
RX ファミリ

R01AN2130JJ0100

Rev.1.00

DTC のブロック転送モードとチェーン転送の設定例

2014.07.01

要旨

本アプリケーションノートでは、RX ファミリのデータトランスファコントローラ(以下、DTC とする)のブロック転送とチェーン転送について説明します。

対象デバイス

- ・RX ファミリ

RX610、RX62N、RX621、RX62T、RX62G グループを除く。

DTC のブロック転送とチェーン転送の設定例として、RX63N グループを用いて説明しています。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	3
2. 動作確認条件	4
3. 関連アプリケーションノート	4
4. ブロック転送モードとチェーン転送の説明	5
4.1 ブロック転送モードとは	5
4.2 チェーン転送とは	7
5. ハードウェア説明	9
5.1 ハードウェア構成例	9
5.2 使用端子一覧	9
6. ソフトウェア説明	10
6.1 DTC の設定	12
6.1.1 DTC の初期設定	12
6.1.2 DTC の再設定	17
6.2 ファイル構成	18
6.3 オプション設定メモリ	19
6.4 構造体/共用体一覧	20
6.5 変数一覧	21
6.6 関数一覧	21
6.7 関数仕様	22
6.8 フローチャート	25
6.8.1 メイン処理	25
6.8.2 ポート初期設定	25
6.8.3 周辺機能初期設定	26
6.8.4 IRQ 初期設定	27
6.8.5 DTC 初期設定	29
6.8.6 IRQ10 割り込み処理	30
6.8.7 IRQ8 割り込み処理	30
7. サンプルコード	31
8. 参考ドキュメント	31

1. 仕様

IRQ を DTC の起動要因にして、DTC のブロック転送とチェーン転送を行います。

外部割り込み要求端子(IRQ10)の立ち下がりエッジを検出すると、ブロック転送を行います。また、外部割り込み要求端子(IRQ8)の立ち下がりエッジを検出すると、チェーン転送を行います。

表 1.1 に使用する周辺機能と用途を、図 1.1 にブロック図を示します。

表1.1 使用する周辺機能と用途

周辺機能	用途
DTCa	ROM データを RAM へ転送(ブロック転送、チェーン転送)
IRQ10	DTC (ブロック転送) の起動要因
IRQ8	DTC (チェーン転送) の起動要因

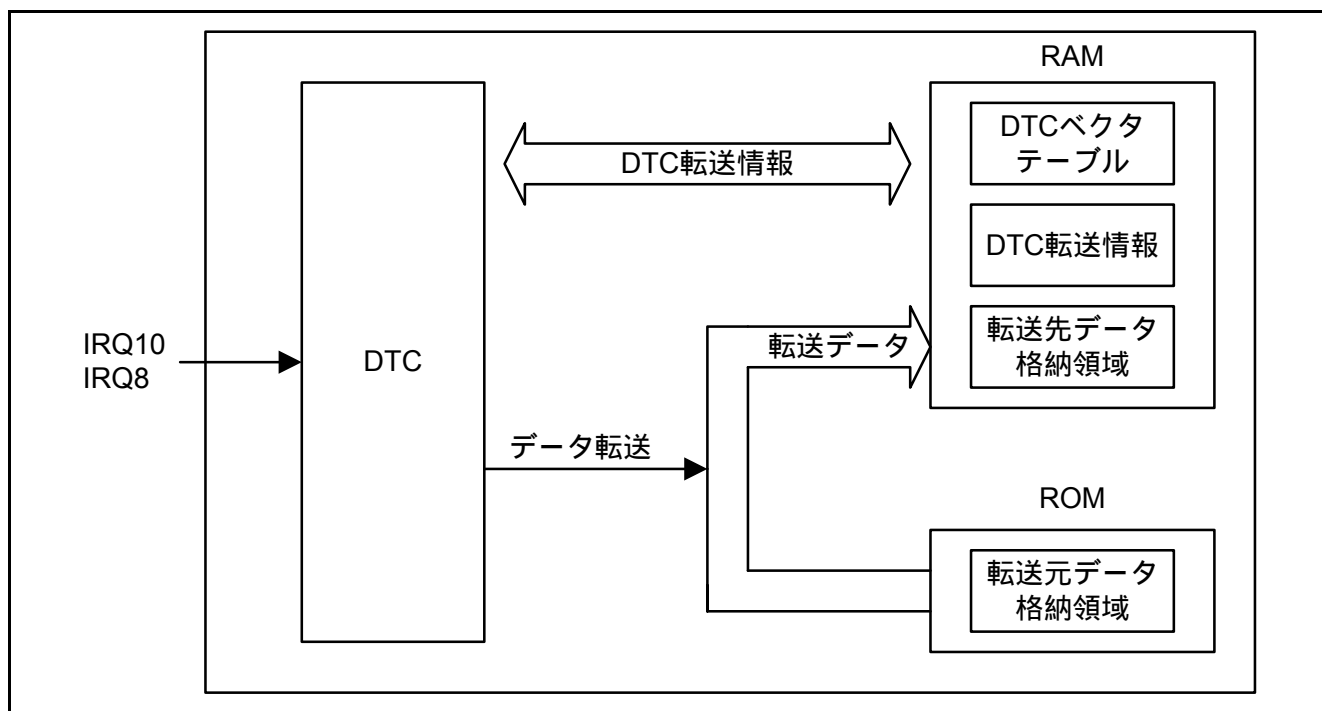


図1.1 ブロック図

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	R5F563NBDDFC (RX63N グループ)
動作周波数	<ul style="list-style-type: none"> メインクロック: 12MHz PLL: 192MHz (メインクロック 1 分周 16 通倍) システムクロック (ICLK): 96MHz (PLL 2 分周) 周辺モジュールクロック B (PCLKB): 48MHz (PLL 4 分周)
動作電圧	3.3V
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded Workshop Version 4.09.01
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family V.1.02 Release 01 コンパイルオプション -cpu=rx600 -output=obj="\$ (CONFIGDIR)¥\$(FILELEAF).obj" -debug -nologo (統合開発環境のデフォルト設定を使用しています)
iodefine.h のバージョン	Version 1.6A
エンディアン	リトルエンディアン
動作モード	シングルチップモード
プロセッサモード	スーパバイザモード
サンプルコードのバージョン	Version 1.00
使用ボード	Renesas Starter Kit+ for RX63N (製品型名: R0K50563NC000BE)

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。

- RX63N グループ、RX631 グループ 初期設定例 Rev.1.10 (R01AN1245JJ)

上記アプリケーションノートの初期設定関数を、本アプリケーションノートのサンプルコードで使用しています。Rev は本アプリケーションノート作成時点のものです。

最新版がある場合、最新版に差し替えて使用してください。最新版はルネサスエレクトロニクスホームページで確認および入手してください。

4. ブロック転送モードとチェーン転送の説明

ブロック転送モードとチェーン転送について補足します。

4.1 ブロック転送モードとは

ブロック転送モードは、1 回の起動要因で 1 ブロック(転送サイズ×ブロックサイズ)のデータを転送するモードで、転送元、転送先のいずれかをブロック領域に設定します

以下に、ブロック転送モードに関連するレジスタについて説明します。

- MRA.SZ ビットと CRAH レジスタ

MRA.SZ ビットと CRAH レジスタでは、転送サイズとブロックサイズを設定します。

MRA.SZ ビットには転送サイズ、8、16、32 ビットのいずれかが設定できます。

CRAH レジスタにはブロックサイズ、1~256 のいずれかが設定できます。

図 4.1 にブロック転送モード時の MRA.SZ ビットと CRAH レジスタの関係を示します

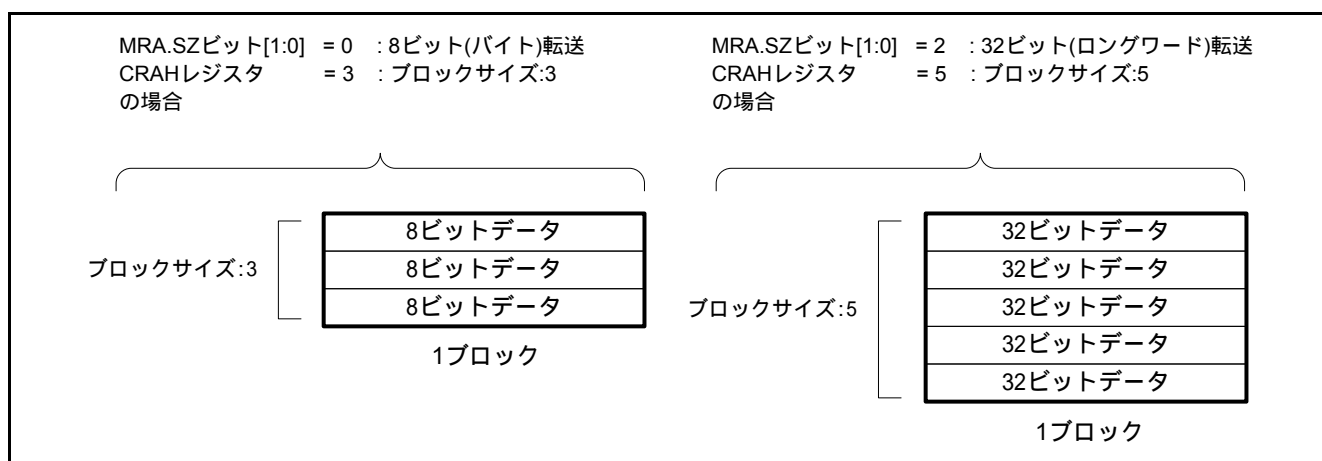


図4.1 ブロック転送モード時の MRA.SZ ビットと CRAH レジスタの関係

• MRB.DTS ビット

MRB.DTS ビットは転送元か転送先のどちらかをブロック領域に設定するビットです。1 ブロックの転送が終わると、転送元アドレスをブロック領域に指定した場合は SAR が、転送先アドレスをブロック領域に指定した場合は DAR が初期化されます。図 4.2に 1 ブロックの転送を示します。

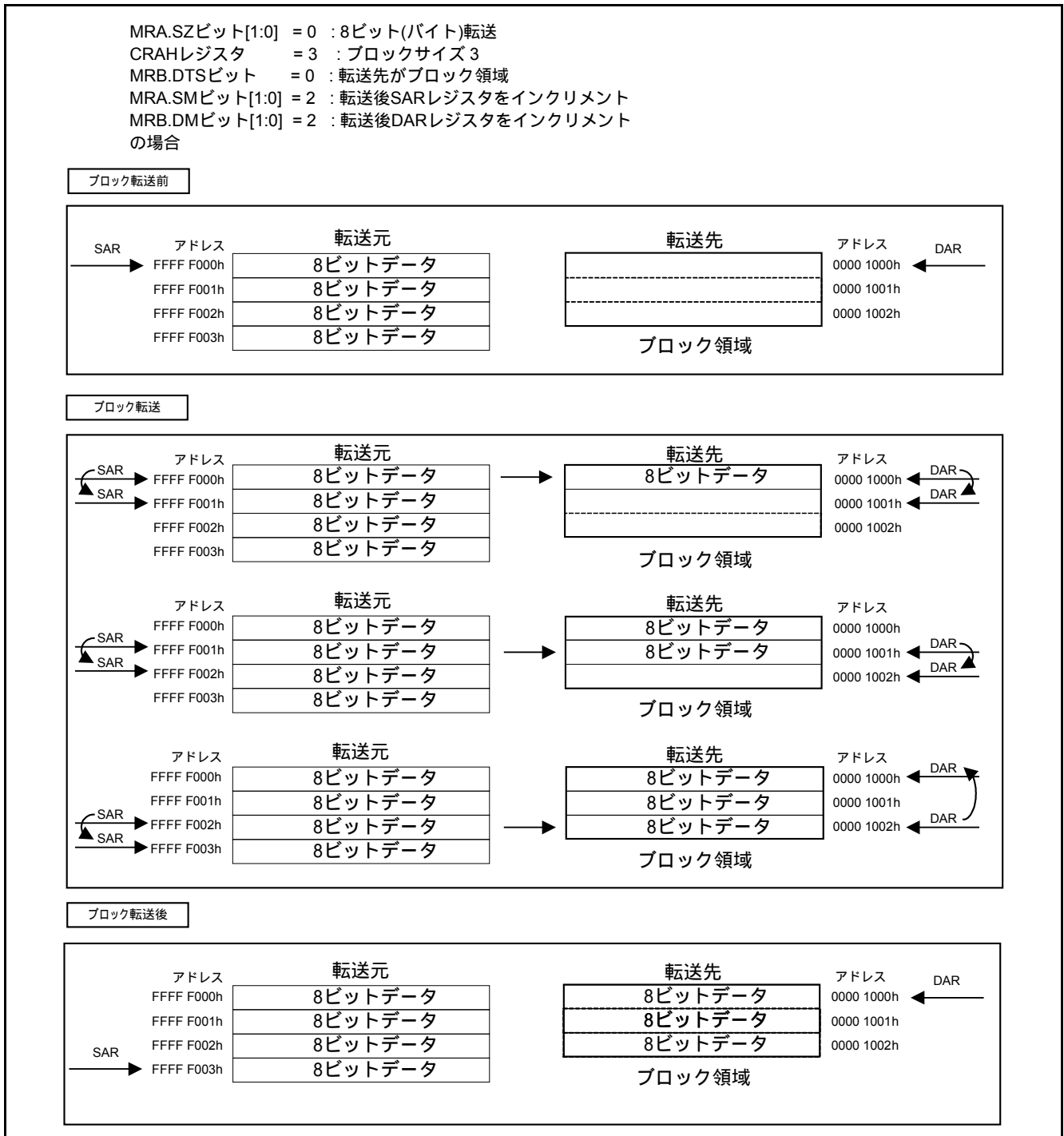


図4.2 1 ブロックの転送

4.2 チェーン転送とは

チェーン転送は、1回の起動要因で複数の転送情報を元にデータ転送をする機能です。ノーマル転送モード、ブロック転送モード、リピート転送モードのいずれのモードでもできます。

以下に、チェーン転送に関連するレジスタについて説明します。

● MRB.CHNS ビット

MRB.CHNS ビットは MRB.CHNE ビットが “ 1 ” (DTC チェーン転送許可)のときのチェーン転送実行の条件を決めるビットです。MRB.CHNS ビットが “ 0 ”のときは、無条件でチェーン転送を実行します。MRB.CHNS ビットが “ 1 ” のときは、転送モードによってチェーン転送実行の条件が異なります。図 4.3に MRB.CHNS ビットが “ 1 ” のときのノーマル転送モード時のチェーン転送例を示します。

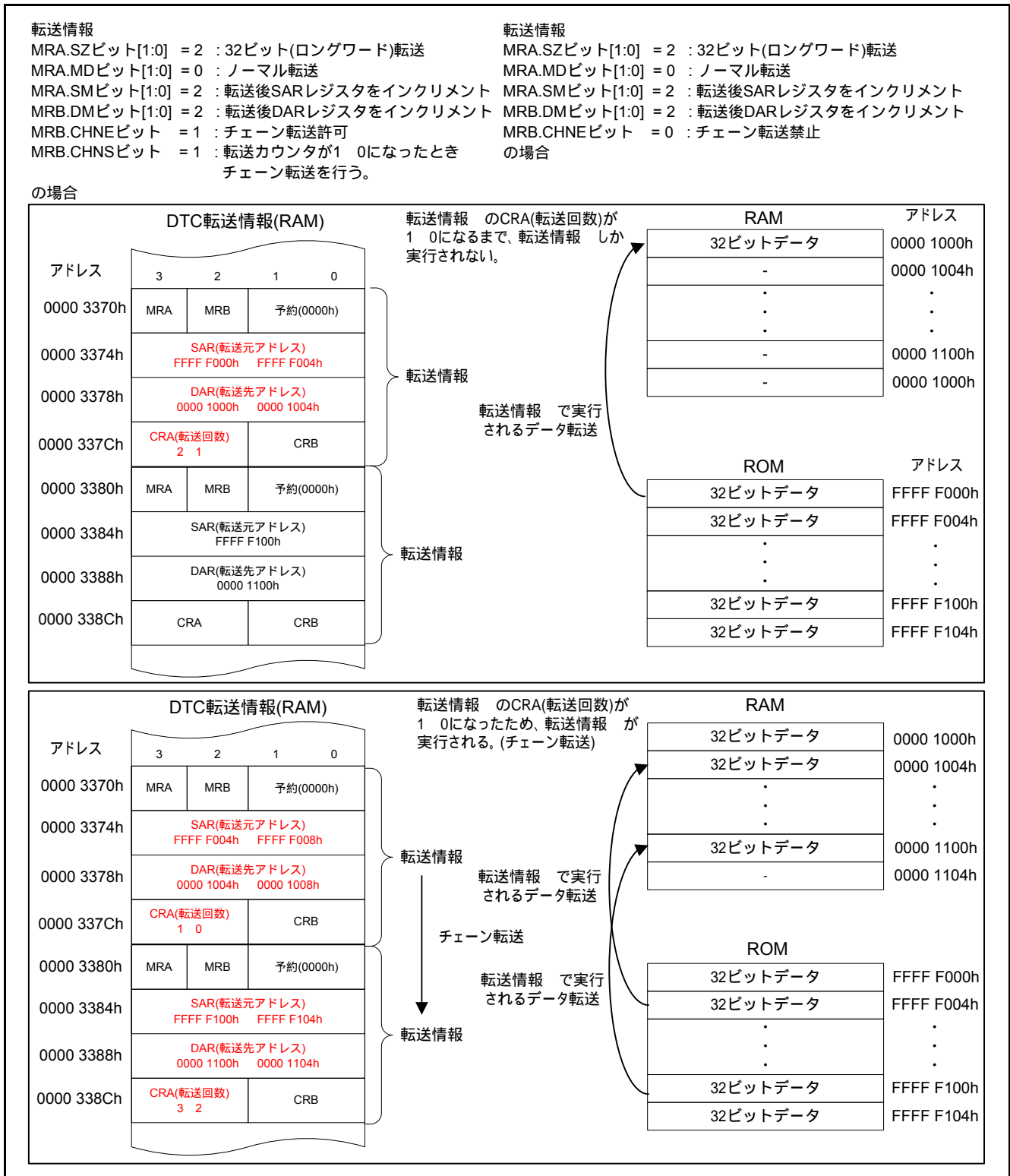


図4.3 ノーマル転送モード時のチェーン転送例

5. ハードウェア説明

5.1 ハードウェア構成例

図 5.1に接続例を示します。

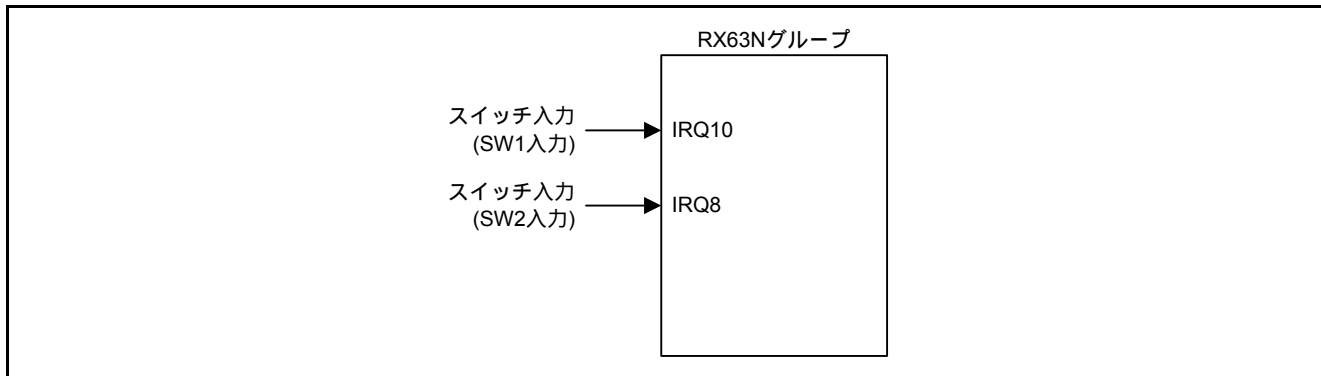


図5.1 接続例

5.2 使用端子一覧

表 5.1に使用端子と機能を示します。

表5.1 使用端子と機能

端子名	入出力	内容
P02/IRQ10	入力	DTC (ブロック転送) の起動要因の SW 入力
P00/IRQ8	入力	DTC (チェーン転送) の起動要因の SW 入力

6. ソフトウェア説明

本サンプルコードでは、SW を押すと設定した回数の DTC 転送を行い、設定回数の転送が完了すると IRQ 端子割り込み要求が発生します。

割り込み処理では、DTC 転送を再度行うため、DTC の転送情報を再設定しています。

SW1 を押すと、ブロック転送モードの DTC 転送が行われ、5 回の転送が完了すると、IRQ10 端子割り込みの処理を実行します。

SW2 を押すと、チェーン転送を使用した 2 つのノーマル転送モードによる DTC 転送が行われます。転送が完了すると、IRQ8 端子割り込みの処理を実行します。

使用する周辺機能の設定を以下に示します。

<DTC>

[ブロック転送モードの DTC 設定]

- 起動要因 : IRQ10 端子割り込み要求
- DTC アドレスモード : フルアドレスモード
- 転送モード : ブロック転送モード
- 1 ブロックサイズ : 32 ビット(ロングワード)×3
- 転送元アドレッシングモード : 転送後 SAR レジスタをインクリメント
- 転送元アドレス : dtc_block_sar (ROM) の先頭アドレス
- 転送先アドレッシングモード : 転送後 DAR レジスタをインクリメント
- 転送先アドレス : dtc_block_dar (RAM) の先頭アドレス
- データ転送単位 : 32 ビット(ロングワード)
- 転送回数 : 5 回
- チェーン転送 : 禁止
- 割り込み : 指定された転送終了後、CPU への割り込み要求が発生

[チェーン転送使用時の転送 1 回目の DTC 設定]

- 起動要因 : IRQ8 端子割り込み要求
- DTC アドレスモード : フルアドレスモード
- 転送モード : ノーマル転送モード
- 転送元アドレッシングモード : 転送後 SAR レジスタをインクリメント
- 転送元アドレス : dtc_chain0_sar(ROM)の先頭アドレス
- 転送先アドレッシングモード : 転送後 DAR レジスタをインクリメント
- 転送先アドレス : dtc_chain0_dar(RAM)の先頭アドレス
- データ転送単位 : 32 ビット(ロングワード)
- 転送回数 : 3 回
- チェーン転送 : 許可
- チェーン転送選択 : 連続してチェーン転送を行う

[チェーン転送使用時の転送 2 回目の DTC 設定]

- DTC アドレスモード : フルアドレスモード
- 転送モード : ノーマル転送モード
- 転送元アドレッシングモード : 転送後 SAR レジスタをインクリメント
- 転送元アドレス : dtc_chain1_sar(ROM)の先頭アドレス
- 転送先アドレッシングモード : 転送後 DAR レジスタをインクリメント
- 転送先アドレス : dtc_chain1_dar(RAM)の先頭アドレス
- データ転送単位 : 32 ビット(ロングワード)
- 転送回数 : 3 回
- チェーン転送 : 禁止
- 割り込み : 指定された転送終了後、CPU への割り込み要求が発生

<IRQ10>

- 検出方法 : 立ち下がリエッジ
- デジタルフィルタ : 有効(PCLK/64)
- 割り込み : 使用する(DTC の起動要因)

<IRQ8>

- 検出方法 : 立ち下がリエッジ
- デジタルフィルタ : 有効(PCLK/64)
- 割り込み : 使用する(DTC の起動要因)

6.1 DTC の設定

6.1.1 DTC の初期設定

(1) 転送情報を配置

MRA、MRB、SAR、DAR、CRA、CRB レジスタは DTC の内部レジスタです。CPU から直接アクセスすることはできません。これら内部レジスタの設定値は RAM 領域に転送情報として配置します。図 6.1 にリトルエンディアンの場合の転送情報の配置を示します。

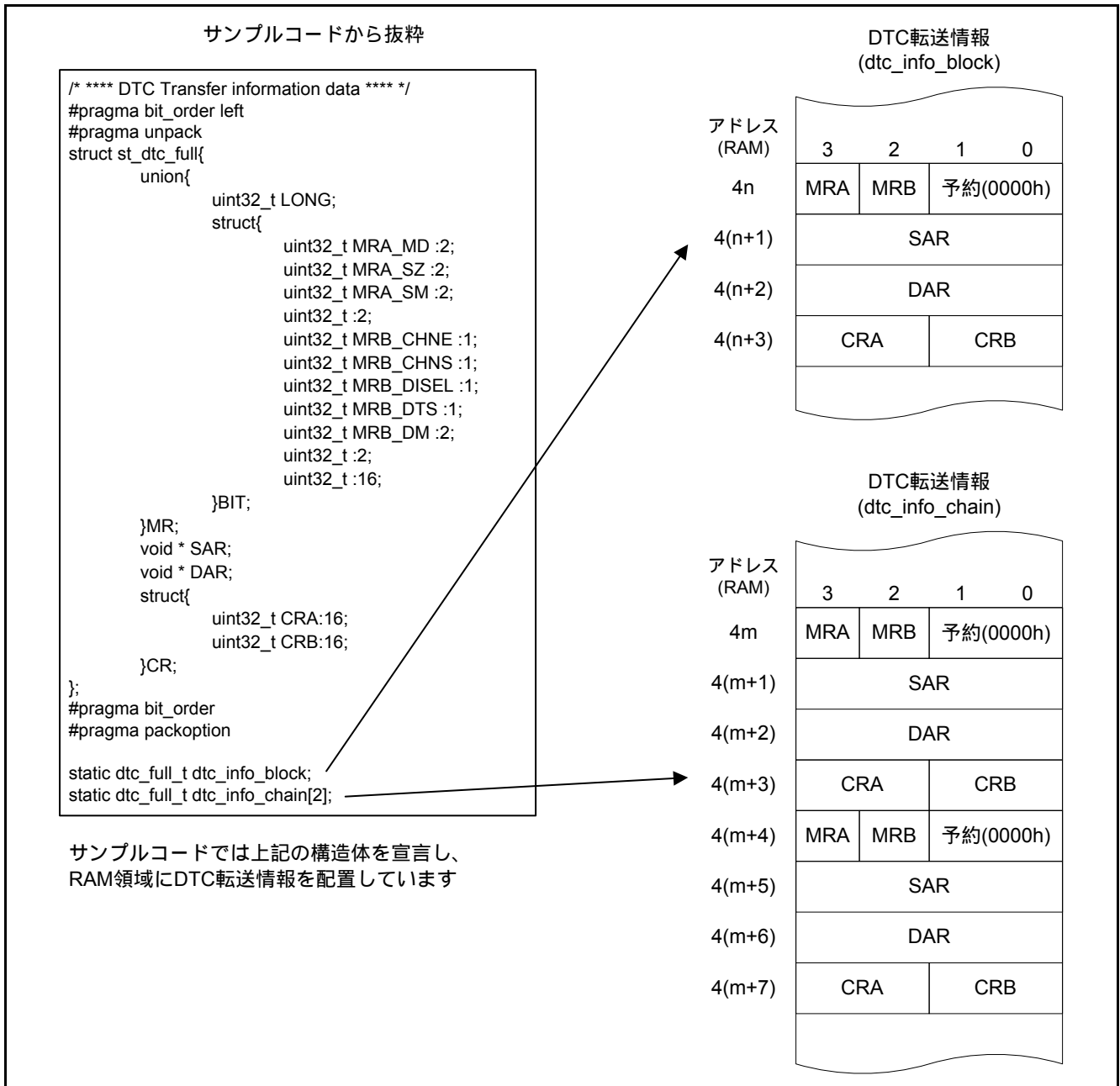


図6.1 転送情報の配置

(2) セクションの追加

DTC は起動要因に応じて、DTC ベクタテーブルから指定された転送情報を読み出します。DTC ベクタテーブルには、転送情報の先頭アドレスを設定します。

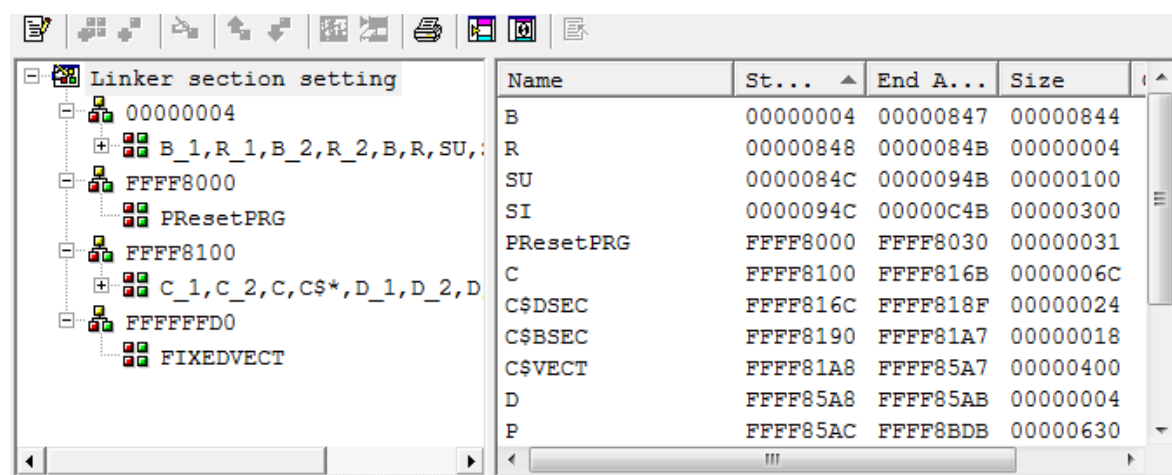
サンプルコードでは、DTC ベクタテーブルの先頭 3000h(RAM)に配置するため、セクション(以降、DTC_SECTION とする)を追加しています。セクションの追加は、統合開発環境 High-performance Embedded Workshop(以降 HEW とする)の Map Section Information ウィンドウで設定しています。セクションの追加 / 変更および削除の詳細は、最新の High-performance Embedded Workshop ユーザーズマニュアルを参照してください。

HEW でのセクションの追加方法

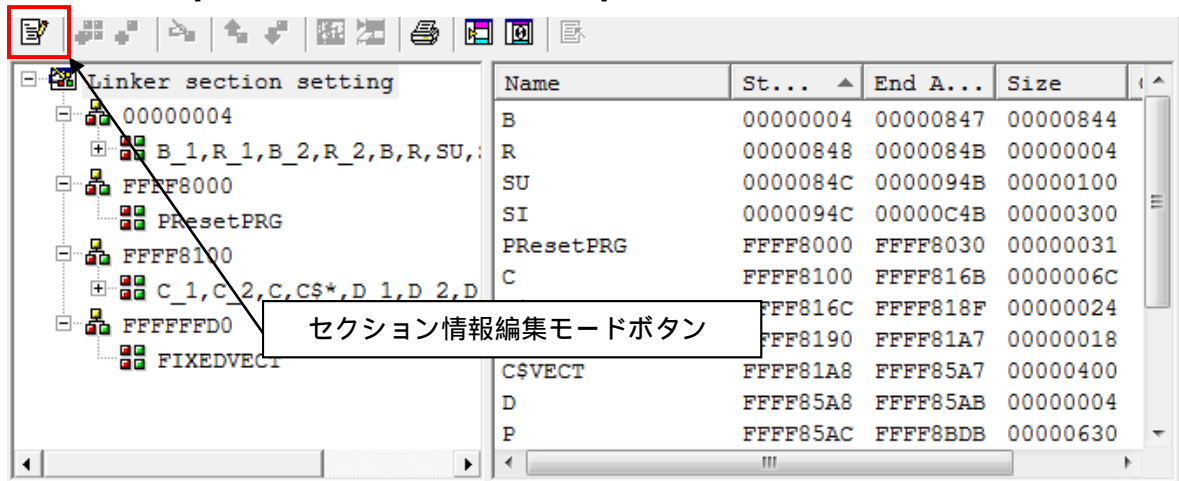
以下の手順で HEW の Map Section Information ウィンドウを開きます。

1. メニューバーから[表示 -> マップ]を選択します。
2. [マップ種別]ドロップダウンリストから"Map Section Information"を選択します。
3. [OK]ボタンをクリックします。

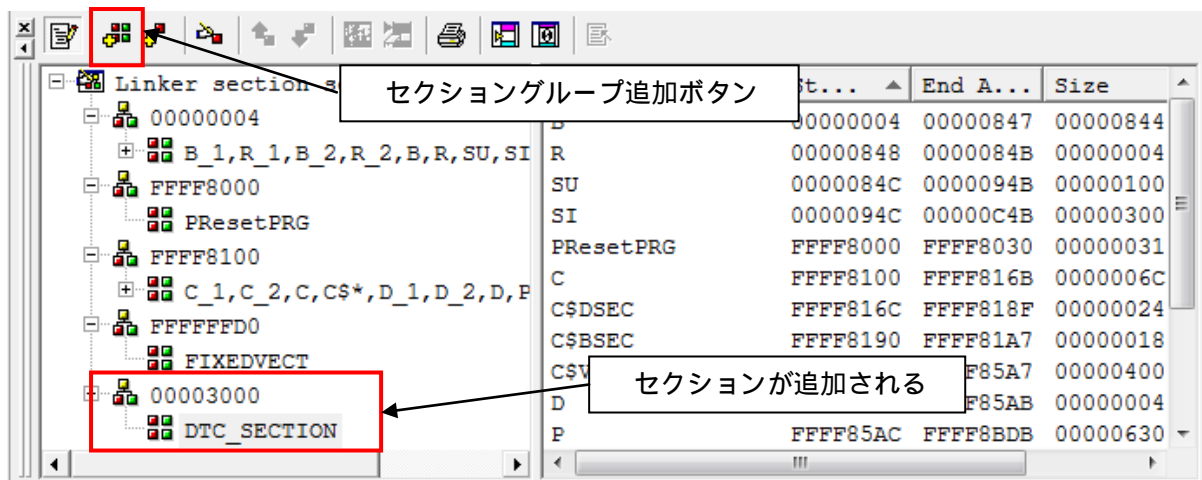
[Map Section Information ウィンドウ]



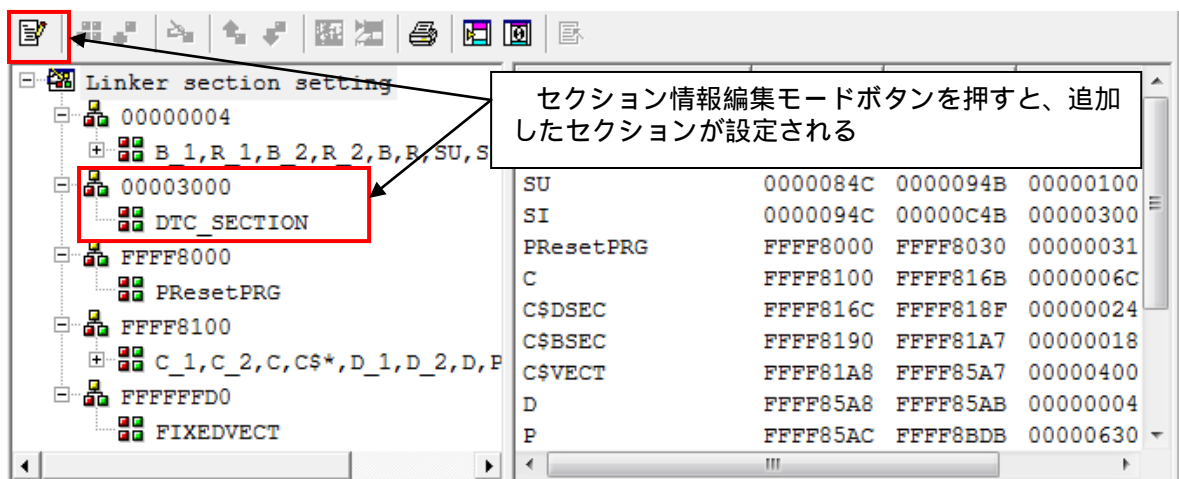
ツールバーから[セクション情報編集モードボタン]をクリックします。



ツールバーから[セクショングループ追加ボタン]をクリックし、セクション開始アドレスとセクショングループ名を設定します。サンプルコードでは、セクション開始アドレスを 3000h、セクショングループ名を DTC_SECTION に設定します。



ツールバーから[セクション情報編集モードボタン]をクリックし、変更したリンカセクション情報を確定させます。



(3) DTC ベクタテーブルの配置

追加した DTC_SECTION に DTC ベクタテーブルを配置します。図 6.2 に DTC ベクタ テーブルを配置するためのサンプルコードと、DTC ベクタテーブルの範囲を示します。セクションについては、最新の RX ファミリ C/C++コンパイラパッケージユーザズマニュアルを参照してください。

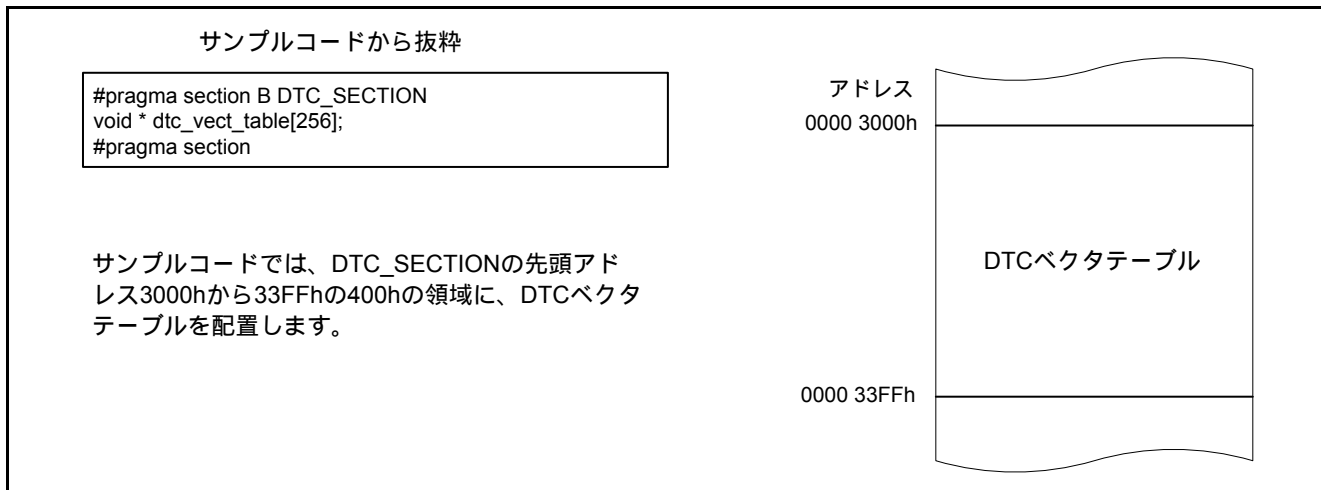


図6.2 DTC ベクタテーブルの配置

(4) DTC 転送情報の先頭アドレスを DTC ベクタテーブルに設定

DTC ベクタテーブルに転送情報の先頭アドレスを設定します。サンプルコードでは、IRQ10 割り込みと IRQ8 割り込み要求発生時に DTC 転送が行われるように転送情報を設定しています。図 6.3 に DTC ベクタテーブルへの DTC 転送情報の設定を示します。

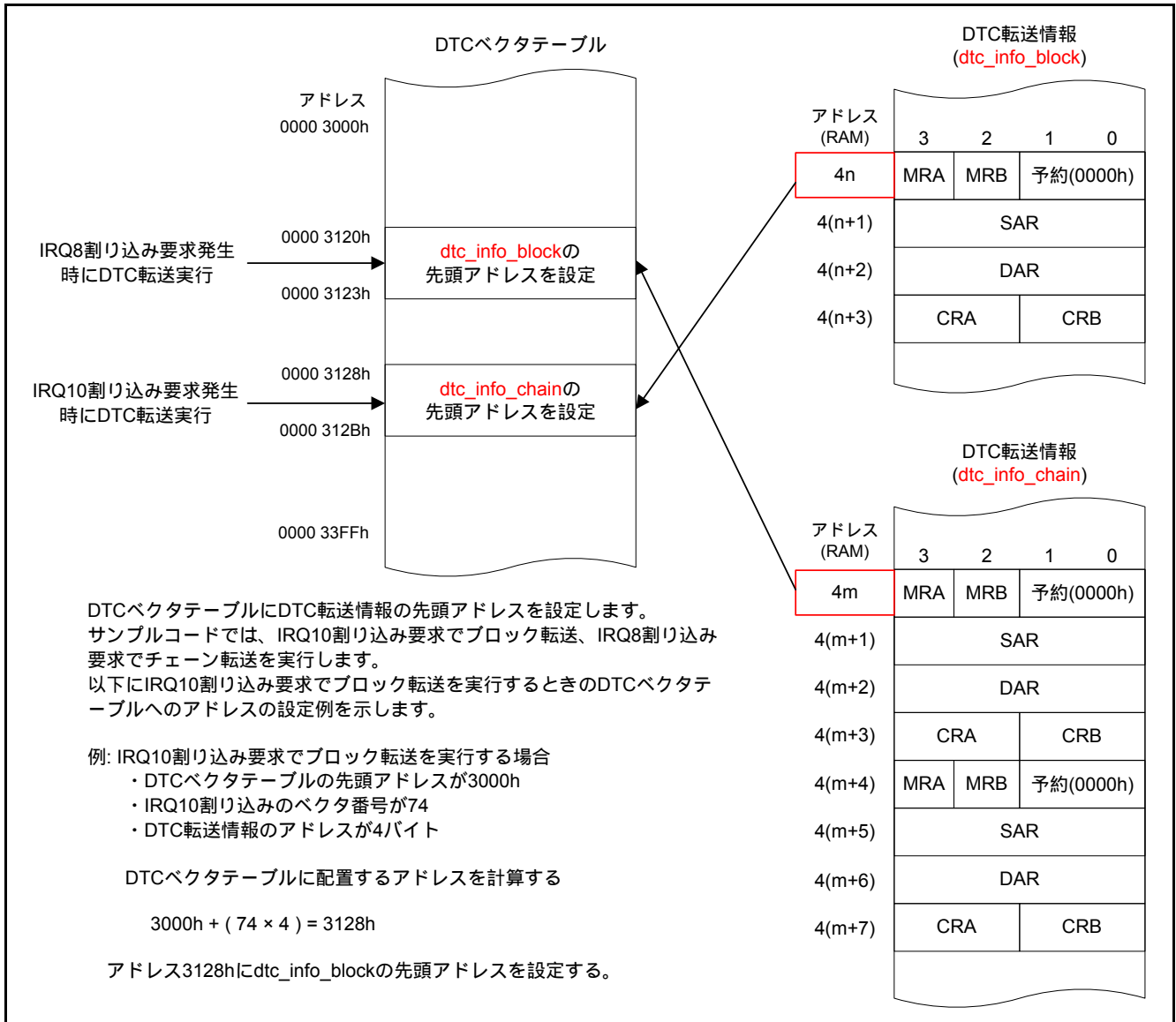


図6.3 DTC ベクタテーブルへの DTC 転送情報の設定

6.1.2 DTC 関連レジスタの再設定

再度 DTC 転送する場合、転送完了後に再設定が必要なレジスタがあります。表 6.1 に DTC 関連レジスタの再設定を、表 6.2 に指定した回数のデータ転送終了後の SAR、DAR の値を示します。

表6.1 再設定する DTC 関連レジスタ

レジスタ名	再設定	内容	
DTCERn (n = 割り込みベクタ番号)	必須	指定した回数のデータ転送が終了したとき(チェーン転送の場合は、最後のチェーン転送の指定した回数のデータ転送が終了したとき)DTCE ビットが“0”(DTC 起動禁止)になるため、再設定する必要があります。	
MRA	任意	前回と同様の転送をする場合は再設定不要です。	
MRB	任意	前回と同様の転送をする場合は再設定不要です。	
SAR、DAR	任意	各転送モードと MRA.SM[1:0]ビット、MRB.DM[1:0]ビットの設定に依存します。詳細は表 6.2 を参照。	
CRA	転送モードに依存	ノーマル転送モード	指定した回数のデータ転送が終了したとき、再設定をする必要があります。
		リピート転送モード	転送回数を変更する場合を除き、再設定の必要はありません。
		ブロック転送モード	ブロックサイズを変更する場合を除き、再設定の必要はありません。
CRB	転送モードに依存	ブロック転送モードのみで使用するレジスタです。指定した回数のデータ転送が終了したとき、再設定をする必要があります。	
DTCVBR	不要	-	
DTCST	不要	-	
DTCADM0D	不要	-	

表6.2 指定した回数のデータ転送終了後の SAR、DAR の値

MRA.MD[1:0]ビットの設定	MRA.SM[1:0]ビットの設定	MRB.DM[1:0]ビットの設定	MRB.DTSビットの設定	SAR の値	DAR の値
ノーマル転送モード	アドレス固定	アドレス固定	-	初期設定値	初期設定値
	アドレスをインクリメント、またはデクリメント	アドレスをインクリメント、またはデクリメント	-	インクリメント、またはデクリメントされた値	インクリメント、またはデクリメントされた値
リピート転送モード	アドレスをインクリメント、またはデクリメント	アドレスをインクリメント、またはデクリメント	SAR をリピート領域に指定	初期設定値	インクリメント、またはデクリメントされた値
	アドレスをインクリメント、またはデクリメント	アドレスをインクリメント、またはデクリメント	DAR をリピート領域に指定	インクリメント、またはデクリメントされた値	初期設定値
ブロック転送モード	アドレスをインクリメント、またはデクリメント	アドレスをインクリメント、またはデクリメント	SAR をブロック領域に指定	初期設定値	インクリメント、またはデクリメントされた値
	アドレスをインクリメント、またはデクリメント	アドレスをインクリメント、またはデクリメント	DAR をブロック領域に指定	インクリメント、またはデクリメントされた値	初期設定値

6.2 ファイル構成

表 6.3にサンプルコードで使用するファイル、表 6.4に標準インクルードファイル、表 6.5に参照する関連アプリケーションノートの関数と設定値を示します。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6.3 サンプルコードで使用するファイル

ファイル名	概要
main.c	メイン処理
r_init_stop_module.c	リセット後に動作している周辺機能の停止
r_init_stop_module.h	r_init_stop_module.c のヘッダファイル
r_init_non_existent_port.c	存在しないポートの初期設定
r_init_non_existent_port.h	r_init_non_existent_port.c のヘッダファイル
r_init_clock.c	クロック初期設定
r_init_clock.h	r_init_clock.c のヘッダファイル
dtc.c	ブロック転送、チェーン転送
dtc.h	dtc.cのヘッダファイル

表6.4 標準インクルードファイル

ファイル名	概要
stdint.h	指定した幅の整数型を宣言してマクロを定義します。
stdbool.h	論理型、および論理値に関するマクロを定義します。
machine.h	RX ファミリ用組み込み関数の形式を定義します。

表6.5 参照する関連アプリケーションノートの関数と設定値

(RX63N グループ、RX631 グループ 初期設定例)

ファイル名	関数	設定値
r_init_stop_module.c	R_INIT_StopModule()	DMAC/DTC、EXDMAC、RAM0、RAM1 はモジュールストップ解除
r_init_stop_module.h	-	
r_init_non_existent_port.c	R_INIT_NonExistentPort()	
r_init_non_existent_port.h	-	176 ピン版を指定
r_init_clock.c	R_INIT_Clock()	
r_init_clock.h	-	サブクロックを使用しない

6.3 オプション設定メモリ

表 6.6 にサンプルコードで使用するオプション設定メモリの状態を示します。必要に応じて、お客様のシステムに最適な値を設定してください。

表6.6 サンプルコードで使用するオプション設定メモリ

シンボル	アドレス	設定値	内容
OFS0	FFFF FF8Fh ~ FFFF FF8Ch	FFFF FFFFh	リセット後、IWDT は停止 リセット後、WDT は停止
OFS1	FFFF FF8Bh ~ FFFF FF88h	FFFF FFFFh	リセット後、電圧監視 0 リセット無効 リセット後、HOCO 発振が無効
MDES	FFFF FF83h ~ FFFF FF80h	FFFF FFFFh	リトルエンディアン

6.4 構造体/共用体一覧

図 6.4にサンプルコードで使用する構造体/共用体を示します。

```
/* **** DTC Transfer information data **** */
#pragma bit_order    left      /* ビットフィールドの並び順指定: 上位ビット側からメンバを割り付ける */
#pragma unpack      /* 構造体メンバの境界調整数指定: メンバの型でアライメントする */
typedef struct
{
    union
    {
        uint32_t LONG;
        struct
        {
            uint32_t MRA_MD      :2;
            uint32_t MRA_SZ      :2;
            uint32_t MRA_SM      :2;
            uint32_t             :2;
            uint32_t MRB_CHNE    :1;
            uint32_t MRB_CHNS    :1;
            uint32_t MRB_DISEL   :1;
            uint32_t MRB_DTS     :1;
            uint32_t MRB_DM      :2;
            uint32_t             :2;
            uint32_t             :16;
        } BIT;
    } MR;
    void * SAR;
    void * DAR;
    struct
    {
        uint32_t CRA:16;
        uint32_t CRB:16;
    } CR;
}dtc_full_t;
#pragma packoption    /* 構造体メンバの境界調整数指定の終了 */
#pragma bit_order    /* ビットフィールドの並び順指定の終了 */
```

図6.4 サンプルコードで使用する構造体/共用体

6.5 変数一覧

表 6.7にstatic 型変数を、表 6.8にconst 型変数を示します。

表6.7 static 型変数

型	変数名	内容	使用関数
static dtc_full_t	dtc_info_block	DTC 転送情報(ブロック転送)	DTC_Init Excep_ICU_IRQ10
static dtc_full_t	dtc_info_chain[2]	DTC 転送情報(チェーン転送)	DTC_Init Excep_ICU_IRQ8
static uint32_t	dtc_block_dar	dtc_info_block の転送先領域	DTC_Init
static uint32_t	dtc_chain0_dar	dtc_info_chain[0]の転送先領域	DTC_Init
static uint32_t	dtc_chain1_dar	dtc_info_chain[1]の転送先領域	DTC_Init
static void*	dtc_vect_table[256]	DTC ベクタテーブル	DTC_Init

表6.8 const 型変数

型	変数名	内容	使用関数
static const uint32_t	dtc_block_sar[15]	dtc_info_block の転送元領域	DTC_Init Excep_ICU_IRQ10
static const uint32_t	dtc_chain0_sar[3]	dtc_info_chain[0]の転送元領域	DTC_Init Excep_ICU_IRQ8
static const uint32_t	dtc_chain1_sar[3]	dtc_info_chain[1]の転送元領域	DTC_Init Excep_ICU_IRQ8

6.6 関数一覧

表 6.9に関数を示します。

表6.9 関数

関数名	概要	記載ファイル
main	メイン処理	main.c
port_init	ポート初期設定	main.c
R_INIT_StopModule	リセット後に動作している周辺機能の停止	r_init_stop_module.c
R_INIT_NonExistentPort	存在しないポートの初期設定	r_init_non_existent_port.c
R_INIT_Clock	クロック初期設定	r_init_clock.c
peripheral_init	周辺機能初期設定	main.c
IRQ_Init	IRQ 初期設定	main.c
DTC_Init	DTC 初期設定	dtc.c
Excep_ICU_IRQ10	IRQ10 割り込み処理	dtc.c
Excep_ICU_IRQ8	IRQ8 割り込み処理	dtc.c

6.7 関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	SW1(IRQ10)と SW2(IRQ8)の入力を待ちます。
引数	なし
リターン値	なし

port_init	
概要	ポート初期設定
ヘッダ	なし
宣言	void port_init(void)
説明	ポートの初期設定を行います。
引数	なし
リターン値	なし

R_INIT_StopModule	
概要	リセット後に動作している周辺機能の停止
ヘッダ	r_init_stop_module.h
宣言	void R_INIT_StopModule(void)
説明	モジュールストップ状態へ遷移する設定を行います。
引数	なし
リターン値	なし
備考	サンプルコードでは、モジュールストップ状態への遷移は行っていません。 本関数の詳細は、アプリケーションノート「RX63N グループ、RX631 グループ 初期設定例 Rev.1.10」を参照してください。

R_INIT_NonExistentPort	
概要	存在しないポートの初期設定
ヘッダ	r_init_non_existent_port.h
宣言	void R_INIT_NonExistentPort(void)
説明	176 ピン未満の製品に対して、存在しないポートの端子に対応するポート方向レジスタの初期設定を行います。
引数	なし
リターン値	なし
備考	サンプルコードでは、176 ピン版(PIN_SIZE=176)に設定しています。 本関数をコールした後に、存在しないポートを含む PDR、PODR レジスタへバイト単位で書き込む場合、存在しないポートの方向制御ビットには“1”、ポート出力データ格納ビットには“0”を設定してください。 本関数の詳細は、アプリケーションノート「RX63N グループ、RX631 グループ 初期設定例 Rev.1.10」を参照してください。

R_INIT_Clock	
概要	クロック初期設定
ヘッダ	r_init_clock.h
宣言	void R_INIT_Clock(void)
説明	クロックの初期設定を行います。
引数	なし
リターン値	なし
備考	サンプルコードでは、システムクロックを PLL とし、サブクロックを使用しない処理を選択しています。 本関数の詳細は、アプリケーションノート「RX63N グループ、RX631 グループ 初期設定例 Rev.1.10」を参照してください。
peripheral_init	
概要	周辺機能初期設定
ヘッダ	なし
宣言	void peripheral_init(void)
説明	使用する周辺機能の初期設定を行います。
引数	なし
リターン値	なし
IRQ_Init	
概要	IRQ 初期設定
ヘッダ	なし
宣言	static void IRQ_Init(void)
説明	IRQ10 と IRQ8 の初期設定を行います。
引数	なし
リターン値	なし
dtc_init	
概要	DTC 初期設定
ヘッダ	なし
宣言	static void dtc_init(void)
説明	DTC の初期設定を行います。
引数	なし
リターン値	なし
Excep_ICU_IRQ10	
概要	IRQ10 割り込み処理
ヘッダ	なし
宣言	static void Excep_ICU_IRQ10 (void)
説明	再度、ブロック転送モードの DTC 転送を行うためにレジスタを再設定します。
引数	なし
リターン値	なし

Excep_ICU_IRQ8

概要	IRQ8 割り込み処理
ヘッダ	なし
宣言	static void Excep_ICU_IRQ8 (void)
説明	再度、チェーン転送の DTC 転送を行うためにレジスタを再設定します。
引数	なし
リターン値	なし

6.8 フローチャート

6.8.1 メイン処理

図 6.5にメイン処理のフローチャートを示します。

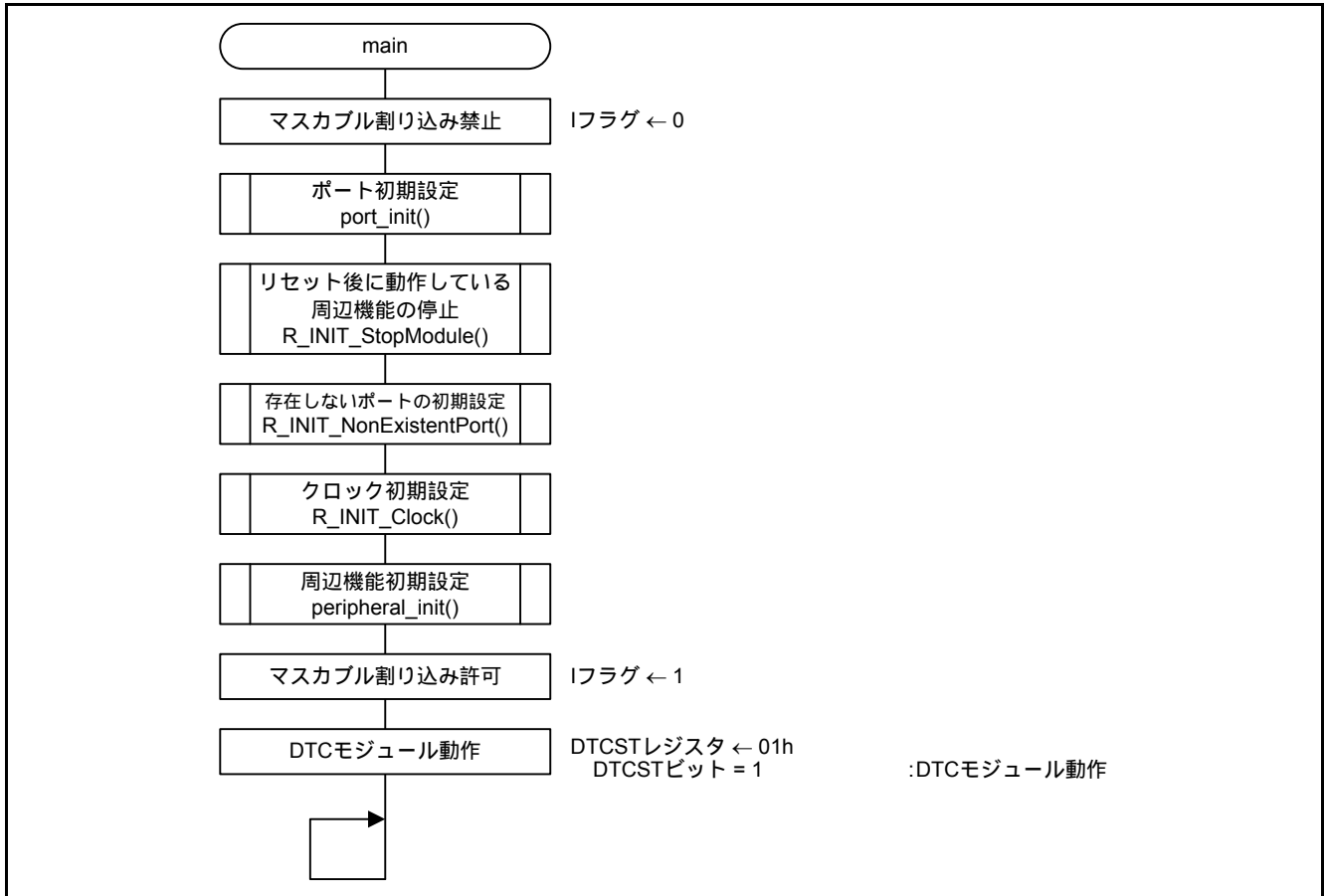


図6.5 メイン処理

6.8.2 ポート初期設定

図 6.6にポート初期設定のフローチャートを示します。

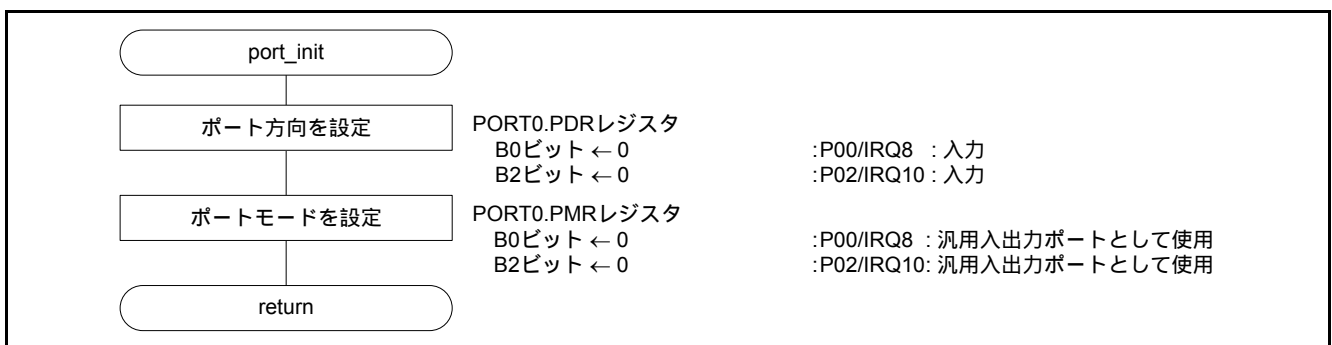


図6.6 ポート初期設定

6.8.3 周辺機能初期設定

図 6.7 に周辺機能初期設定のフローチャートを示します。

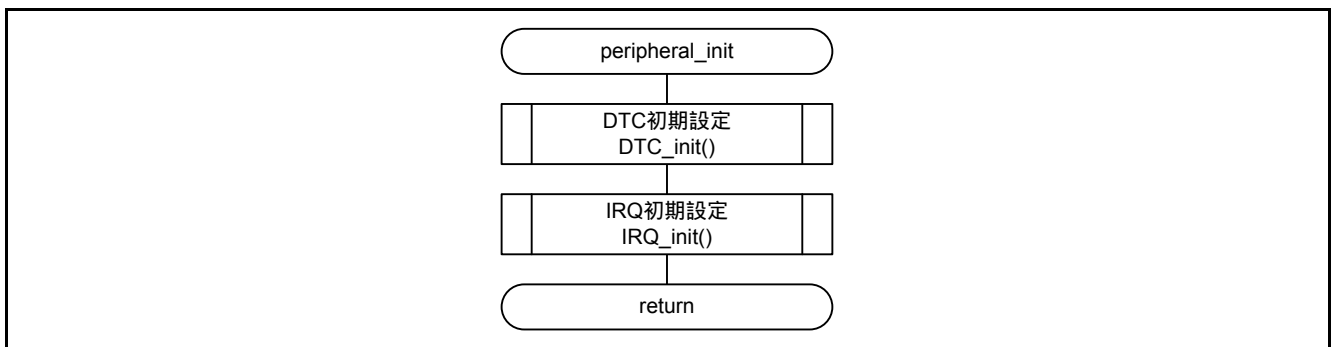


図6.7 周辺機能初期設定

6.8.4 IRQ 初期設定

図 6.8、図 6.9にIRQ 初期設定のフローチャートを示します。

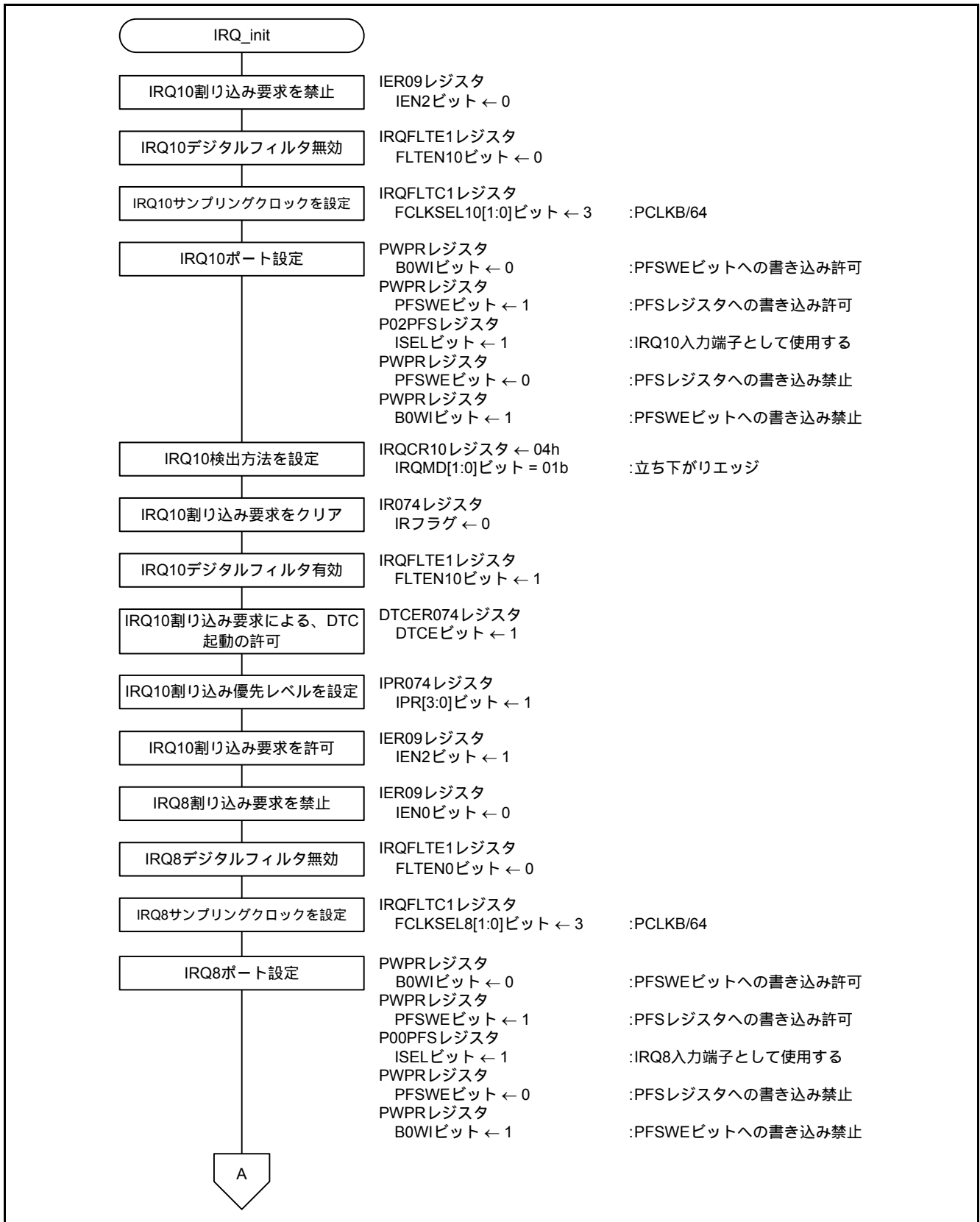


図6.8 IRQ 初期設定(1)

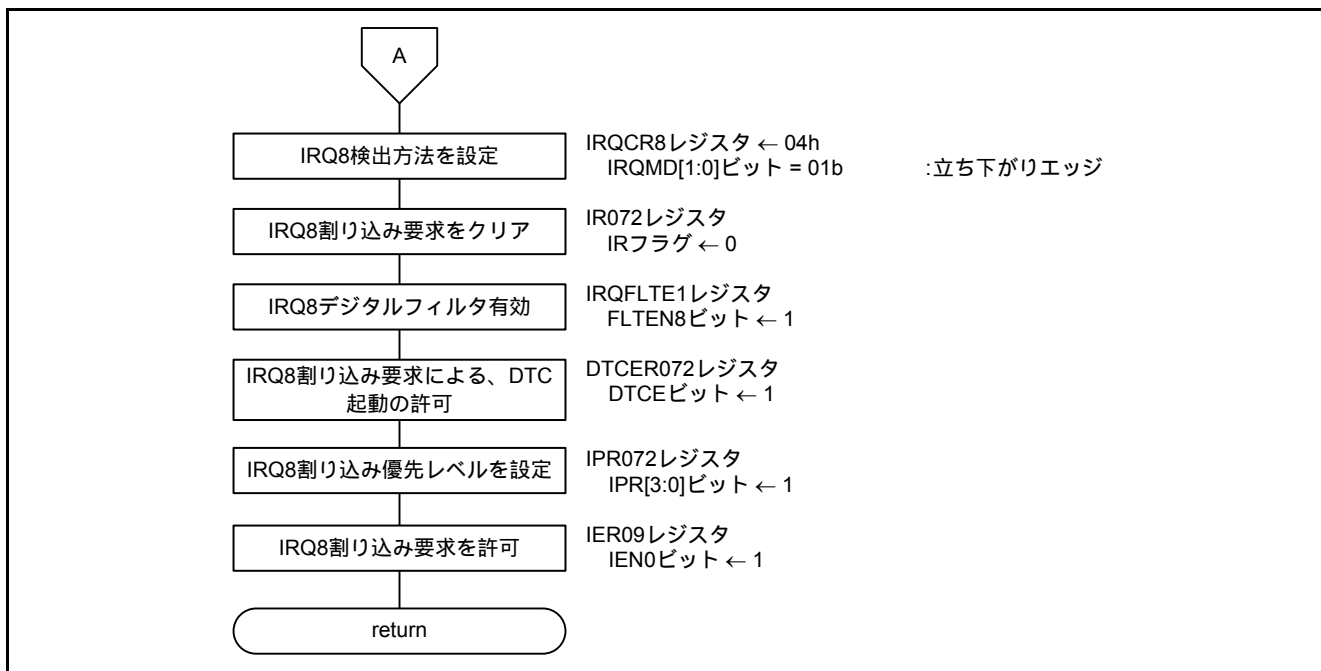


図6.9 IRQ 初期設定(2)

6.8.5 DTC 初期設定

図 6.10にDTC 初期設定のフローチャートを示します。

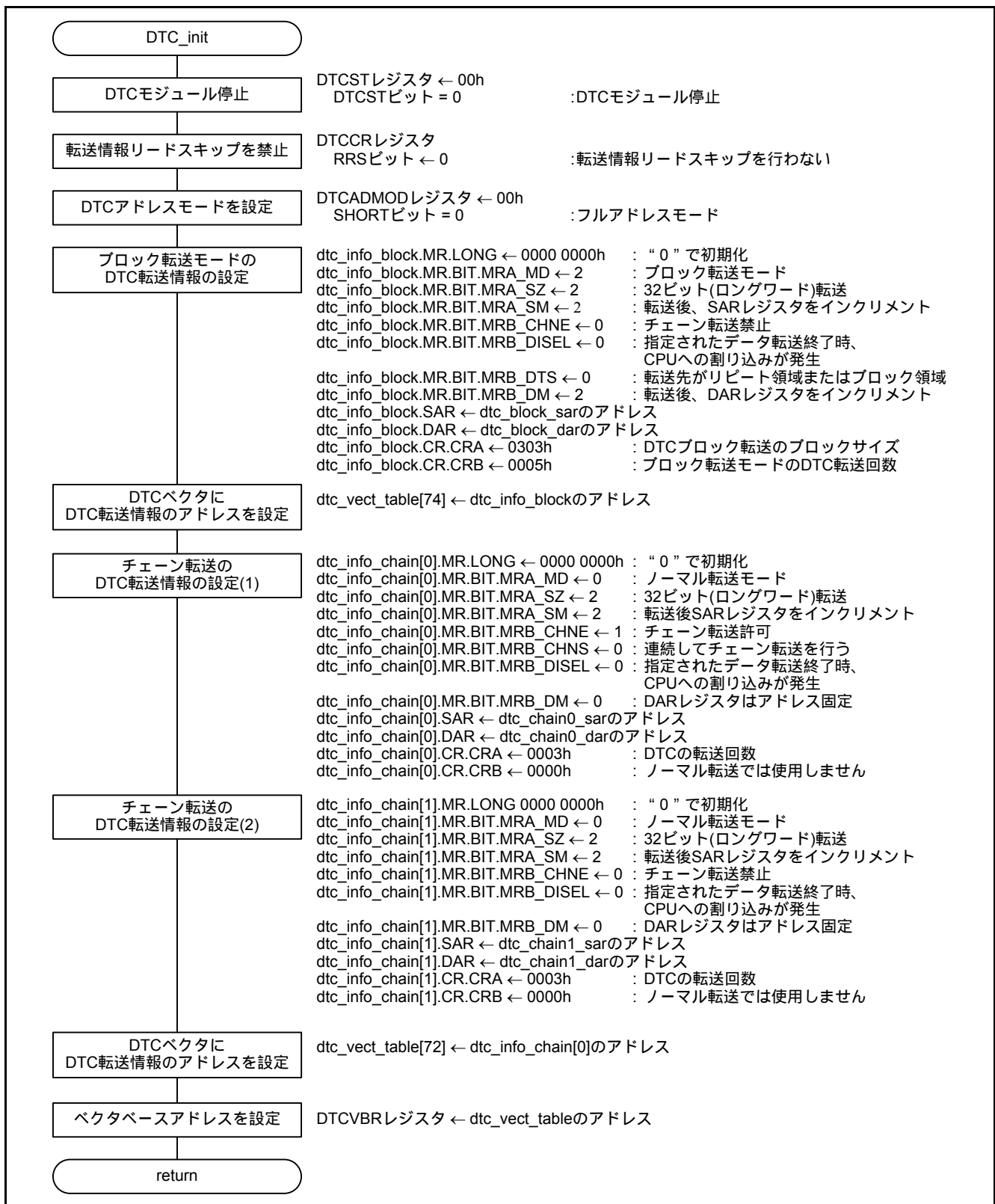


図6.10 DTC 初期設定

6.8.6 IRQ10 割り込み処理

図 6.11にIRQ10 割り込み処理のフローチャートを示します。

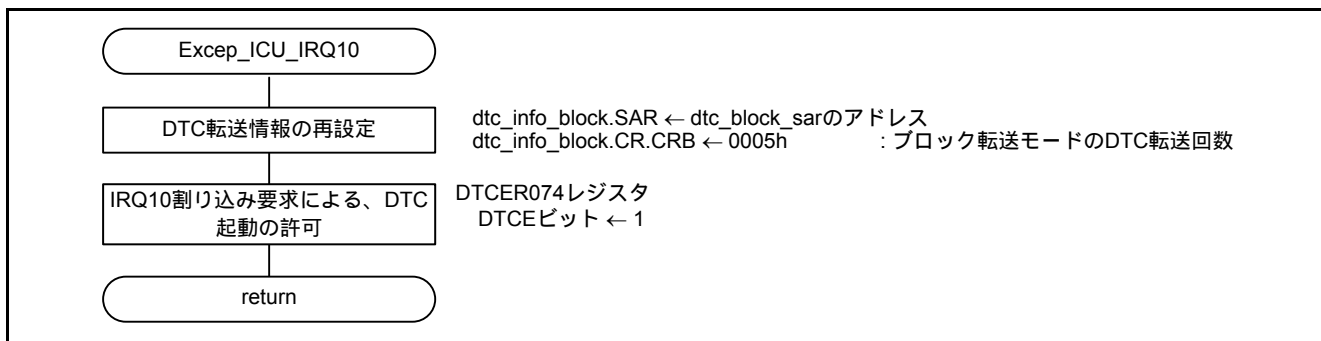


図6.11 IRQ10 割り込み処理

6.8.7 IRQ8 割り込み処理

図 6.12にIRQ8 割り込み処理のフローチャートを示します。

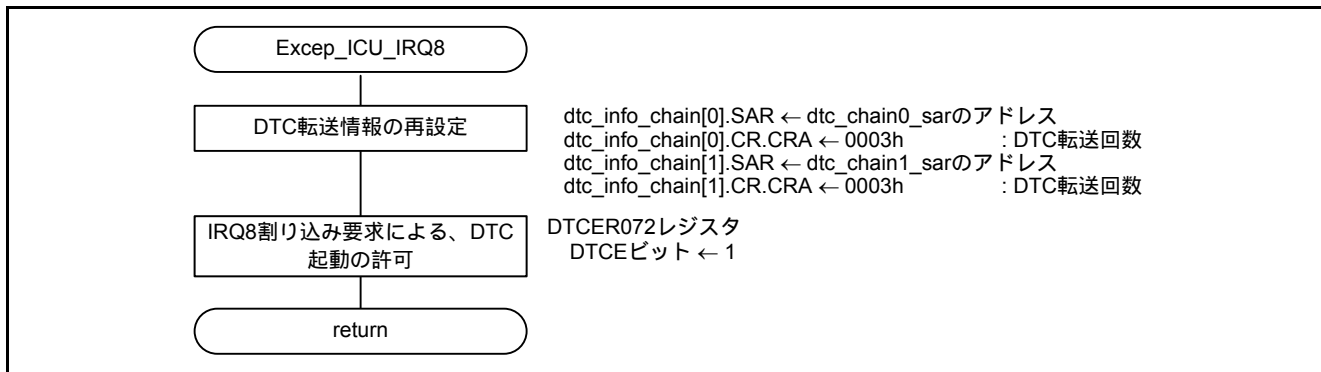


図6.12 IRQ8 割り込み処理

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

ユーザーズマニュアル：ハードウェア

RX ファミリ ユーザーズマニュアル ハードウェア編 Rev.1.80 (R01UH0041JJ)
(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。)

ユーザーズマニュアル：開発環境

RX ファミリ C/C++コンパイラパッケージ V.1.01 ユーザーズマニュアル Rev.1.00 (R20UT0570JJ)
(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。)

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/contact/>

改訂記録	RX ファミリ DTC のブロック転送モードとチェーン転送の設定例
------	-----------------------------------

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2014.07.01	—	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うと、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い合わせください。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用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 〒100-0004 千代田区大手町2-6-2（日本ビル）

■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/>