 3-3に「Bank 切替関数」のフローチャートを示します。

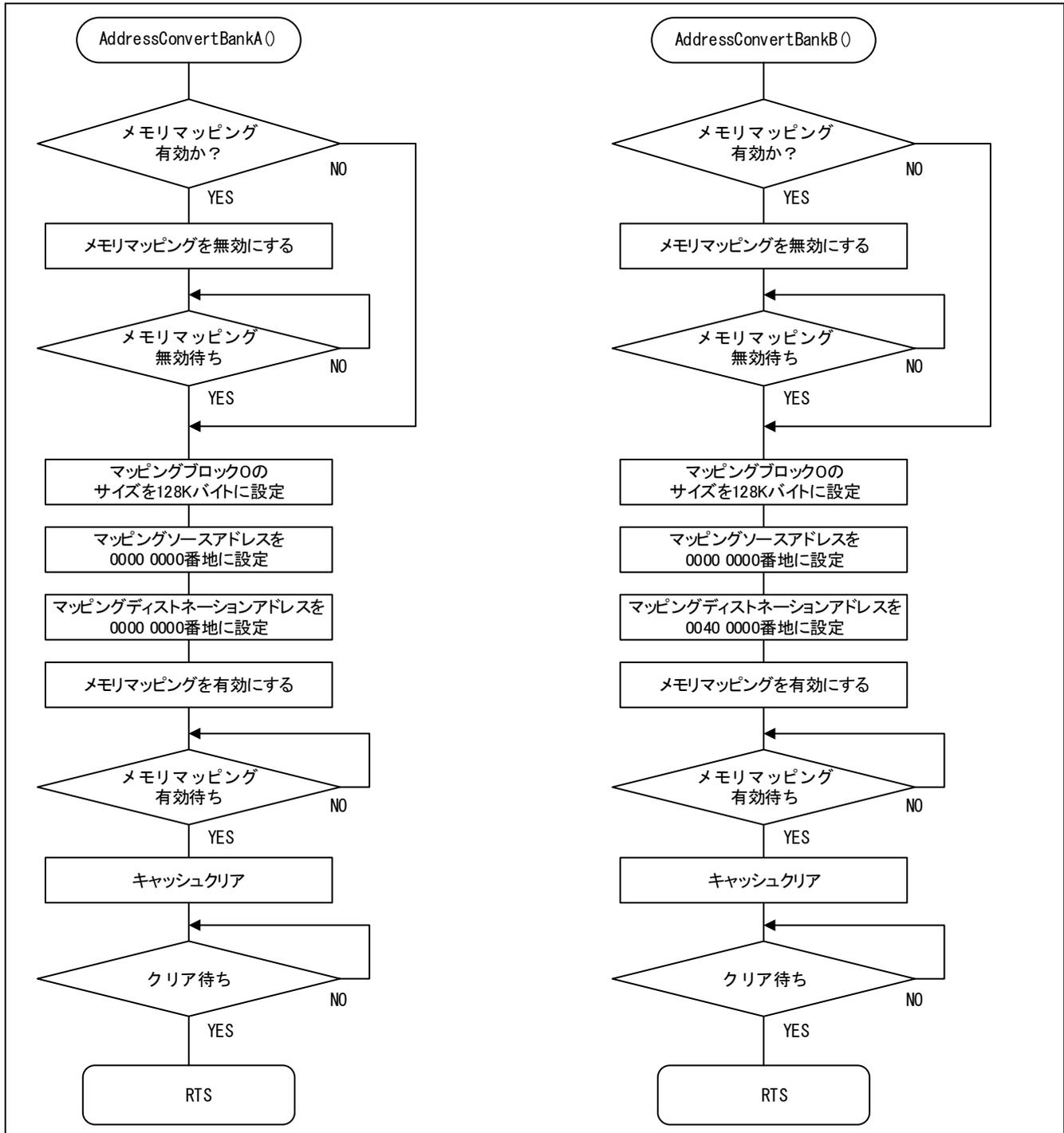


図 3-3 「Bank 切替関数」のフローチャート

3.1.2 ② BGO 書き込み

書き込み対象の Code Flash 領域は起動 Bank とは異なる Bank にすることで、「書き込み制御プログラム」を RAM に転送せずに Code Flash へ書き込むことができます。BankA で起動した場合は BankA の「書き込み制御プログラム」で、BankB の Code Flash 領域を書き換えます。

メインルーチンでは常に LED を点滅するユーザプログラムを実行しており、書き換え開始要求コマンドを受信後に書き換え制御プログラムを並行して OTA を実行します。

図 3-4に Code Flash 消去/書き込み開始動作を示します。

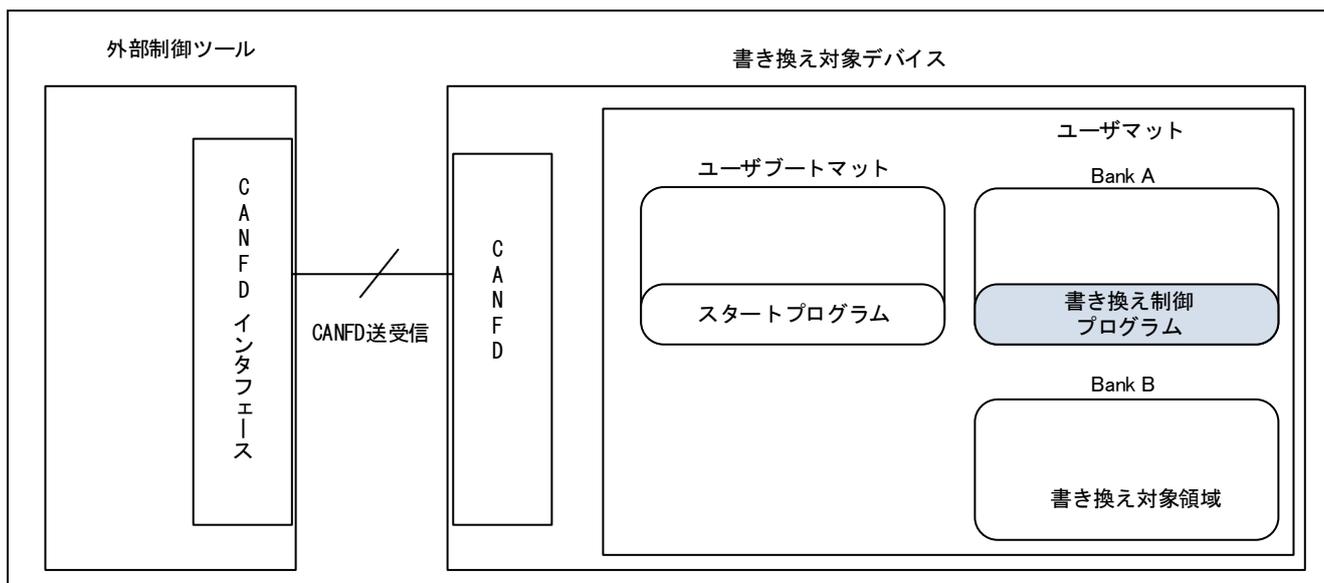


図 3-4 Code Flash 消去/書き込み開始動作

関数説明

表 3-2 「main_BankA ()関数」

関数名	概要
main_BankA ()	LED の点滅を繰り返し、書き換え開始コマンドを受信後、「書き換え制御プログラム」が実行されます。

図 3-5に「main_BankA ()関数」のフローチャートを示します。

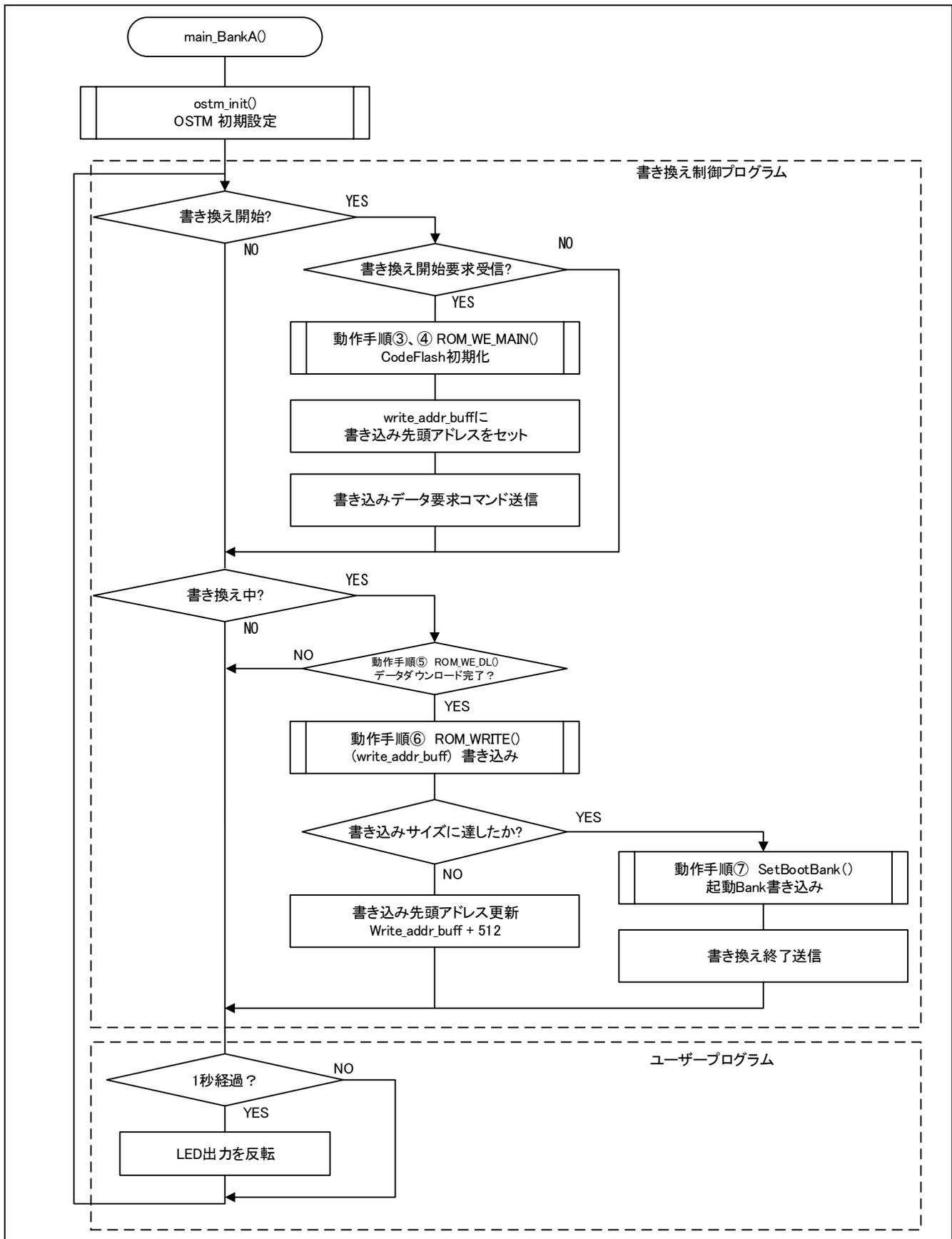


図 3-5 「main_BankA ()関数」のフローチャート

「動作手順②」

関数説明

表 3-3 「Code Flash_WE_MAIN()関数」

関数名	概要
Code Flash_WE_MAIN()	ID 認証、Code Flash 消去、Code Flash 書き込みデータダウンロード、Code Flash 書き込み、の各種関数コールを実施します。

図 3-6に「Code Flash_WE_MAIN()関数」のフローチャートを示します。

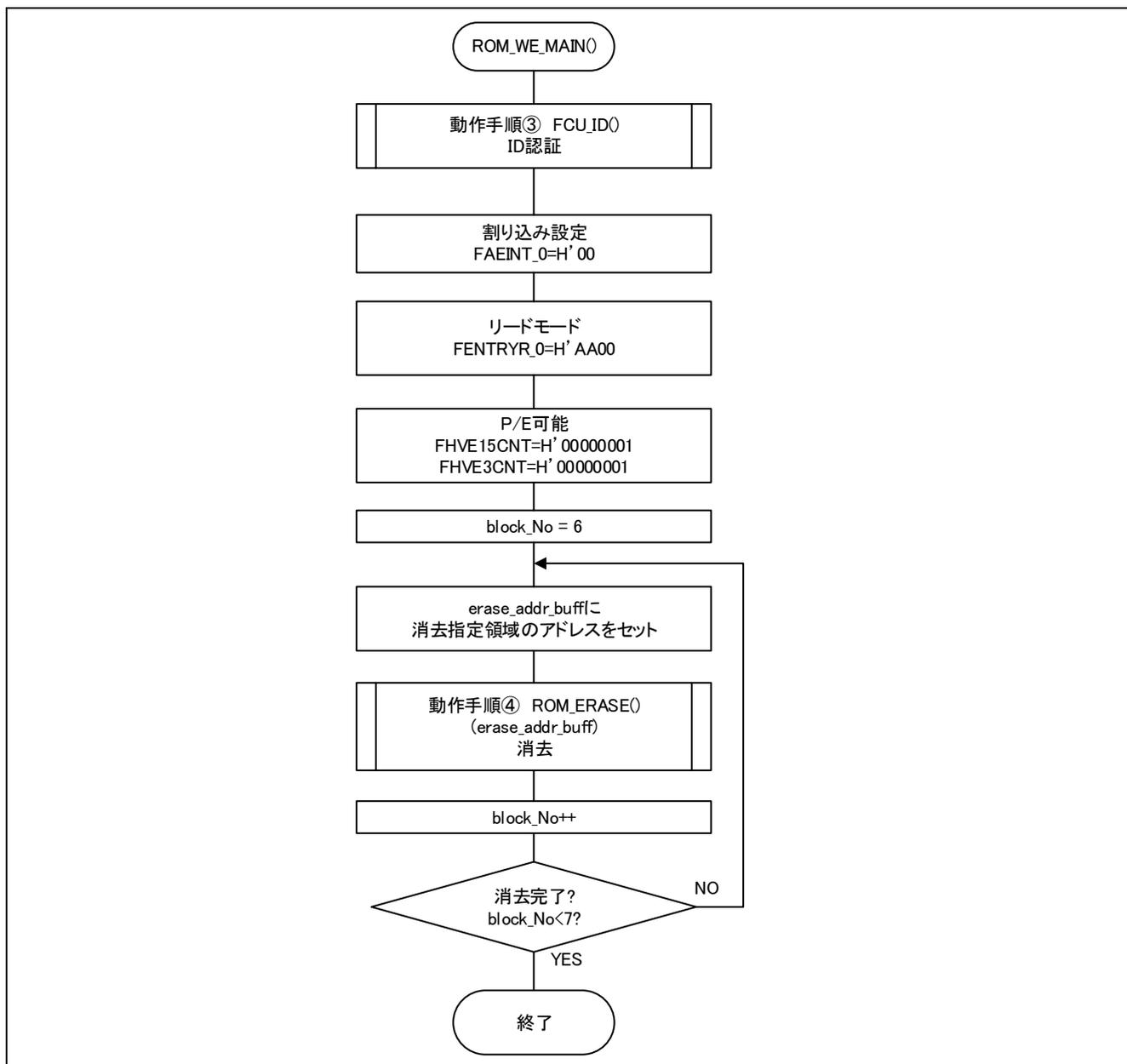


図 3-6 「Code Flash_WE_MAIN()関数」のフローチャート
「動作手順③、④を含む」

3.1.3 ③ ID 認証

「書き換え制御プログラム」の「ID 認証関数」を実行。Flash メモリの特殊領域にあらかじめ設定した 256 ビットの ID と RHSIFIDIN0~7 の値を比較することで ID の認証を実行します。

本アプリケーションノートでは、ID 設定は 1 バイト目を”0”、1 バイト目以外を”F”の設定で使用しています。ID 設定を変更する際には RH850 ファミリー用 Renesas Flash Programmer または、コンフィグレーション設定コマンドを使用し設定します。

関数説明

表 3-4 「FCU_ID()関数」

関数名	概要
FCU_ID()	Flash メモリの特殊領域に設定された ID との比較と ID 認証を実行します。

図 3-7に「FCU_ID()関数」のフローチャートを示します。

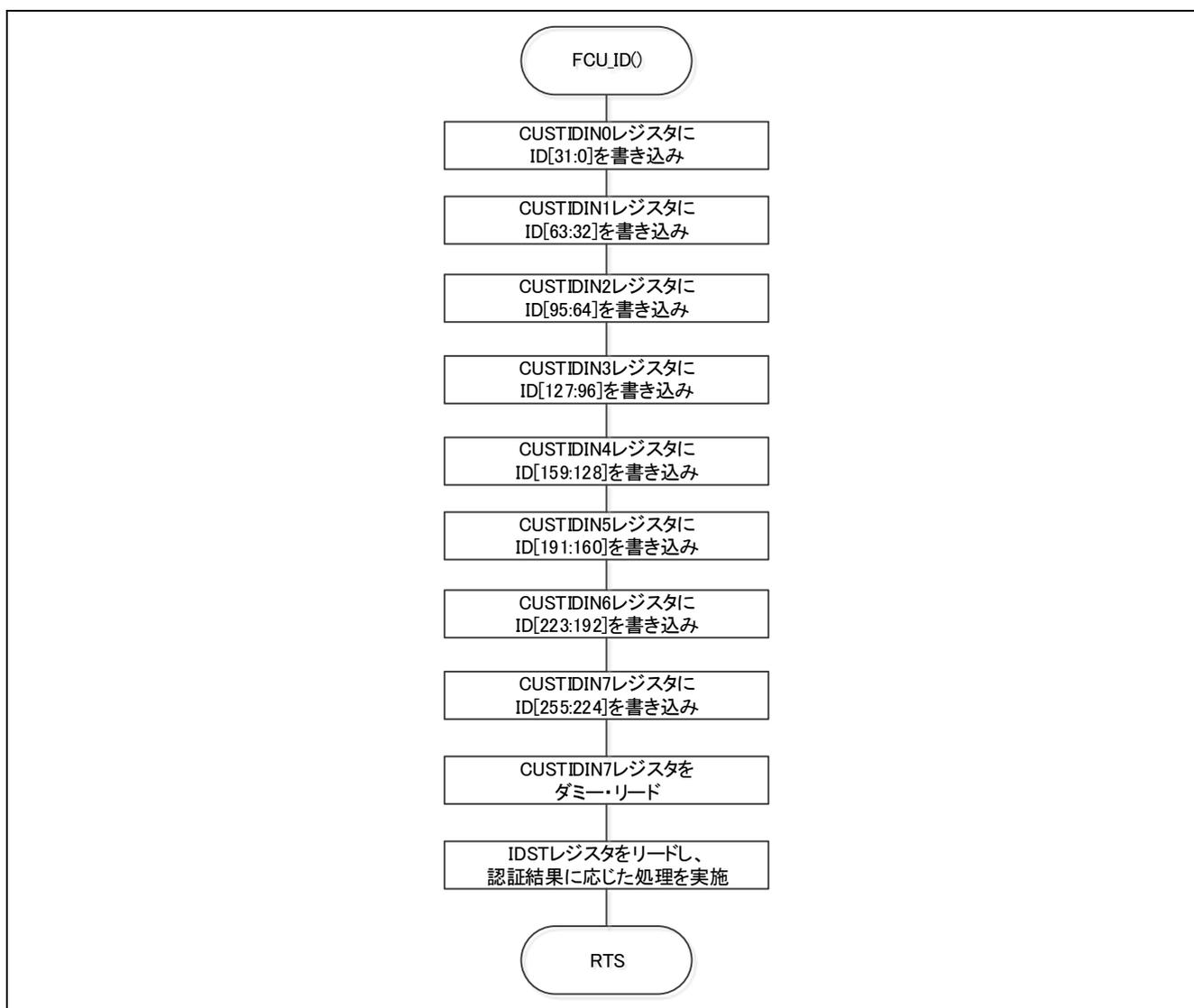


図 3-7 「FCU_ID()関数」のフローチャート

「動作手順③」

3.1.4 ④ Code Flash 消去

ID 認証後、「書き換え制御プログラム」の「Code Flash 消去関数」を実行。

FACI コマンド発行領域にブロックイレーズコマンドを発行し、Code Flash 書き換え指定領域を消去します。

関数説明

表 3-5 「Code Flash_ERASE()関数」

関数名	概要
Code Flash_ERASE()	Code Flash 書き換え指定領域を消去します。

3.1.5 ⑤ Code Flash 書き込みデータダウンロード

「Code Flash 書き込みデータダウンロード関数」を実行し、書き込みデータ要求コマンドを送信します。書き込みデータ要求コマンドを受信した外部デバイスは、書き込みデータダウンロードコマンドで書き込みデータ 512Byte をマイコンへ送信します。「Code Flash 書き込みデータダウンロード関数」では受信した書き込みデータを RAM へ保存します。

関数説明

表 3-6 「Code Flash_WE_DL()関数」

関数名	概要
Code Flash_WE_DL()	外部デバイスから書き込みデータをダウンロード

図 3-10に「Code Flash_WE_DL()関数」のフローチャートを示します。

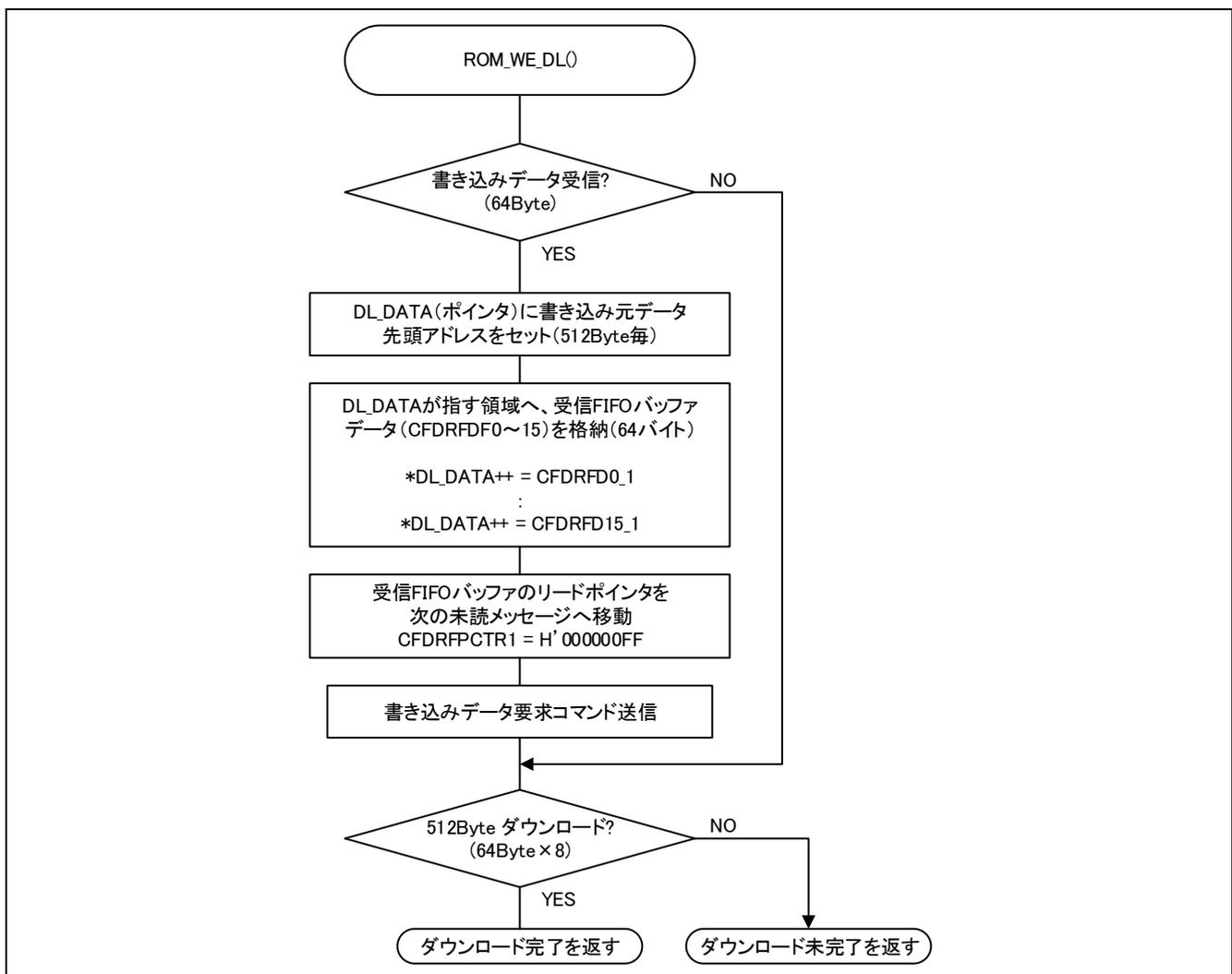


図 3-10 「Code Flash_WE_DL()関数」のフローチャート
「動作手順⑤」

3.1.6 ⑥ Code Flash 書き込み

CAN 通信で外部デバイスから受信した書き込みデータは、「Code Flash 書き込み関数」を使用して Code Flash に書き込みます。

FACI コマンド発行領域にプログラム・コマンドを発行し、Code Flash 書き換え指定領域に書き込みます。書き込みサイズ（96K バイト）に達した場合、Flash 書き換えを終了します。

関数説明

表 3-7 「Code Flash_WRITE()関数」

関数名	概要
Code Flash_WRITE()	Code Flash 書き換え指定領域に書き込みます(512Byte 単位)

図 3-11に「Code Flash_WRITE()関数」のフローチャートを示します。

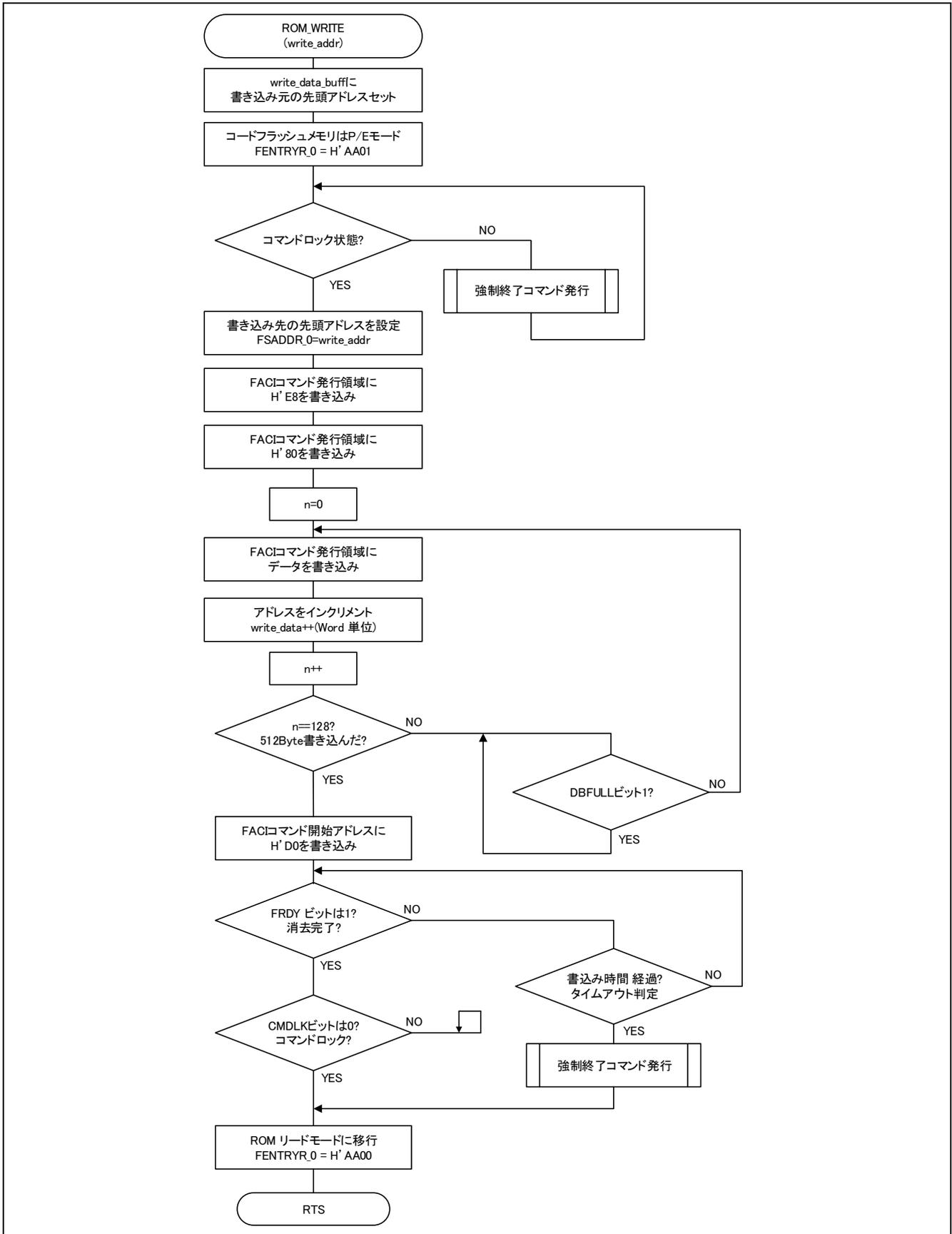


図 3-11 「Code Flash_WRITE()関数」のフローチャート

「動作手順⑥」

3.1.7 ⑦ 起動 Bank 情報書き込み関数

書き換え終了後、次回の起動 Bank の情報を Data Flash に書き込みます。

関数説明

表 3-8 「SetBootBank ()関数」

関数名	概要
SetBootBank ()	次回の起動 Bank 情報を判定し、Data Flash に書き込みます。

図 3-12に「SetBootBank ()関数」のフローチャートを示します。

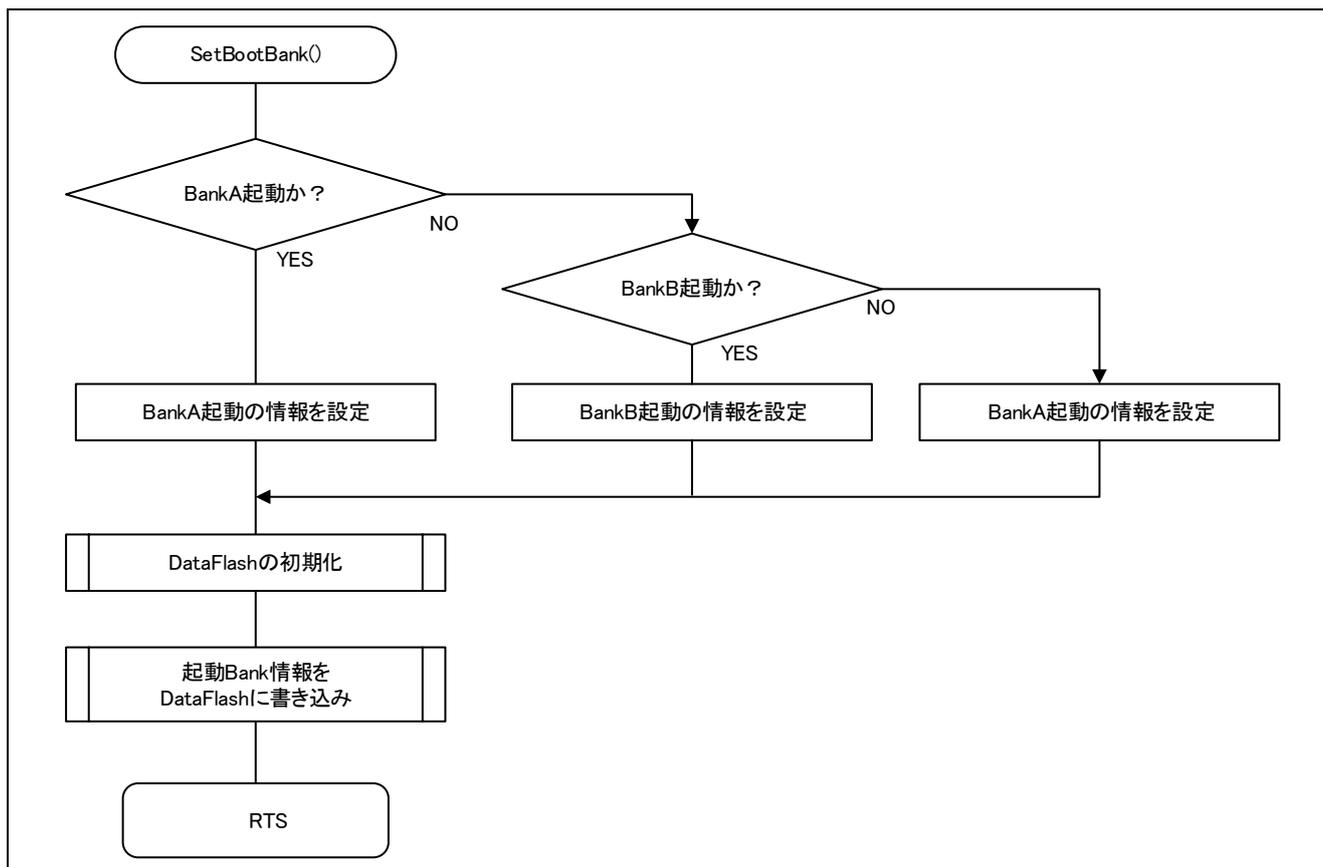


図 3-12 「SetBootBank ()関数」のフローチャート

「動作手順⑦」

4. メモリ割り付け

4.1 アドレスマップ

4.1.1 アドレス配置図

図 4-1にアドレス配置図を示します。

アドレス	セクション	
H' 0000_0000	RESET	ROM(ユーザマツト)
H' 0000_0400	EIINTTBL	
	.rodata	
	.rosdata	
	.ROM.data	
	PE0.rodata	
	PE0.text	
	.secinfo	
	.syscall	
H' 0001_4000	.Bank_textA	
	.Bank_text	
	.Bank_rosdata	
	.Bank_rodata	
H' 0040_0000	PE1.rodata	
H' 0800_0000	UBRESET	
	.text	
H' FDA0_0000	PE1.bss	RAM
	PE1.stack.bss	
H' FDC0_0000	PE0.bss	
	PE0.stack.bss	
	.data	
	.bss	
H' FE00_0000	CRAM0.bss	
H' FE08_0000	CRAM1.bss	
H' FE0A_0000	RRAM.bss	
アドレス	FCU,FACI	
H' FFA2_0000	FACIコマンド発行領域	

図 4-1 アドレス配置図

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com/>

お問い合わせ先

<http://japan.renesas.com/contact/>

すべての商標および登録商標は、それぞれの所有者に帰属します。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。

外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレス（予約領域）のアクセス禁止

【注意】リザーブアドレス（予約領域）のアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違くと、内部Code Flash、レイアウトパターンの相違などにより、電

气的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置等
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を生じさせるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じて、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の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合せください。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、適用される環境関連法令を十分調査のうえ、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、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/contact/>