

SH7268/SH7269 グループ

E10A-USB フラッシュメモリダウンロード機能の応用例 (NOR フラッシュメモリへのダウンロード)

R01AN1176JJ0100
Rev. 1.00
2012.07.18

要旨

E10A-USB エミュレータにはフラッシュメモリにロードモジュールをダウンロードする機能があります。本機能はフラッシュメモリにアクセスするために、ダウンロードプログラム（以下、FMTOOL と称します）を使用します。

本アプリケーションノートは、この FMTOOL を応用した NOR 型フラッシュメモリ（JEDEC 標準型コマンド互換方式）へのダウンロード方法について説明します。

対象デバイス

SH7268/SH7269 グループ

以下、総称して「SH7269」として説明します。

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

目次

1. 仕様	4
2. 動作確認条件	5
3. 関連アプリケーションノート	5
4. ハードウェア説明	6
4.1 使用端子一覧	6
4.2 参考回路	7
5. ソフトウェア説明	8
5.1 動作概要	8
5.1.1 バッチファイル	8
5.1.2 イレーズモジュール	8
5.1.3 ライトモジュール	9
5.2 ファイル構成	10
5.3 定数一覧	11
5.4 変数一覧	14
5.5 関数一覧	14
5.6 関数仕様	15
5.7 フローチャート	18
5.7.1 バッチファイル	18
5.7.2 イレーズモジュール	19
5.7.3 ライトモジュール	20
6. 応用例	21
6.1 ユーザプログラムダウンロード手順	21
6.1.1 ダウンロード環境の準備	21
6.1.2 バッチファイルの登録	22
6.1.3 Configurationダイアログボックスの設定	23
6.1.4 ダウンロードモジュールの追加	24
6.1.5 ユーザプログラムのダウンロード	24
6.2 FMTOOLのカスタマイズ	25
6.2.1 サンプルコードに対応するデバイス仕様	25
6.2.2 カスタマイズで対応可能なフラッシュメモリのブート型	26
6.2.3 カスタマイズの内容	27
6.2.4 ユニフォーム型のセクタサイズやセクタ数をカスタマイズする方法	29
6.2.5 フラッシュメモリのブート型をボトムブート型にカスタマイズする方法	30
6.2.6 フラッシュメモリのブート型をトップブート型にカスタマイズする方法	31

6.2.7	フラッシュメモリのブート型をデュアルブート型にカスタマイズする方法.....	32
6.2.8	フラッシュメモリのコマンドの確認	33
7.	サンプルコード	34
8.	参考ドキュメント.....	34

1. 仕様

外部アドレス空間 (CS0 空間) に配置したロードモジュールを、フラッシュメモリにダウンロードします。ダウンロードには FMTOOL を使用します。FMTOOL は、CS0 空間に接続したフラッシュメモリ (16 ビットバス接続) をアクセスします。

表 1.1 に使用するマイコンの資源と用途を、図 1.1 に FMTOOL を使用したダウンロード処理の流れを示します。

表 1.1 使用するマイコンの資源と用途

周辺機能	用途
バーステートコントローラ (CS0)	フラッシュメモリへのダウンロード
高速内蔵 RAM	ワークメモリ
H-UDI	E10A-USB エミュレータの接続

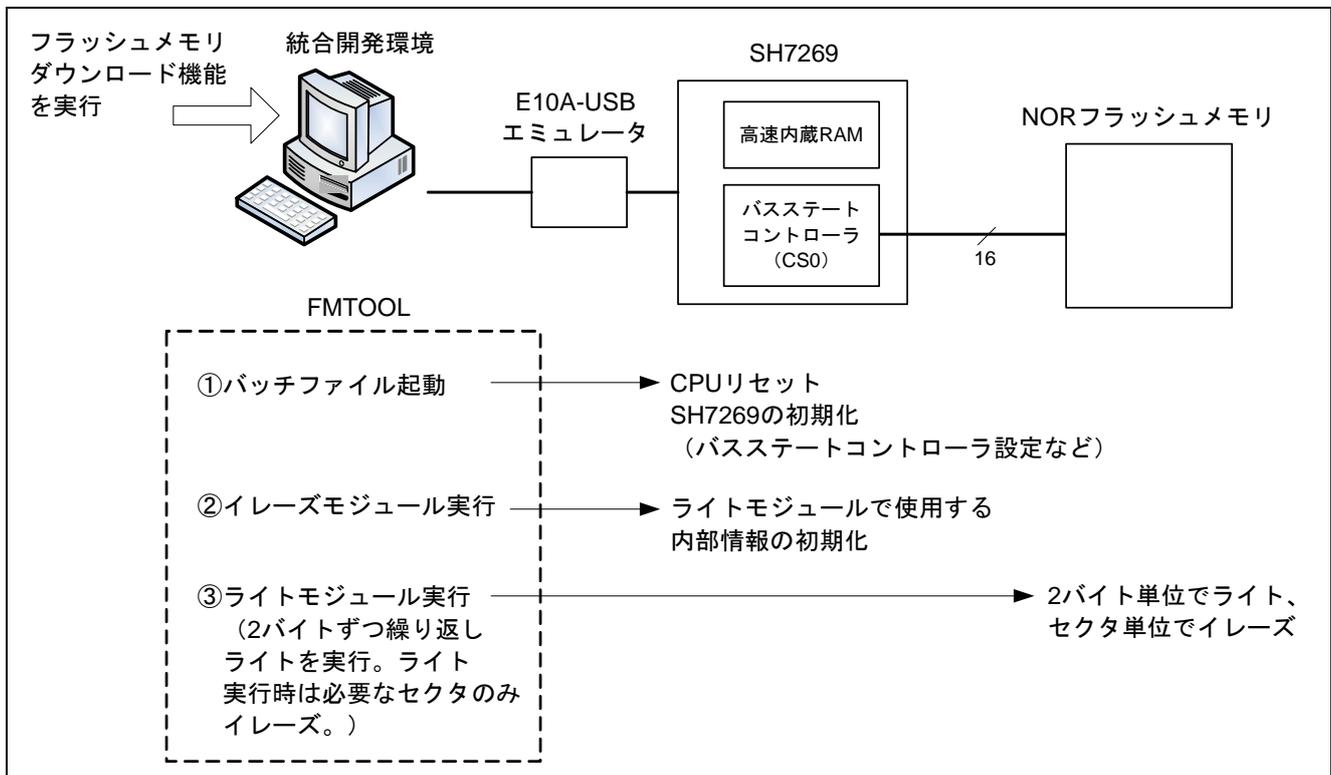


図 1.1 FMTOOL を使用したダウンロード処理の流れ

2. 動作確認条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表 2.1 動作確認条件

項目	内容
使用マイコン	SH7269
使用デバイス	フラッシュメモリ (16 ビットバスで接続) メーカー : Spansion 社 型名 : S29GL256P90TFIR1
動作周波数	CPU 内部クロック (I ϕ) : 266.67MHz 内部クロック (B ϕ) : 133.33MHz 周辺クロック 1 (P1 ϕ) : 66.67MHz 周辺クロック 0 (P0 ϕ) : 33.33MHz
動作電圧	電源電圧 (I/O) : 3.3V 電源電圧 (内部) : 1.25V
統合開発環境	ルネサス エレクトロニクス製 High-performance Embedded Workshop Ver.4.07.00
C コンパイラ	ルネサス エレクトロニクス製 SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.03 Release02 コンパイラオプション -cpu=sh2afpu -fpu=single -object="\$\$(CONFIGDIR)\\$(FILELEAF).obj" -debug -gbr=auto -chgincpath -errorpath -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1 -nologo (統合開発環境のデフォルト設定を使用しています)
使用ボード	R0K572690C000BR

3. 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。併せて参照してください。

- E10A-USB エミュレータ用 フラッシュメモリ ダウンロードプログラム アプリケーションノート (RJJ10J0957)

4. ハードウェア説明

4.1 使用端子一覧

表 4.1に使用端子と機能を示します。

表 4.1 使用端子と機能

端子名	入出力	内容
A24~A1	出力	アドレスバス
D15~D0	入出力	データバス
CS0#	出力	フラッシュメモリへのデバイス選択信号出力
RD#	出力	フラッシュメモリへのリード制御信号出力
WE0#	出力	フラッシュメモリへのライトイネーブル制御信号出力
MD_BOOT1、MD_BOOT0	入力	ブートモードの選択
AUDCK	出力	E10A-USB エミュレータ (38pin) へのクロック出力
AUDATA3~AUDATA0	出力	E10A-USB エミュレータ (38pin) へのアドレス出力
AUDSYNC#	出力	E10A-USB エミュレータ (38pin) への同期信号出力
TCK	入力	E10A-USB エミュレータからのクロック入力
TMS	入力	E10A-USB エミュレータからのモード選択
TRST#	入力	E10A-USB エミュレータからのリセット入力
TDI	入力	E10A-USB エミュレータからのデータ入力
TDO	出力	E10A-USB エミュレータへのデータ出力
ASEBRKAK#/ASEBRK#	入出力	ブレーク要求と応答
RES#	入力	システムリセット信号
ASEMD#	入力	ASE モードの選択

【注】 #は負論理 (またはアクティブロー) を示す記号です。

4.2 参考回路

図 4.1にFMTOOL使用時にNORフラッシュメモリにダウンロードする場合の接続例を示します。

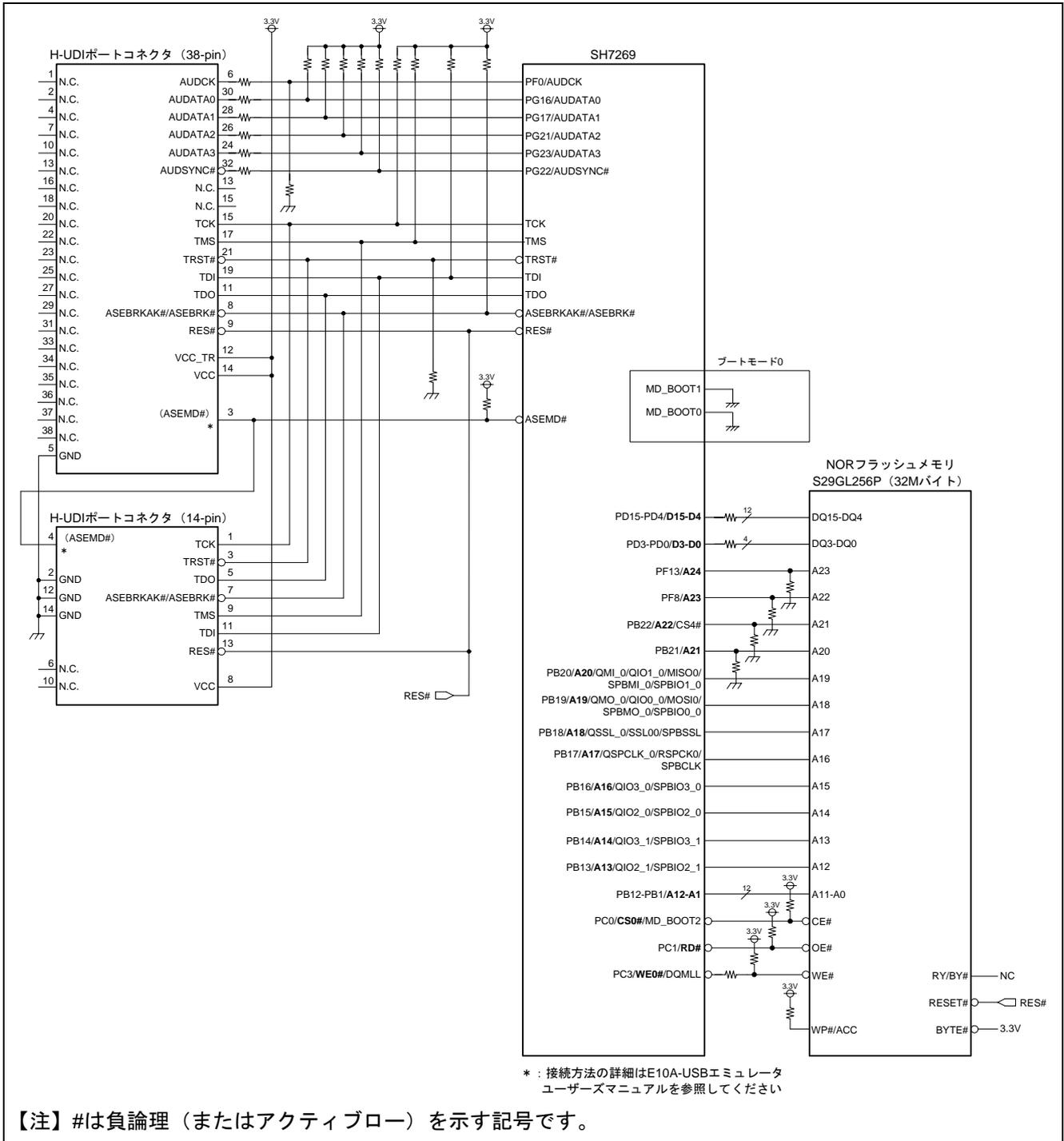


図 4.1 FMTOOL 使用時に NOR フラッシュメモリにダウンロードする場合の接続例

5. ソフトウェア説明

5.1 動作概要

FMTOOL は、イレーズモジュールとライトモジュールの2つのプログラムから構成されます。E10A-USB エミュレータは、この2つのプログラムを使用して、フラッシュメモリにプログラムデータを書き込みます。イレーズモジュールおよびライトモジュールの詳細仕様は、E10A-USB エミュレータ ユーザーズマニュアル「6.22 フラッシュメモリへのダウンロード機能」を参照してください。

サンプルコードでは、ダウンロード処理の高速化のために、チップイレーズは使用せず、必要なセクタのみイレーズします。なお、セクタのイレーズ処理は、ライトモジュールで行っています。

5.1.1 バッチファイル

SH7269 を初期化するために、リセットコマンドを実行します。コマンドの実行には、ロードモジュールのダウンロード前に起動されるバッチファイルを使用します。バッチファイル、およびリセットコマンドについては、統合開発環境のマニュアルを参照してください。

5.1.2 イレーズモジュール

ロードモジュールのダウンロードを開始すると、FMTOOL が SH7269 の高速内蔵 RAM 上に転送されます。転送後に、イレーズモジュールは一度のみ実行されます。

イレーズモジュールの本来の動作は、フラッシュメモリのチップイレーズ処理を実装することを想定していますが、サンプルコードではライトモジュールで使用する内部情報の初期化を行います。

5.1.3 ライトモジュール

ライトモジュールは、ロードモジュールのダウンロード時に、繰り返し実行されます。その際、アクセスサイズ単位に分割したプログラムデータが、引数として渡されます。

図 5.1にライトモジュールの概要を示します。ライトモジュールはSH7269 の高速内蔵RAM上で実行され、プログラムデータのアドレスとワードデータを引数として受け取り、フラッシュメモリに書き込みを行います。また、書き込み先アドレスが未消去セクタの場合は、セクタイレース処理を行います。

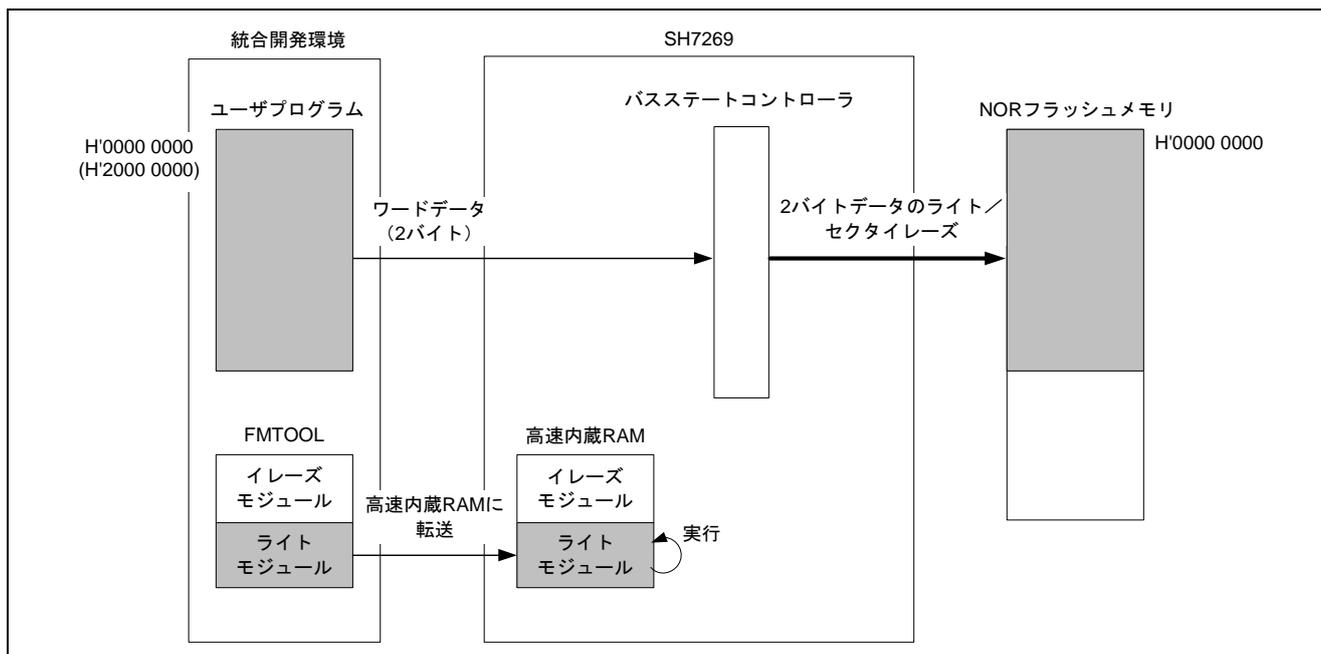


図 5.1 ライトモジュールの概要

5.2 ファイル構成

表 5.1にサンプルコードで使用するファイルを示します。なお、統合開発環境で自動生成されるファイルは除きます。

表 5.1 ファイル構成

ファイル名	概要	備考
fmttool_entry.src	FMTOOL のエントリモジュール	イレーズモジュールと ライトモジュールのエントリ
fmttool_main.c	FMTOOL のメインモジュール	
flash_cfg.h	フラッシュメモリの定義	Spansion 社製フラッシュメモリ S29GL256P の定義ファイル 上記のフラッシュメモリとデバイ ス仕様が異なる場合は、カスタ マイズが必要
sh7269_fmttool.hdc	バッチファイル	統合開発環境に登録

5.3 定数一覧

表 5.2にサンプルコードで使用する定数を示します。

表 5.2 サンプルコードで使用する定数一覧 (1/3)

定数名	設定値	内容	変更の 要否
FM_UNIFORM	0	ユニフォーム型	
FM_TOP_BOOT	1	トップブート型	
FM_BOTTOM_BOOT	2	ボトムブート型	
FM_DUAL_BOOT	3	デュアルブート型	
FM_BOOT_TYPE	FM_UNIFORM	セクタ構成のタイプ	○
FM_B_BOOT_SECTOR_START	0x00000000UL	ボトムブートセクタの先頭アドレス (ワード単位)	○
FM_B_BOOT_SECTOR_SIZE	0x00000000UL	ボトムブートセクタの、1セクタのサイ ズ (ワード単位)	○
FM_B_BOOT_SECTOR_NUM	0	ボトムブートセクタの数	○
FM_NORMAL_SECTOR_START	0x00000000UL	ブートセクタ以外の領域の先頭アドレス (ワード単位)	○
FM_NORMAL_SECTOR_SIZE	0x00010000UL	ブートセクタ以外の領域の、1セクタの サイズ (ワード単位)	○
FM_NORMAL_SECTOR_NUM	256	ブートセクタ以外の領域のセクタの数	○
FM_T_BOOT_SECTOR_START	0x00000000UL	トップブートセクタの先頭アドレス (ワード単位)	○
FM_T_BOOT_SECTOR_SIZE	0x00000000UL	トップブートセクタの、1セクタのサイ ズ (ワード単位)	○
FM_T_BOOT_SECTOR_NUM	0	トップブートセクタの数	○
FM_END_ADDRESS	0x00FFFFFFUL	フラッシュメモリの最終アドレス (ワー ド単位)	○
FM_CMD_RESET	0x00F0	リセットコマンド	○
FM_CMD_S_ERASE_ADDR_1	0x0555	セクタイレースコマンド (第一サイク ル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_S_ERASE_ADDR_2	0x02AA	セクタイレースコマンド (第二サイク ル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_S_ERASE_ADDR_3	0x0555	セクタイレースコマンド (第三サイク ル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_S_ERASE_ADDR_4	0x0555	セクタイレースコマンド (第四サイク ル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_S_ERASE_ADDR_5	0x02AA	セクタイレースコマンド (第五サイク ル) のアドレス (ワードコマンド用アドレス)	○

【注】"変更の要否"欄で"○"で示した項目は、使用するフラッシュメモリのデバイスに依存します。「6.2 FMTOOLのカスタマイズ」にカスタマイズの方法を記載しています。

表 5.3 サンプルコードで使用する定数一覧 (2/3)

定数名	設定値	内容	変更の 要否
FM_CMD_S_ERASE_DATA_1	0x00AA	セクタイレースコマンド (第一サイクル) のデータ	○
FM_CMD_S_ERASE_DATA_2	0x0055	セクタイレースコマンド (第二サイクル) のデータ	○
FM_CMD_S_ERASE_DATA_3	0x0080	セクタイレースコマンド (第三サイクル) のデータ	○
FM_CMD_S_ERASE_DATA_4	0x00AA	セクタイレースコマンド (第四サイクル) のデータ	○
FM_CMD_S_ERASE_DATA_5	0x0055	セクタイレースコマンド (第五サイクル) のデータ	○
FM_CMD_SECTOR_ERASE	0x0030	セクタイレースコマンド (第六サイクル) のデータ	○
FM_CMD_PROGRAM_ADDR_1	0x0555	シングルワードプログラムコマンド (第一サイクル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_PROGRAM_ADDR_2	0x02AA	シングルワードプログラムコマンド (第二サイクル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_PROGRAM_ADDR_3	0x0555	シングルワードプログラムコマンド (第三サイクル) のアドレス (ワードコマンド用アドレス)	○
FM_CMD_PROGRAM_DATA_1	0x00AA	シングルワードプログラムコマンド (第一サイクル) のデータ	○
FM_CMD_PROGRAM_DATA_2	0x0055	シングルワードプログラムコマンド (第二サイクル) のデータ	○
FM_CMD_PROGRAM_DATA_3	0x00A0	シングルワードプログラムコマンド (第三サイクル) のデータ	○
FM_CHK_DQ5	0x0020	タイミングリミット経過 (DQ5) ビットのマスク情報	
FM_CHK_DQ6	0x0040	トグルビット (DQ6) のマスク情報	
FM_CHK_DQ7	0x0080	DATA#ポーリング (DQ7) ビットのマスク情報	
FM_CS0_NON_CACHE_START	0x20000000UL	SH7269 の CS0 空間 (キャッシュ無効空間) の先頭アドレス	
FM_TYPE_BYTE	0x4220	ライトモジュールの R5 パラメータ (バイトサイズ単位でフラッシュメモリ をアクセス)	
FM_TYPE_WORD	0x5720	ライトモジュールの R5 パラメータ (ワードサイズ単位でフラッシュメモリ をアクセス)	
FM_TYPE_LONG	0x4C20	ライトモジュールの R5 パラメータ (ロングサイズ単位でフラッシュメモリ をアクセス)	

【注】 "変更の要否"欄で"○"で示した項目は、使用するフラッシュメモリのデバイスに依存します。「6.2 FMTOOLのカスタマイズ」にカスタマイズの方法を記載しています。

表 5.4 サンプルコードで使用する定数一覧 (3/3)

定数名	設定値	内容	変更の 要否
FM_TOOL_OK	0	リターン値：エラーなし	
FM_TOOL_E_ERASE	-1	リターン値：イレーズエラー	
FM_TOOL_E_WRITE	-2	リターン値：ライトエラー	
FM_TOOL_E_VERIFY	0x4254	リターン値：ベリファイエラー	
FM_TOOL_E_ARGUMENT	-16	リターン値：引数エラー	
FM_WRITE_TOP	H'FFF80000	ライトモジュールの先頭アドレス	
FM_ERASE_TOP	H'FFF81000	イレーズモジュールの先頭アドレス	
FM_STACK_TOP	H'FFF90000	FMTOOL が使用するスタックポインタ	
FM_SECTOR_NUM	セクタ数の合計	全セクタの数 (FM_T_BOOT_SECTOR_NUM + FM_B_BOOT_SECTOR_NUM + FM_NORMAL_SECTOR_NUM)	

【注】 "変更の要否"欄で"○"で示した項目は、使用するフラッシュメモリのデバイスに依存します。「6.2 FMTOOLのカスタマイズ」にカスタマイズの方法を記載しています。

5.4 変数一覧

表 5.5にサンプルコードで使用するstatic型変数を示します。

表 5.5 サンプルコードで使用する static 型変数一覧

型	変数名	内容	使用する関数名
uint32_t	fmtool_pre_erase_sctno[(FM_SECTOR_NUM/32)+1]	イレース済みセクタの管理情報 (セクタ数分)	FmtoolInit、 FmtoolWrite、 FmCheckPreErase

5.5 関数一覧

表 5.6にサンプルコードで使用する関数を示します。

表 5.6 サンプルコードで使用する関数一覧

関数名	概要
_FMTOOL_ERASE	イレースモジュールのエントリ処理
_FMTOOL_WRITE	ライトモジュールのエントリ処理
FmtoolInit	イレースモジュールのメイン処理 (初期化処理)
FmtoolWrite	ライトモジュールのメイン処理 (イレースおよびライト処理)
FmResetCmd	フラッシュメモリのリセット処理
FmSectorEraseCmd	セクタイレース処理
FmWordProgramCmd	シングルワードプログラム処理
FmCheckPreErase	消去済みセクタの判定処理

5.6 関数仕様

サンプルコードの関数仕様を示します。

_FMTOOL_ERASE

概要	イレーズモジュールのエントリ処理
ヘッダ	なし
宣言	<code>_FMTOOL_ERASE:</code>
説明	イレーズモジュールのエントリ部で、H'FFF8 1000 番地に配置します。本モジュールは E10A-USB フラッシュメモリダウンロード機能によって起動されます。 本モジュールはスタックポインタの設定後、レジスタの退避を行い、FmtoolInit 関数を実行します。その後、レジスタの復帰を行い、RTS 命令により、E10A-USB エミュレータに制御を戻します。
引数	R4 レジスタ : アクセスサイズ (バイト= H'4220、ワード= H'5720、ロング= H'4C20)
リターン値	なし
備考	アセンブリ言語で記述されています。

_FMTOOL_WRITE

概要	ライトモジュールのエントリ処理
ヘッダ	なし
宣言	<code>_FMTOOL_WRITE:</code>
説明	ライトモジュールのエントリ部で、H'FFF8 0000 番地に配置します。本モジュールは E10A-USB フラッシュメモリダウンロード機能によって起動されます。 本モジュールはスタックポインタの設定後、レジスタの退避を行い、FmtoolWrite 関数を実行します。その後、レジスタの復帰を行い、RTS 命令により、E10A-USB エミュレータに制御を戻します。
引数	R4 レジスタ : ライトアドレス R5 レジスタ : アクセスサイズ (バイト= H'4220、ワード= H'5720、ロング= H'4C20) R6 レジスタ : ライトデータ
リターン値	R0 レジスタ=0 : 正常終了 R0 レジスタ=0 以外 : エラー終了
備考	アセンブリ言語で記述されています。

FmtoolInit

概要	イレーズモジュールのメイン処理 (初期化処理)
ヘッダ	"flash_cfg.h"
宣言	<code>void FmtoolInit(uint32_t access_size);</code>
説明	ライトモジュールのメイン処理で使用する変数の初期化を行います。本関数は FMTOOL のエントリポイント (<code>_FMTOOL_ERASE</code>) から実行します。
引数	第一引数 : <code>access_size</code> : アクセスサイズ ※未使用 (バイト= H'4220、ワード= H'5720、ロング= H'4C20)
リターン値	なし
備考	

FmtoolWrite

概要	ライトモジュールのメイン処理（イレーズおよびライト処理）
ヘッダ	"flash_cfg.h"
宣言	int32_t FmtoolWrite(uint32_t addr, uint32_t access_size, uint32_t data);
説明	フラッシュメモリのイレーズおよびライト処理を行います。イレーズはセクタ単位、ライトは 16 ビット単位でフラッシュメモリをアクセスします。 引数 addr で指定された書き込み先のアドレスが未消去のセクタの場合は、イレーズ処理後にライト処理を行い、消去済みのセクタの場合はライト処理のみを行います。本関数は FMTOOL のエントリポイント（_FMTOOL_WRITE）から実行します。
引数	第一引数 : addr : ライトアドレス 第二引数 : access_size : アクセスサイズ (バイト= H'4220、ワード= H'5720、ロング= H'4C20) 第三引数 : data : ライトデータ
リターン値	0 : 正常終了 -1 : エラー終了（イレーズエラー） -2 : エラー終了（ライトエラー） H'4254 : エラー終了（ベリファイエラー） -16 : エラー終了（引数エラー）
備考	アクセスサイズはワードサイズのみ対応します。 ワード以外が指定された場合は、本関数を引数エラーで終了します。

FmResetCmd

概要	フラッシュメモリのリセット処理
ヘッダ	"flash_cfg.h"
宣言	void FmResetCmd(volatile uint16_t *pResetAddr);
説明	フラッシュメモリに対してリセットコマンドを実行します。
引数	第一引数 : *pResetAddr リセット対象アドレスへのポインタ
リターン値	なし
備考	

FmSectorEraseCmd

概要	セクタイレース処理
ヘッダ	"flash_cfg.h"
宣言	int32_t FmSectorEraseCmd(volatile uint16_t *pEraseBlock);
説明	指定されたセクタに対してセクタイレースします。 イレースコマンド実行後、トグルビットのチェック、ポーリングビットのチェックを行い、イレースエラーの場合は本関数をエラー終了します。
引数	第一引数: *pEraseBlock : イレース対象セクタの先頭アドレスへのポインタ
リターン値	0: 正常終了 -1: エラー終了 (イレースエラー)
備考	

FmWordProgramCmd

概要	シングルワードプログラム処理
ヘッダ	"flash_cfg.h"
宣言	int32_t FmWordProgramCmd(volatile uint16_t *pWriteAddr, uint16_t Data);
説明	指定されたアドレスに指定されたデータをライトします。 ライトコマンド実行後、トグルビットのチェック、ポーリングビットのチェックを行い、ライトエラーの場合は本関数をエラー終了します。エラーなくライトしたときはベリファイを行い、ベリファイエラーの場合は本関数をエラー終了します。
引数	第一引数: *pWriteAddr : ライト対象アドレスへのポインタ 第二引数: Data : ライトデータ
リターン値	0: 正常終了 -2: エラー終了 (ライトエラー) H4254: エラー終了 (ベリファイエラー)
備考	

FmCheckPreErase

概要	消去済みセクタの判定処理
ヘッダ	"flash_cfg.h"
宣言	int32_t FmCheckPreErase(uint32_t sect_no);
説明	指定されたライトアドレスのセクタが消去済みかどうかを判定し、結果を返します。
引数	第一引数: sect_no : セクタ番号
リターン値	0: ライトアドレスのセクタが未消去 1: ライトアドレスのセクタが消去済み
備考	

5.7 フローチャート

バッチファイルの処理の流れ、およびサンプルコードで使用する主な関数の処理の流れを示します。

5.7.1 バッチファイル

図 5.2に、バッチファイルの処理の流れを示します。

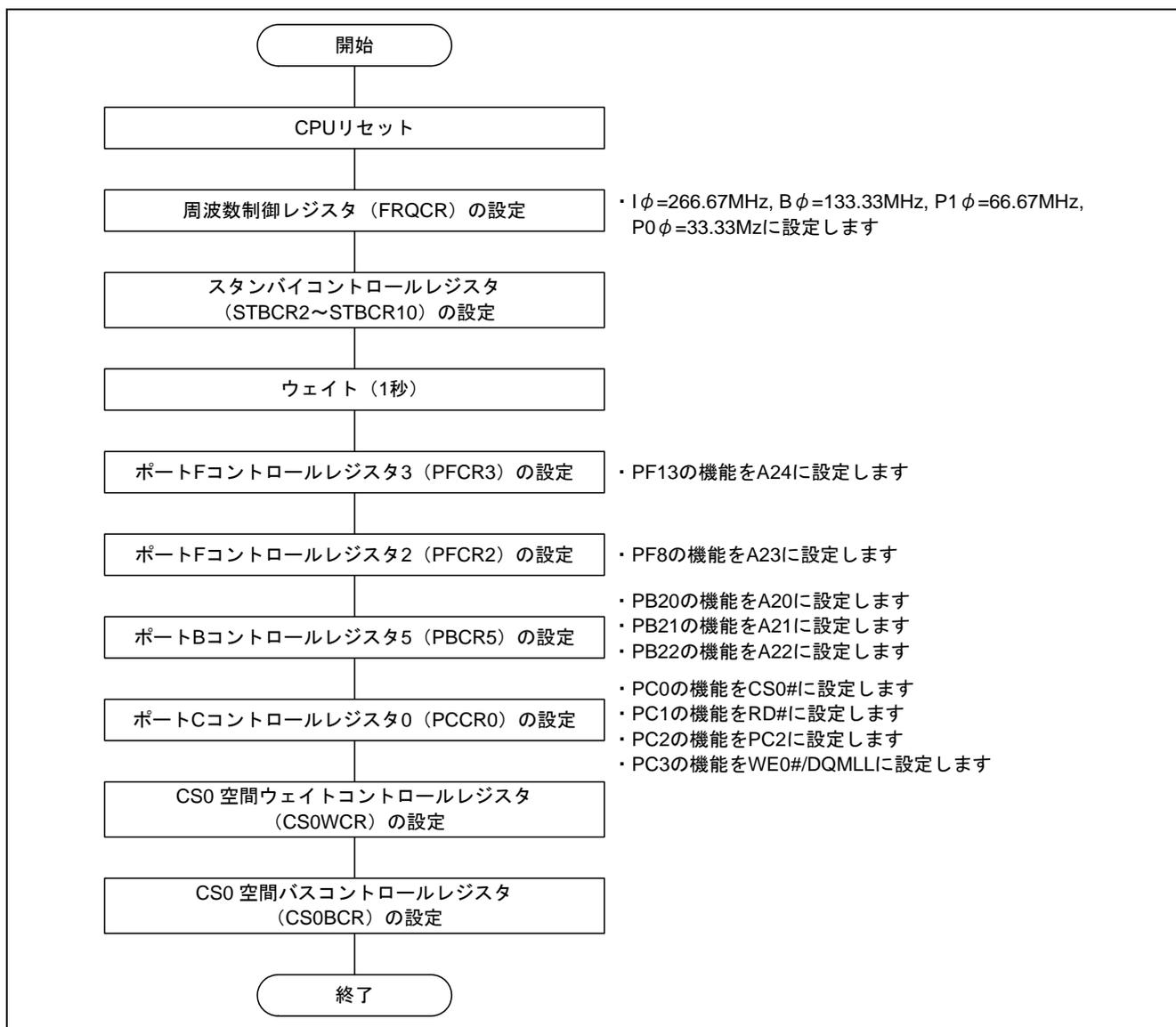


図 5.2 バッチファイルの処理の流れ

【注】 SH7269 では、A25～A21 のアドレス端子は初期状態でポート機能となっています。A25～A21 のアドレス端子を使用する場合は、バッチファイルにてアドレス端子機能に設定してください。

5.7.2 イレーズモジュール

図 5.3にイレーズモジュールのフローチャートを示します。

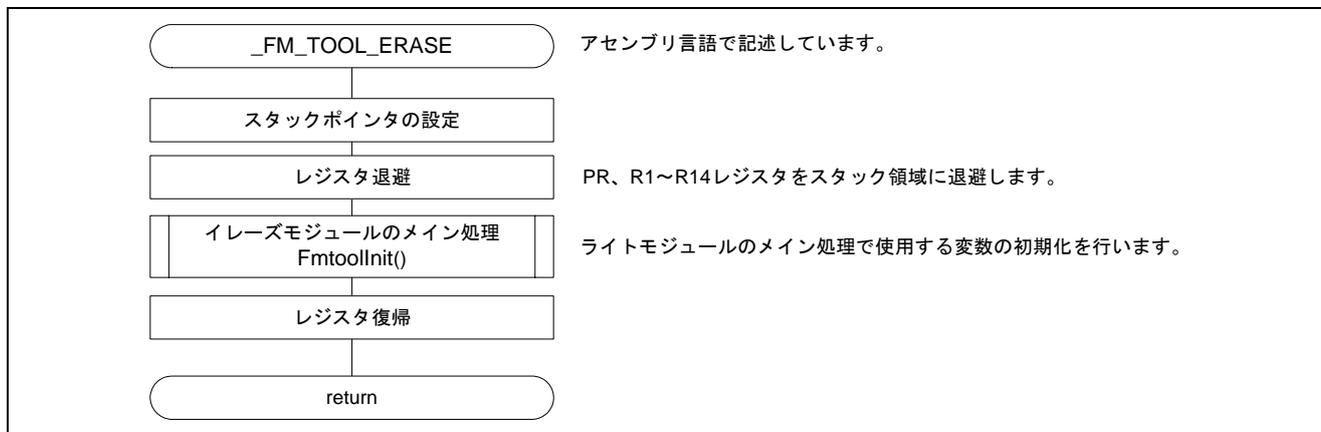
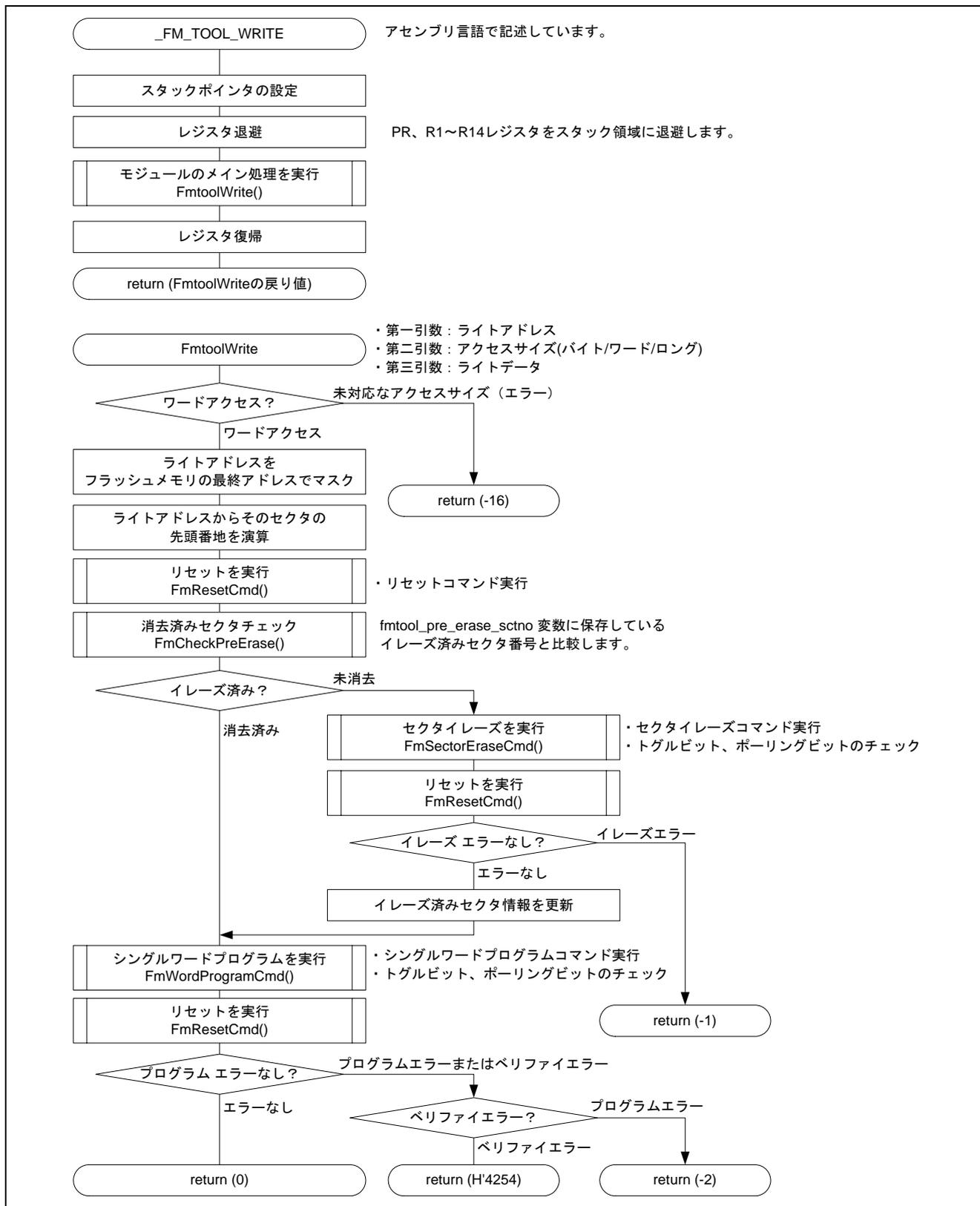


図 5.3 イレーズモジュール

5.7.3 ライトモジュール

図 5.4にライトモジュールのフローチャートを示します。



6. 応用例

6.1 ユーザプログラムダウンロード手順

作成した FMTOOL (sh7269_fmtool.mot) を使用してユーザプログラムをフラッシュメモリにダウンロードする手順を以下に示します。

6.1.1 ダウンロード環境の準備

- (1) PC に接続した E10A-USB エミュレータと、ユーザシステムを接続します。
- (2) High-performance Embedded Workshop を起動し、ユーザプログラムのワークスペースを開きます。
- (3) 図 6.1に示すCPU selectダイアログボックスが表示されます。

ご使用の CPU をドロップダウンリストボックスより選択し、OK ボタンを押してください。

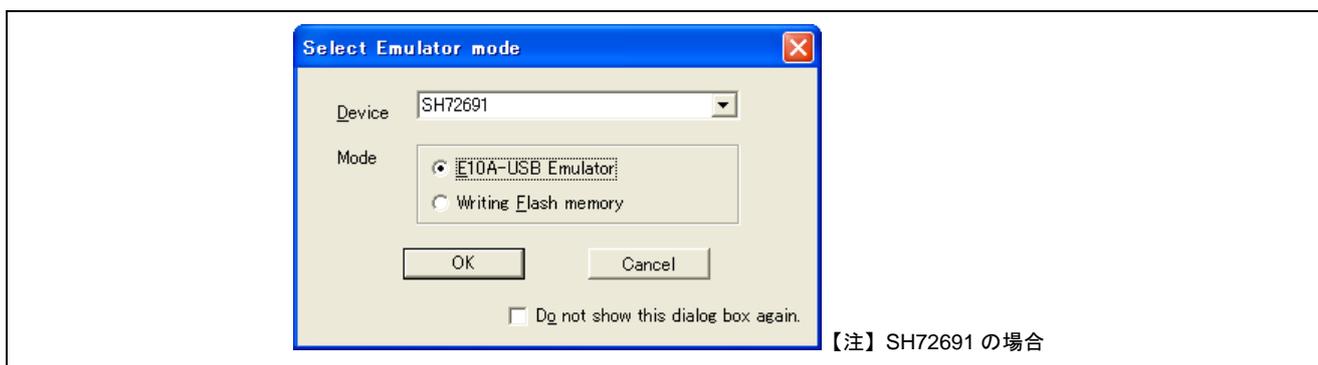


図 6.1 CPU select ダイアログボックス

- (4) Connectingダイアログボックスが表示され、エミュレータの接続を開始します。

図 6.2に示すRESET信号入力要求メッセージのダイアログボックスが表示されます。



図 6.2 RESET 信号入力要求メッセージのダイアログボックス

- (5) ユーザシステムに電源を投入します。

ユーザシステムからRESET信号を入力し、OKボタンをクリックします。

High-performance Embedded Workshop のOutputウィンドウに "Connected" と表示されたら、E10A-USBエミュレータの起動は完了です。

6.1.2 バッチファイルの登録

- (1) メニューの[デバッグ]→[デバッグの設定]を選択します。
- (2) 図 6.3に示すデバッグの設定ウィンドウが開きます。
- (3) "コマンドバッチファイル実行タイミング" のプルダウンメニューから、"Before download modules"を選択します。
- (4) "コマンドバッチファイル実行順序" の追加ボタンをクリックし、バッチファイルを追加します。
- (5) OK ボタンをクリックして登録は完了です。

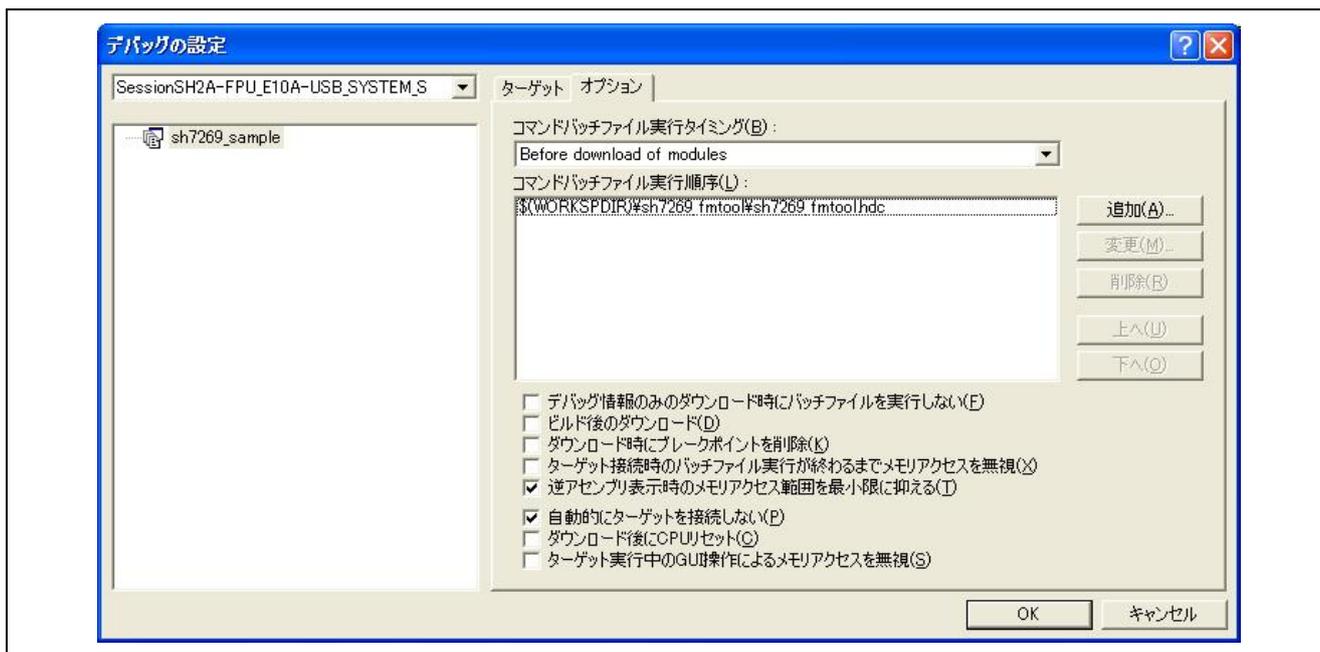


図 6.3 デバッグの設定ウィンドウ

6.1.3 Configuration ダイアログボックスの設定

(1) メニューの[基本設定]→[エミュレータ]→[システム]を選択します。

(2) 図 6.4に示すConfigurationダイアログボックス (Loading flash memoryページ) で、E10A-USBエミュレータを使用して外部フラッシュメモリにユーザプログラムをダウンロードするための設定を行います。

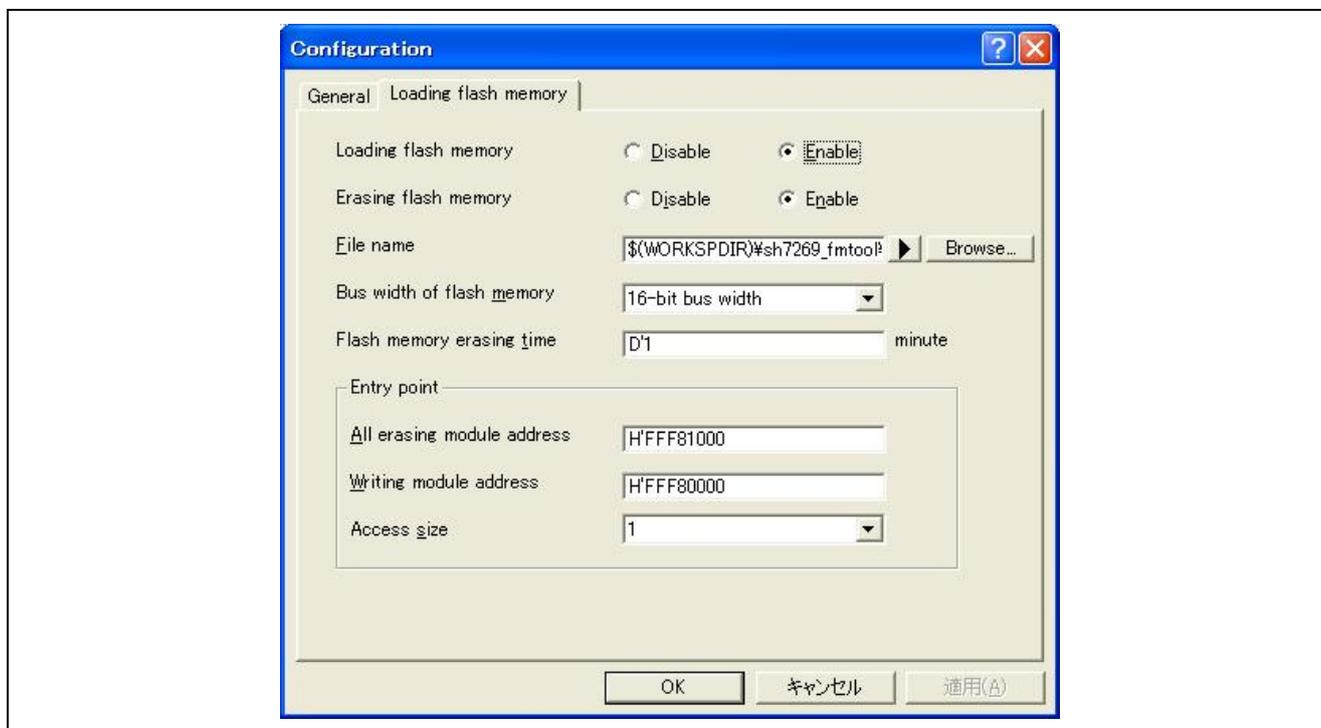


図 6.4 Configuration ダイアログボックス (Loading flash memory ページ)

各項目の設定を表 6.1に示します。設定が完了したらOKボタンをクリックして完了です。

表 6.1 Configuration ダイアログボックスの設定値

項目	設定値
Loading flash memory	Enable
Erasing flash memory	Enable
File Name	(FMTOOL を格納したディレクトリ)\sh7269_fmtool.mot
Bus width of flash memory	16-bit bus width
All erasing module address	イレーズモジュールの先頭アドレスを指定 (H'FFF8 1000)
Writing module address	ライトモジュールの先頭アドレスを指定 (H'FFF8 0000)

6.1.4 ダウンロードモジュールの追加

デバッグメニューからデバッグの設定ウィンドウを開き、追加ボタンをクリックしてください。図 6.5に示すダウンロードモジュールウィンドウが表示されますので、ユーザプログラム（フラッシュメモリにロードしたいプログラム）をダウンロードモジュールに追加します。

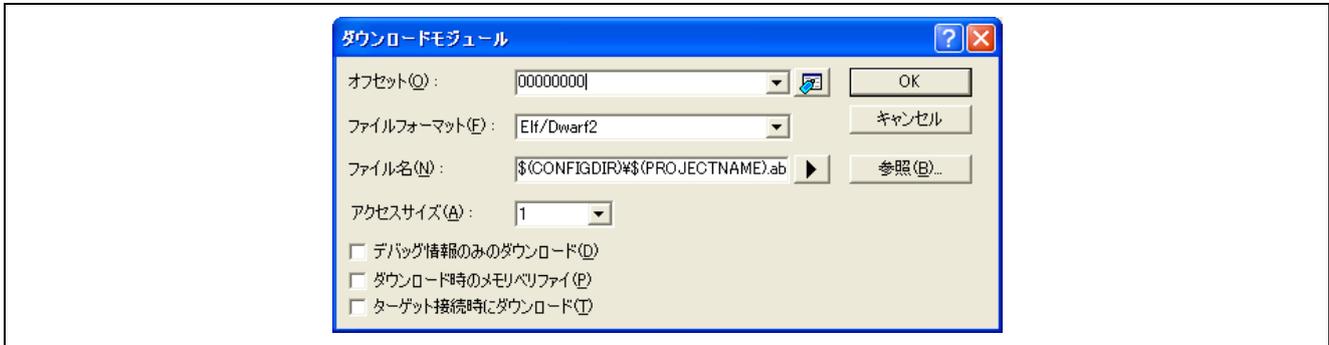


図 6.5 ダウンロードモジュールウィンドウ

6.1.5 ユーザプログラムのダウンロード

図 6.6に示すダウンロード機能にて、ユーザプログラムをダウンロードします。

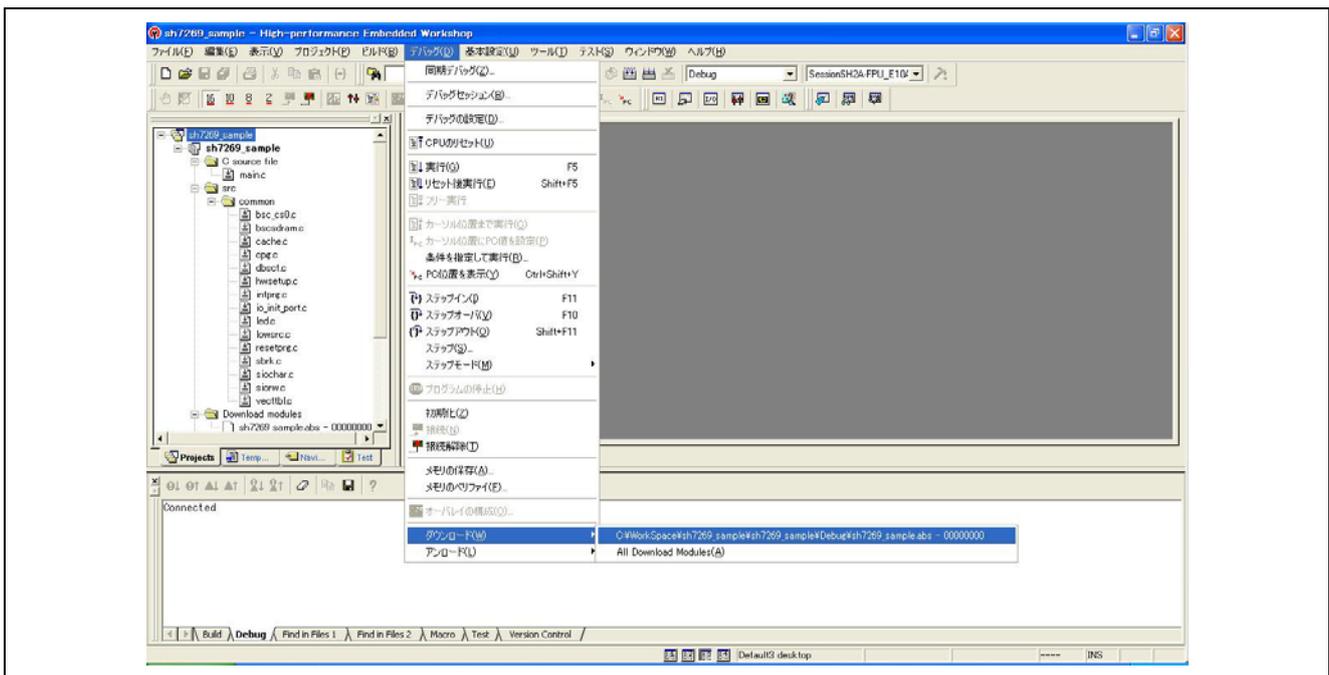


図 6.6 ユーザプログラムのダウンロード

6.2 FMTOOL のカスタマイズ

サンプルコードはフラッシュメモリのデバイス仕様に依存するため、デバイス変更時はプログラムのカスタマイズが必要な場合があります。

サンプルコードは、JEDEC 標準型コマンド互換方式に対応しています。JEDEC 標準型コマンド互換方式のフラッシュメモリをご使用の場合は、サンプルコードを流用することが可能です。

CUI (Command User Interface) コマンド方式のフラッシュメモリをご使用の場合は、サンプルコードは流用できません。この場合は、お客様が新規にダウンロードプログラムを作成してください。

- 【注】**
1. JEDEC 標準型コマンド互換方式は、あらかじめ決められたアドレス (H'555、H'2AA など) にコマンドを発行して書き込む方式です。
 2. CUI コマンド方式は、CUI に書き込みコマンド (H'40)、消去コマンド (H'20) を発行する方式です。

6.2.1 サンプルコードに対応するデバイス仕様

表 6.2と表 6.3に、使用デバイスの詳細仕様およびサンプルコードで使用するコマンドを示します。

表 6.2 使用デバイスの詳細仕様

項目	内容
メーカー	Spansion Inc.
型名	S29GL256P90TFIR1
容量	32M バイト
データバス幅	16 ビット
アクセスタイム	90ns
セクタ構造	ユニフォーム型
セクタサイズ	64K ワード
セクタ数	256
書き込み方式	JEDEC 標準型コマンド互換方式

表 6.3 サンプルコードで使用するコマンド

項目	内容		
	サイクル	アドレス	データ
イレーズコマンド (セクタイレーズ)	第一サイクル	H'555	H'AA
	第二サイクル	H'2AA	H'55
	第三サイクル	H'555	H'80
	第四サイクル	H'555	H'AA
	第五サイクル	H'2AA	H'55
	第六サイクル	SA *1	H'30
プログラムコマンド	第一サイクル	H'555	H'AA
	第二サイクル	H'2AA	H'55
	第三サイクル	H'555	H'A0
	第四サイクル	PA *2	PD *3
リセットコマンド	第一サイクル	- *4	H'F0

【注】 *1. SA はセクタアドレスを示します。イレーズするセクタアドレスを指定します。

*2. PA はプログラムアドレスを示します。ライトするアドレスを指定します。

*3. PD はプログラムデータを示します。ライトするデータを指定します。

*4. "-" はフラッシュメモリが配置されている空間のアドレスを指定します。フラッシュメモリが配置されている空間であれば、どのアドレスを指定しても問題はありません。

6.2.2 カスタマイズで対応可能なフラッシュメモリのブート型

サンプルコードをカスタマイズすることにより、以下の4種類のフラッシュメモリのブート型に対応することが可能です。

- (1) ユニフォーム型
- (2) ボトムブート型
- (3) トップブート型
- (4) デュアルブート型

図 6.7に、サンプルコードのカスタマイズで対応可能なフラッシュメモリのブート型のメモリマップ例を示します。

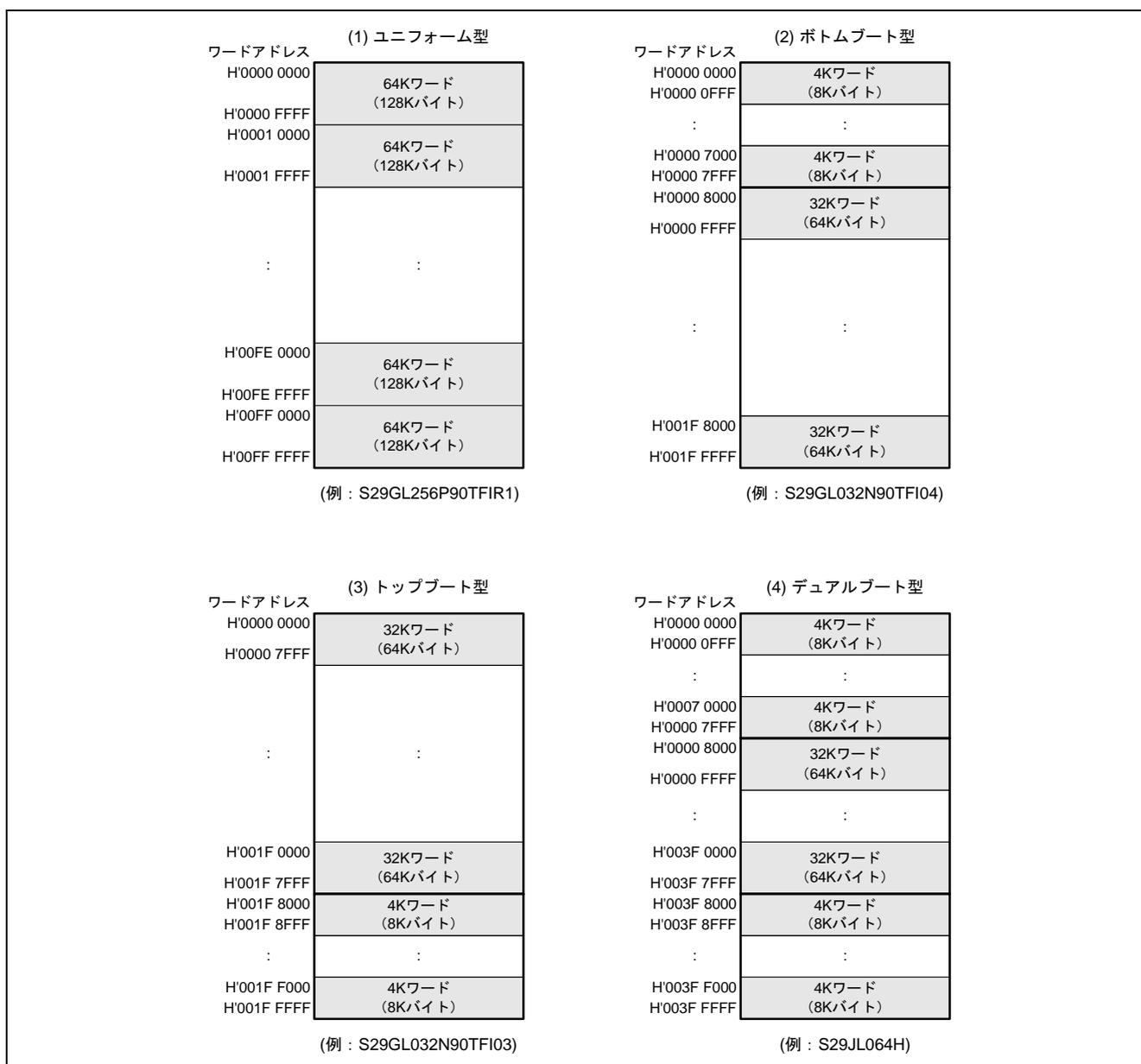


図 6.7 サンプルコードのカスタマイズで対応可能なフラッシュメモリのブート型

6.2.3 カスタマイズの内容

以下にカスタマイズが必要なケースと、その変更内容を示します。

表 6.4 カスタマイズが必要なケースとカスタマイズ内容 (1/2)

項目	カスタマイズ内容
フラッシュメモリのブート型が異なる場合	<p>サンプルコードはユニフォーム型のフラッシュメモリの消去および書き込みに対応しています。</p> <p>トップブート型、ボトムブート型、およびデュアルブート型のフラッシュメモリを使用する場合は、FM_BOOT_TYPE のマクロ定義を以下の値に変更してください。</p> <p>トップブート型 : FM_TOP_BOOT ボトムブート型 : FM_BOTTOM_BOOT デュアルブート型 : FM_DUAL_BOOT</p> <p>(初期値は、ユニフォーム型の"FM_UNIFORM"を定義していません。)</p>
フラッシュメモリのブート型を変更する場合 1 (ボトムブート型に変更)	<p>セクタの定義を変更してください。</p> <ul style="list-style-type: none"> • FM_B_BOOT_SECTOR_START • FM_B_BOOT_SECTOR_SIZE • FM_B_BOOT_SECTOR_NUM • FM_NORMAL_SECTOR_START • FM_NORMAL_SECTOR_SIZE • FM_NORMAL_SECTOR_NUM <p>定義の変更方法は、6.2.5を参照してください。</p> <p>以下のセクタ定義の初期値は"0"で定義しています。そのまま使用してください。</p> <ul style="list-style-type: none"> • FM_T_BOOT_SECTOR_START • FM_T_BOOT_SECTOR_SIZE • FM_T_BOOT_SECTOR_NUM
フラッシュメモリのブート型を変更する場合 2 (トップブート型に変更)	<p>セクタの定義を変更してください。</p> <ul style="list-style-type: none"> • FM_NORMAL_SECTOR_START • FM_NORMAL_SECTOR_SIZE • FM_NORMAL_SECTOR_NUM • FM_T_BOOT_SECTOR_START • FM_T_BOOT_SECTOR_SIZE • FM_T_BOOT_SECTOR_NUM <p>定義の変更方法は、6.2.6を参照してください。</p> <p>以下のセクタ定義の初期値は"0"で定義しています。そのまま使用してください。</p> <ul style="list-style-type: none"> • FM_B_BOOT_SECTOR_START • FM_B_BOOT_SECTOR_SIZE • FM_B_BOOT_SECTOR_NUM

【注】 FMTOOLはフラッシュメモリの仕様に依存するため、表 6.4および表 6.5が全てのチェック項目を網羅しているわけではありません。使用されるデバイスのデータシートをご確認の上、デバイスの仕様に合わせてFMTOOLを修正してください。

表 6.5 カスタマイズが必要なケースとカスタマイズ内容 (2/2)

項目	カスタマイズ内容
フラッシュメモリのブート型を変更する場合 3 (デュアルブート型に変更)	セクタの定義を変更してください。 <ul style="list-style-type: none"> • FM_B_BOOT_SECTOR_START • FM_B_BOOT_SECTOR_SIZE • FM_B_BOOT_SECTOR_NUM • FM_NORMAL_SECTOR_START • FM_NORMAL_SECTOR_SIZE • FM_NORMAL_SECTOR_NUM • FM_T_BOOT_SECTOR_START • FM_T_BOOT_SECTOR_SIZE • FM_T_BOOT_SECTOR_NUM 定義の変更方法は、6.2.7を参照してください。
セクタのサイズ、セクタ数が異なる場合	セクタの定義を変更してください。 <ul style="list-style-type: none"> • FM_NORMAL_SECTOR_START • FM_NORMAL_SECTOR_SIZE • FM_NORMAL_SECTOR_NUM 定義の変更方法は、6.2.4を参照してください。 以下のセクタ定義の初期値は"0"で定義しています。そのまま使用してください。 <ul style="list-style-type: none"> • FM_B_BOOT_SECTOR_START • FM_B_BOOT_SECTOR_SIZE • FM_B_BOOT_SECTOR_NUM • FM_T_BOOT_SECTOR_START • FM_T_BOOT_SECTOR_SIZE • FM_T_BOOT_SECTOR_NUM
メモリ容量が異なる場合	FM_END_ADDRESS マクロの設定値を変更してください。
図 6.7に示す 4 種類のブート型とは異なる場合	フラッシュメモリの操作関数のカスタマイズが必要です。詳細はサンプルコードを参照してください。
イレーズ、ライト、およびリセットのコマンド仕様が異なる場合	
書き込みコマンドが CUI コマンド方式の場合	
SH7269 とフラッシュメモリを 16 ビット以外で接続している場合	

【注】 FMTOOLはフラッシュメモリの仕様に依存するため、表 6.4および表 6.5が全てのチェック項目を網羅しているわけではありません。使用されるデバイスのデータシートをご確認の上、デバイスの仕様に合わせてFMTOOLを修正してください。

6.2.4 ユニフォーム型のセクタサイズやセクタ数をカスタマイズする方法

図 6.8に、ユニフォーム型のフラッシュメモリのセクタサイズやセクタ数を変更するカスタマイズ方法を示します。フラッシュメモリS29GL256P90TFIR1 とセクタサイズやセクタ数の異なるユニフォーム型のフラッシュメモリを使用する場合は、ユニフォーム型のセクタ情報のマクロの設定値を変更します。フラッシュメモリの容量が異なる場合は、最終アドレスのマクロの設定値も変更します。

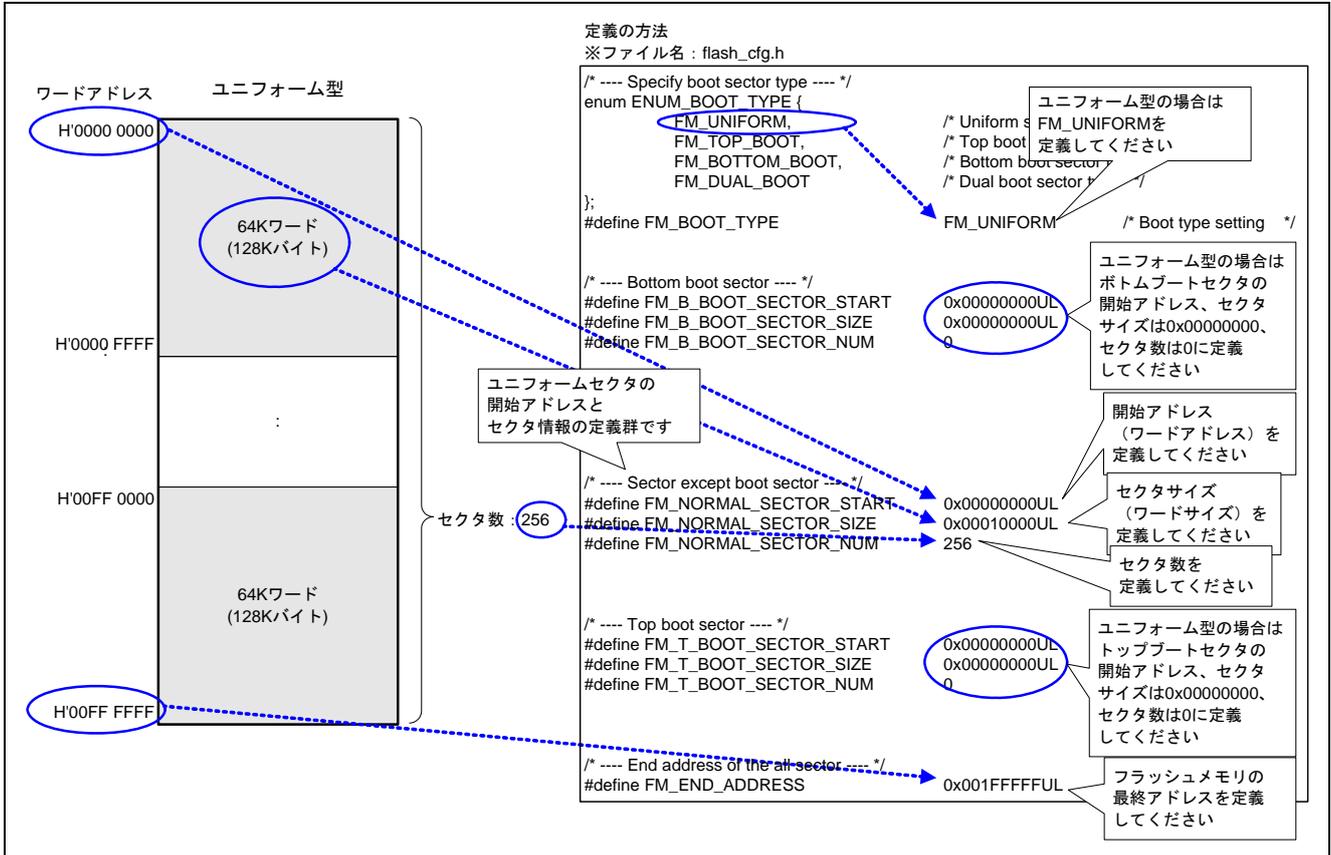


図 6.8 ユニフォーム型のフラッシュメモリのセクタサイズやセクタ数を変更するカスタマイズ方法

6.2.5 フラッシュメモリのブート型をボトムブート型にカスタマイズする方法

図 6.9にフラッシュメモリのブート型をボトムブート型に変更する場合のカスタマイズ方法を示します。ボトムブート型のフラッシュメモリを使用する場合は、ボトムブートセクタ領域およびブートセクタ以外の領域の2種類のセクタ情報のマクロの設定値を変更します。フラッシュメモリの容量が異なる場合は、最終アドレスのマクロの設定値も変更します。

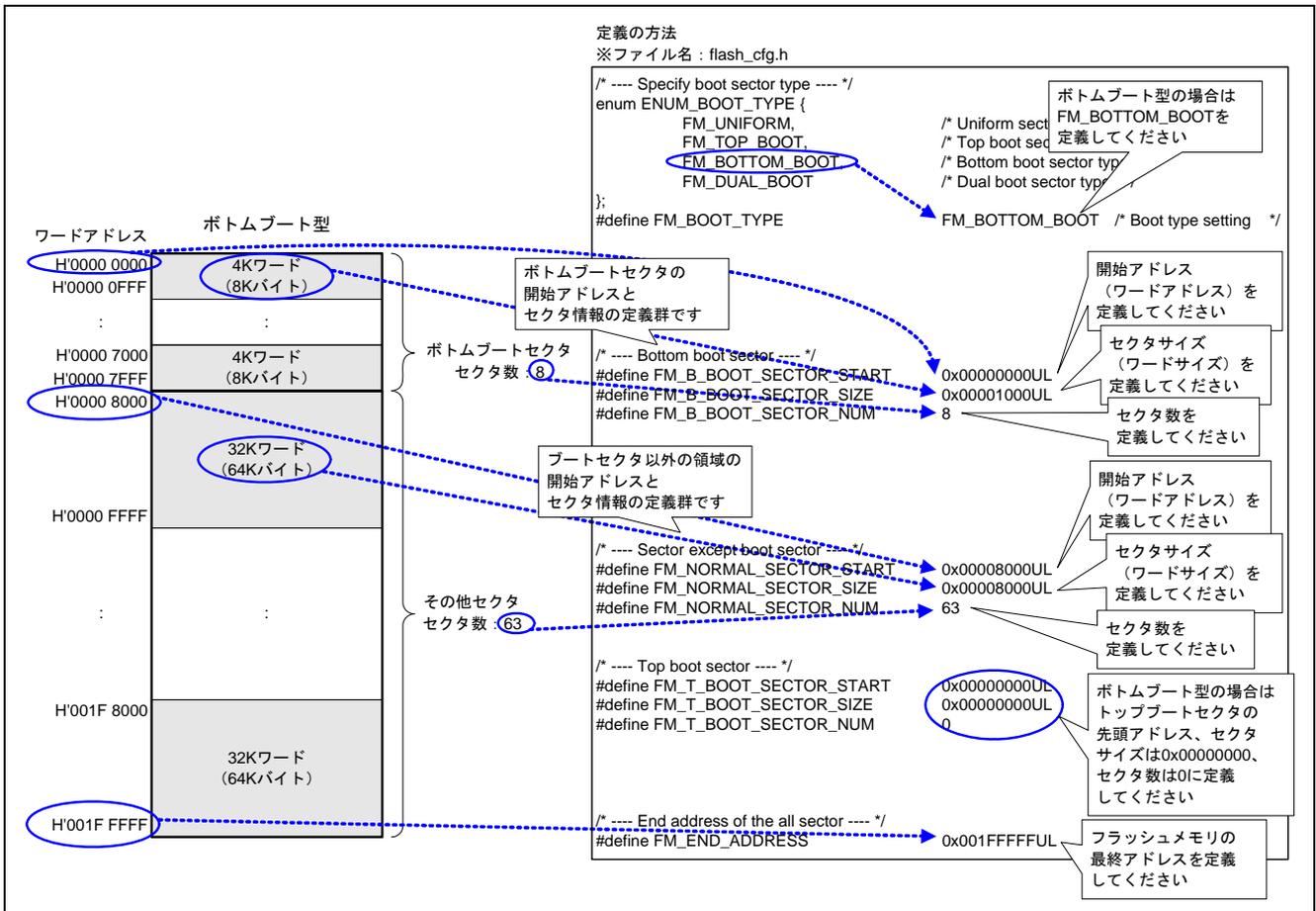


図 6.9 フラッシュメモリのブート型をボトムブート型に変更する場合のカスタマイズ方法

6.2.6 フラッシュメモリのブート型をトップブート型にカスタマイズする方法

図 6.10に、フラッシュメモリのブート型をトップブート型に変更する場合のカスタマイズ方法を示します。トップブート型のフラッシュメモリを使用する場合は、トップブートセクタ領域およびブートセクタ以外の領域の2種類のセクタ情報のマクロの設定値を変更します。フラッシュメモリの容量が異なる場合は、最終アドレスのマクロの設定値も変更します。

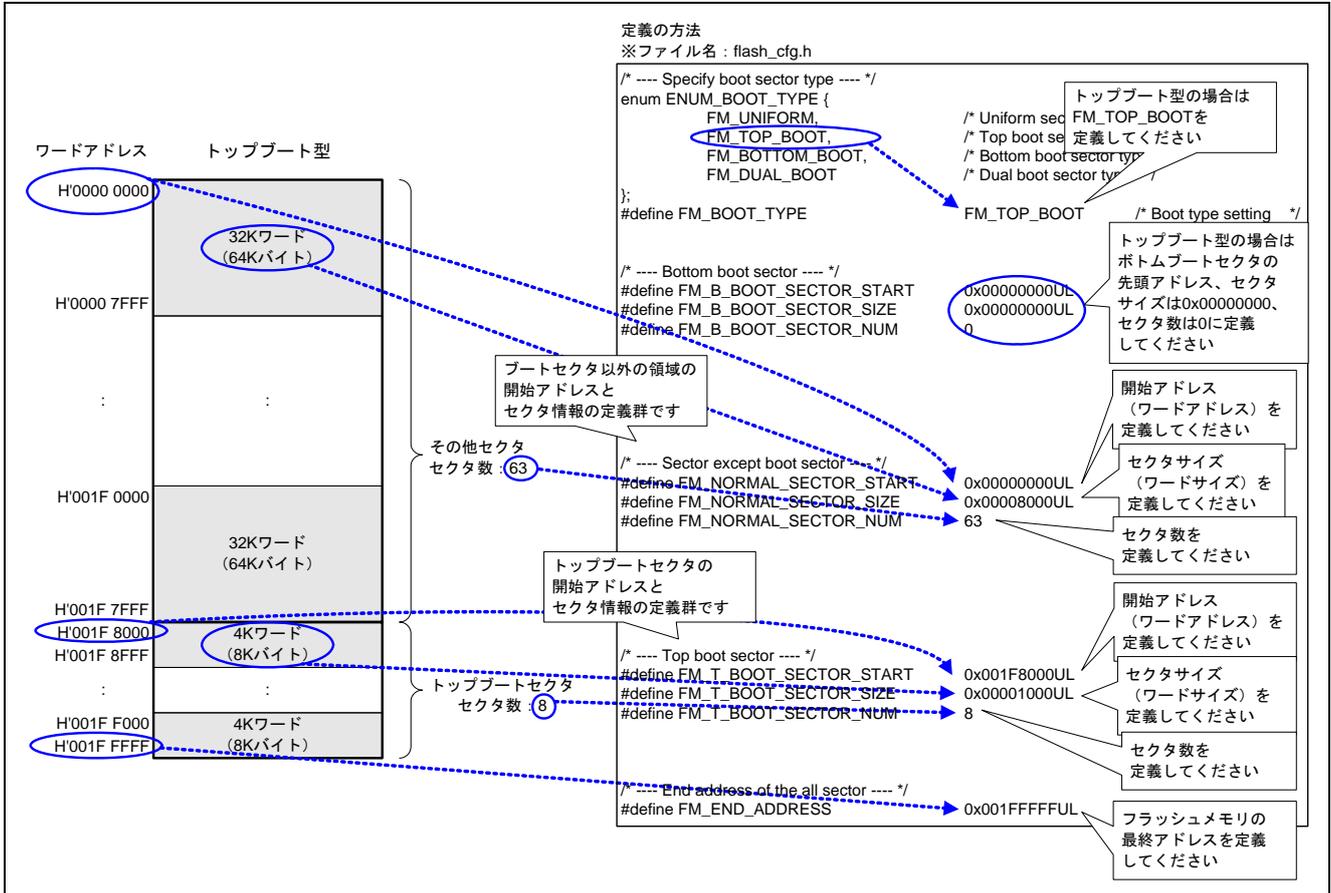


図 6.10 フラッシュメモリのブート型をトップブート型に変更する場合のカスタマイズ方法

6.2.7 フラッシュメモリのブート型をデュアルブート型にカスタマイズする方法

図 6.11に、フラッシュメモリのブート型をデュアルブート型に変更する場合のカスタマイズ方法を示します。デュアルブート型のフラッシュメモリを使用する場合は、トップブートセクタ領域、ボトムブートセクタ領域、およびブートセクタ以外の領域の3種類のセクタ情報のマクロの設定値を変更します。フラッシュメモリの容量が異なる場合は、最終アドレスのマクロの設定値も変更します。

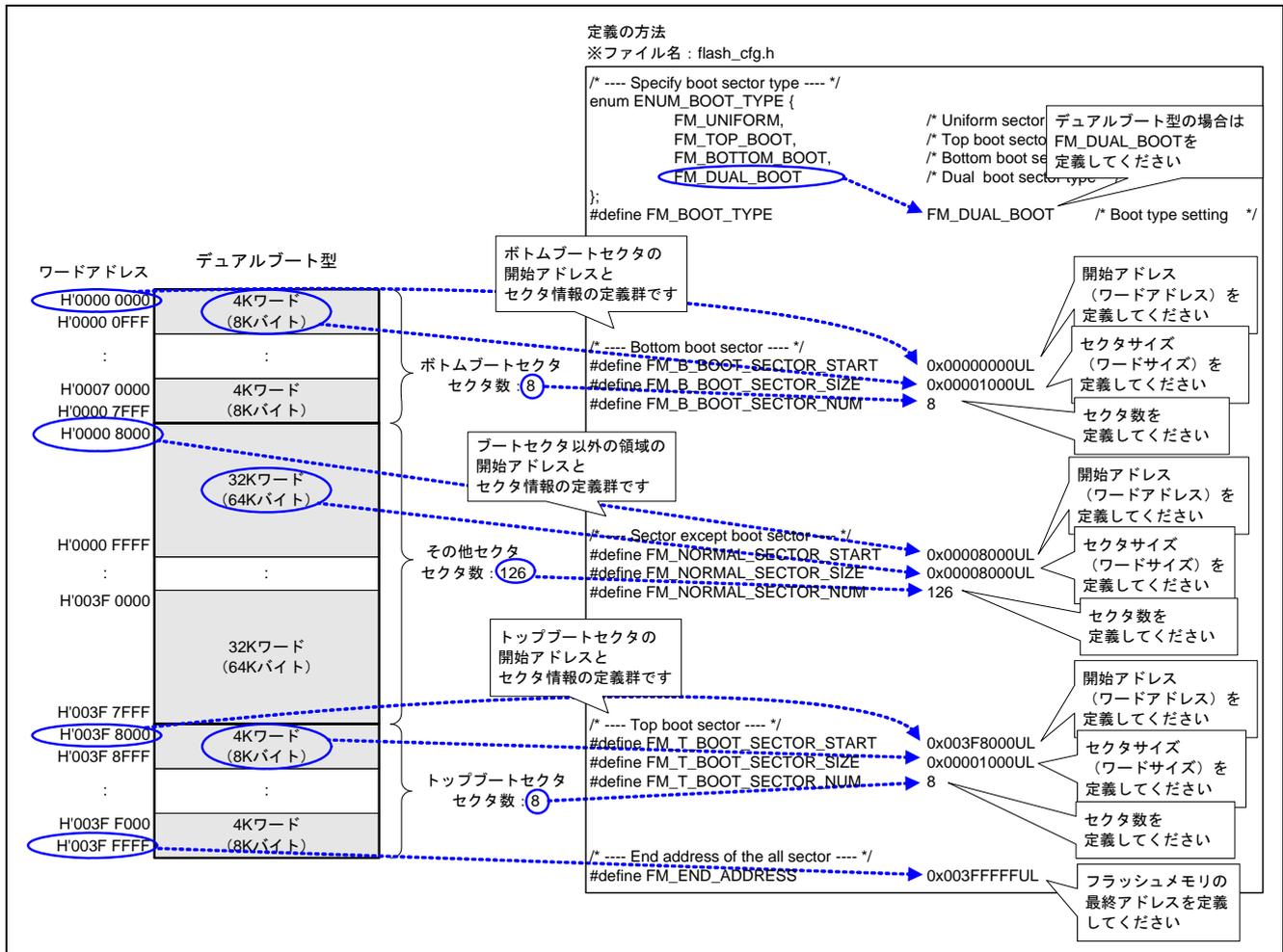


図 6.11 フラッシュメモリのブート型をデュアルブート型に変更する場合のカスタマイズ方法

6.2.8 フラッシュメモリのコマンドの確認

図 6.12に、フラッシュメモリのコマンドを示します。サンプルコードは、JEDEC標準型コマンド互換方式に対応しており、SH7269 とフラッシュメモリを 16 ビットで接続する場合のコマンドを定義しています。フラッシュメモリのデータシートでワードモード (x16 ビットモード) のコマンドを参照して、リセットコマンド、セクタイレースコマンド、およびプログラムコマンド (ライトコマンド) のコマンド内容に相違がないことを確認してください。なお、サンプルコードでは、ページプログラム (ページライト) には対応していません。

※ファイル名 : flash_cfg.h

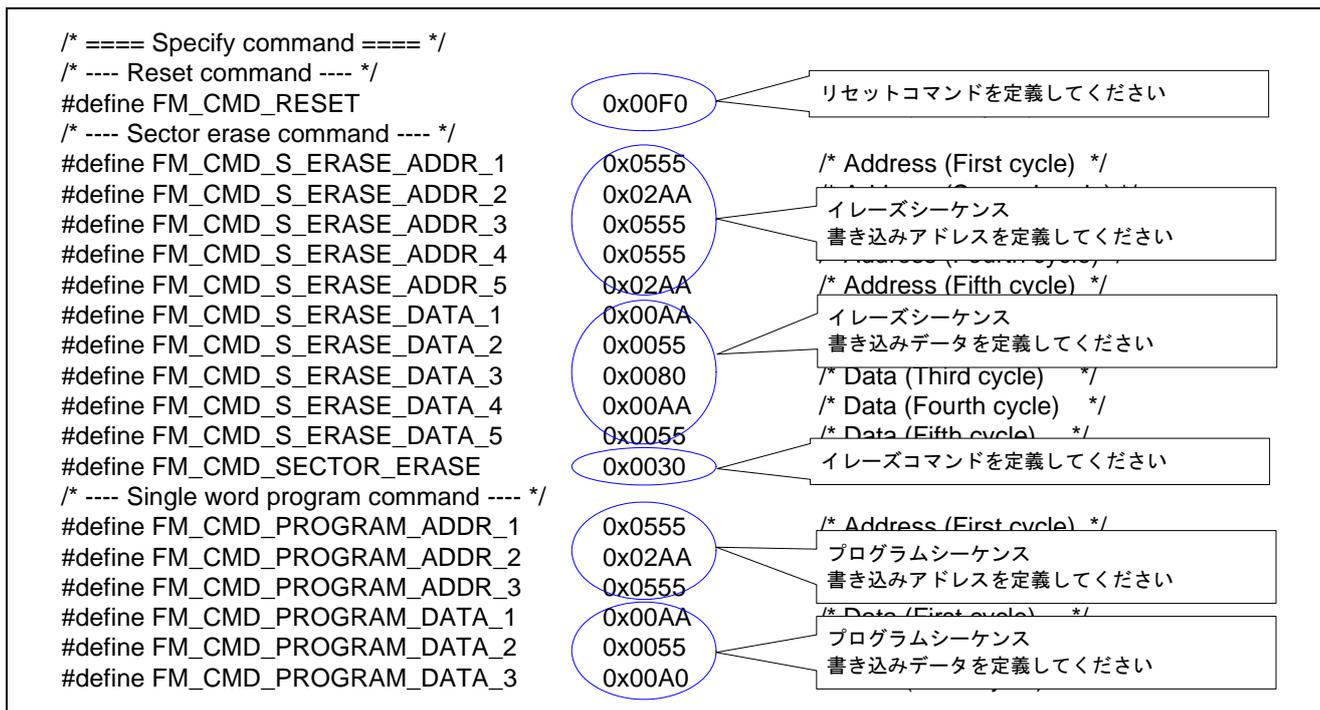


図 6.12 フラッシュメモリのコマンド定義

7. サンプルコード

サンプルコードは、ルネサス エレクトロニクスホームページから入手してください。

8. 参考ドキュメント

SH7268/SH7269 グループ ユーザーズマニュアル ハードウェア編 Rev.1.00

(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート/テクニカルニュース

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

C コンパイラマニュアル

SuperH RISC engine ファミリ C/C++コンパイラパッケージ V.9.04

C コンパイラユーザーズマニュアル Rev.1.01

(最新版をルネサス エレクトロニクスホームページから入手してください。)

SuperH ファミリ用 E10A-USB エミュレータ ユーザーズマニュアル Rev.9.00

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

SuperH ファミリ用 E10A-USB エミュレータ ユーザーズマニュアル 別冊

SH7264、SH7262、SH7266、SH7267、SH7268、SH7269 ご使用時の補足説明 Rev.4.00

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

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>
- お問合せ先
<http://japan.renesas.com/inquiry>

改訂記録	SH7268/SH7269 グループ E10A-USB フラッシュメモリダウンロード機能の応用例（NOR フラッシュメモリへのダウンロード）
------	--

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2012.07.18	—	初版発行

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、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

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. リザーブアドレスのアクセス禁止

【注意】リザーブアドレス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、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違っていると、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。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。

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発生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も、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・電話番号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 エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■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/>