
RX210、RX21A、RX220 グループ

R01AN1017JJ0101

Rev.1.01

2014.07.01

DTC ベクタテーブルの効率的な配置方法

要旨

本アプリケーションノートでは、RX210、RX21A、RX220 グループのデータトランスファコントローラ(以下、DTC)の DTC ベクタテーブルを効率的に配置する方法について説明します。

対象デバイス

RX210、RX21A、RX220 グループ

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	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
5. ソフトウェア説明	7
5.1 動作概要	8
5.2 セクション構成.....	9
5.3 ファイル構成	9
5.4 オプション設定メモリ	10
5.5 定数一覧	11
5.6 構造体/共用体一覧.....	12
5.7 変数一覧	12
5.8 関数一覧	13
5.9 関数仕様	14
5.10 フローチャート.....	16
5.10.1 メイン処理	16
5.10.2 ポート初期設定	16
5.10.3 周辺機能初期設定	17
5.10.4 消費電力低減機能初期設定	17
5.10.5 CMT 初期設定	18
5.10.6 DTC 初期設定	19
6. RX21A、RX220 グループ 初期設定例 アプリケーションノートとの組み合わせ方.....	20
7. サンプルコード.....	21
8. 参考ドキュメント	21

1. 仕様

DTC を使用して、500ms 周期で PORT1 から LED に出力するパターンを変更し、250ms 周期で PORTA から出力するパターンを変更します。

本アプリケーションノートでは、DTC ベクタテーブルの配置方法について、表 1.1 DTC ベクタテーブルの配置方法一覧に示す 2 種類の方法を説明します。

表 1.2 に使用する周辺機能と用途を、図 1.1 に接続図を示します。また、図 1.2 に DTC ベクタテーブルを ROM 領域に配置した場合(サンプルコード 1)のメモリマップ、図 1.3 に DTC ベクタテーブルを RAM 領域に配置した場合(サンプルコード 2)のメモリマップを示します。

表1.1 DTC ベクタテーブルの配置方法一覧

種別	概要	特徴
サンプルコード 1	DTC ベクタテーブルを ROM 領域に配置	すべての DTC ベクタを設定(4×256 バイト使用) DTC ベクタテーブルはセクションを新規に追加して配置
サンプルコード 2	DTC ベクタテーブルを RAM 領域に配置	必要な DTC ベクタのみ設定 空き領域に DTC 転送情報を配置 DTC ベクタテーブルは#pragma で絶対番地アドレスを指定

表1.2 使用する周辺機能と用途

周辺機能	用途
DTC	出力データをポートに転送
コンペアマッチタイマ チャンネル 0(以下、CMT0)	DTC の起動要因(500ms 周期。LED に出力するパターンを変更)
コンペアマッチタイマ チャンネル 3(以下、CMT3)	DTC の起動要因(250ms 周期。PORTA から出力するパターンを変更)

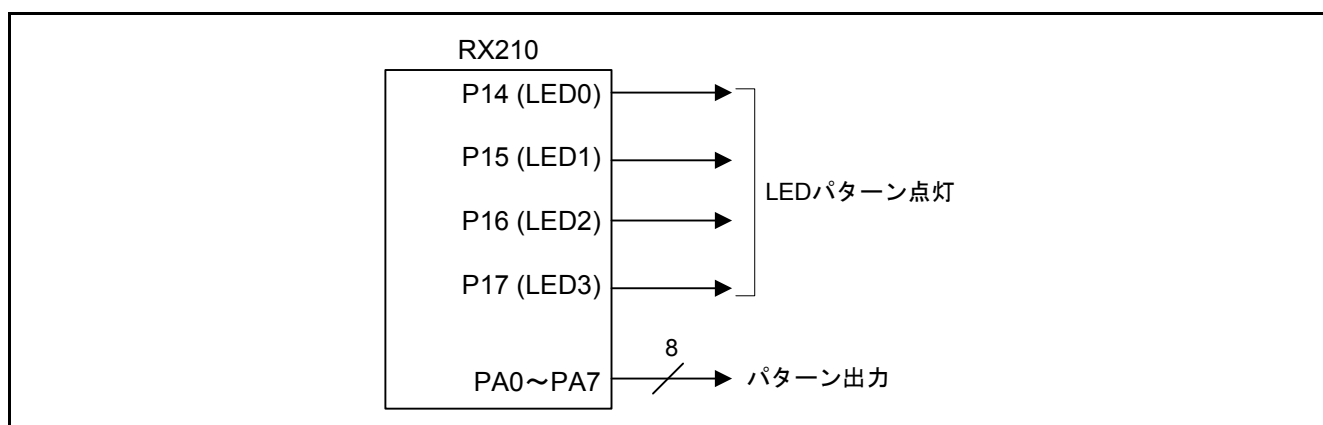


図1.1 接続図

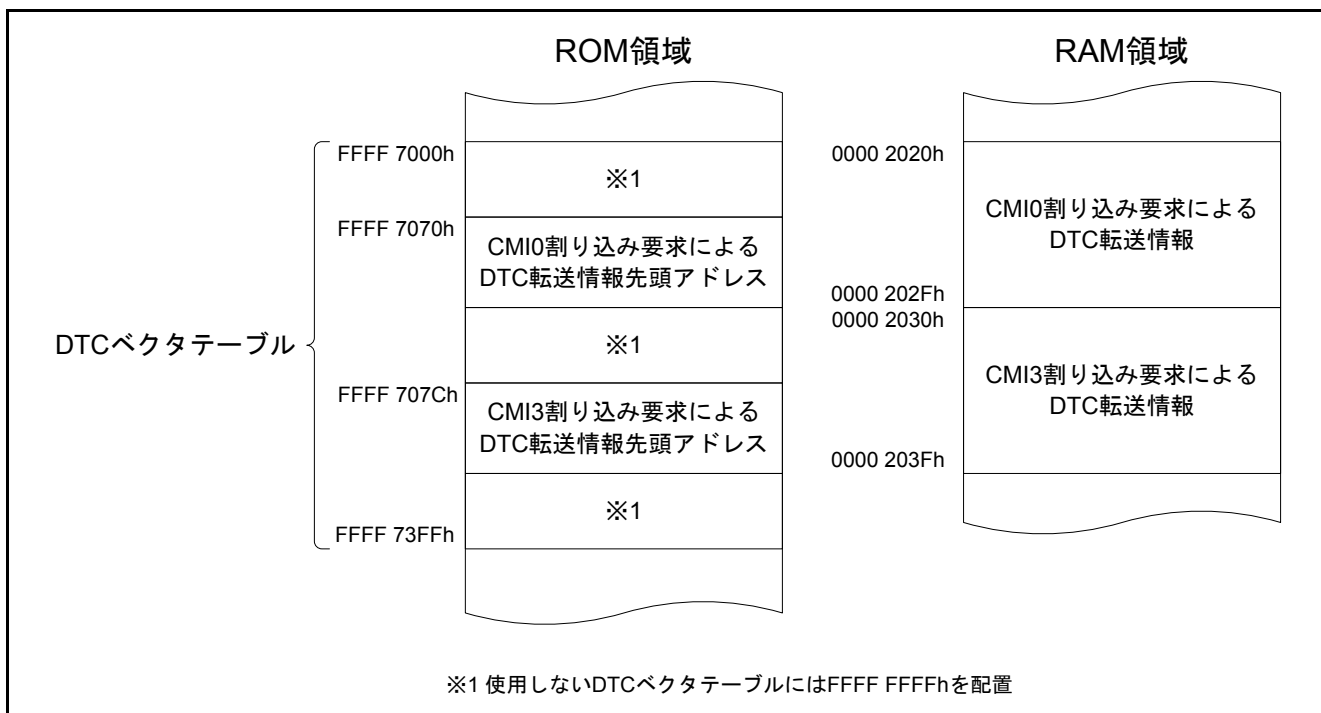


図1.2 DTC ベクタテーブルを ROM 領域に配置した場合(サンプルコード 1)のメモリマップ

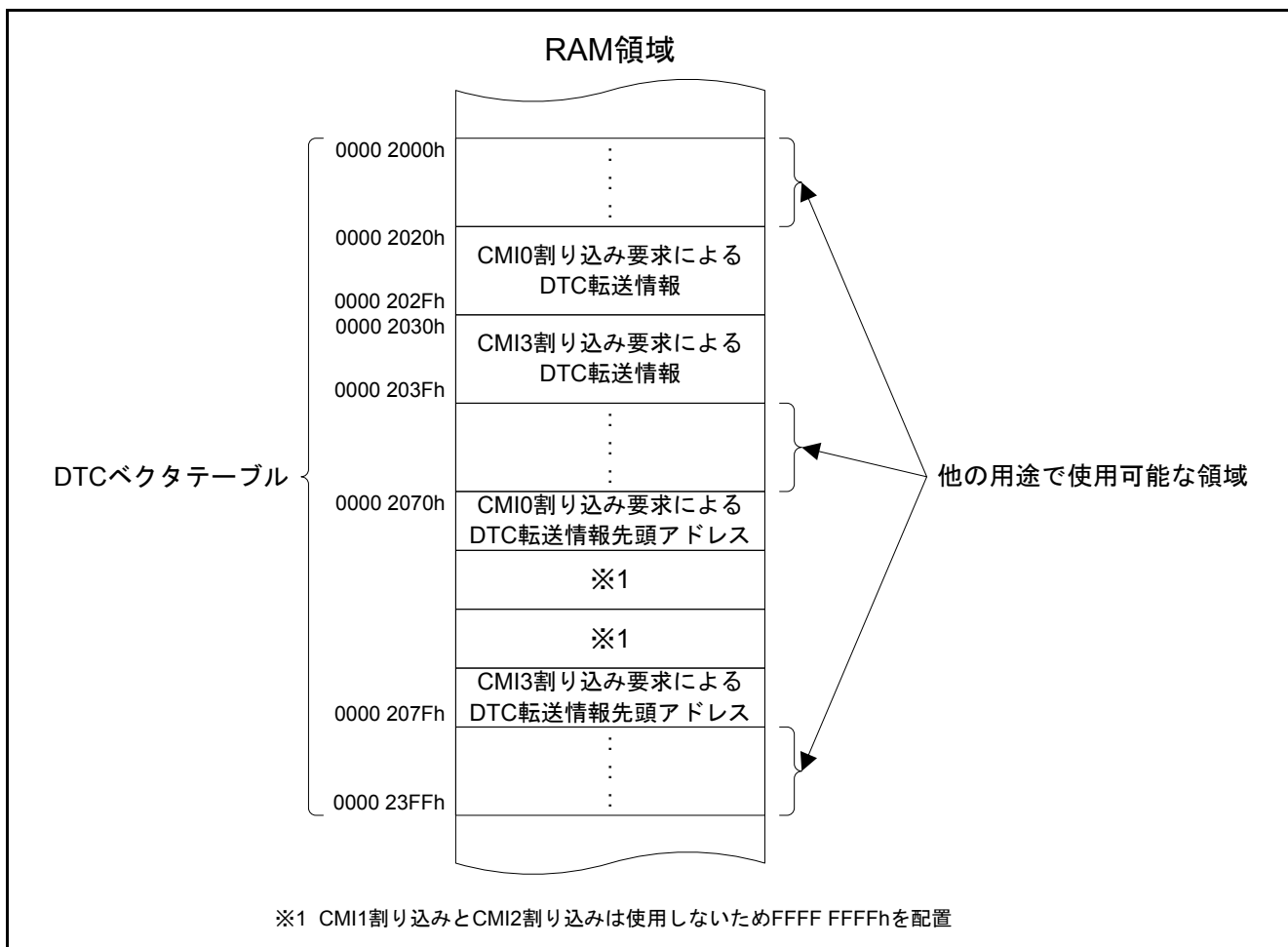


図1.3 DTC ベクタテーブルを RAM 領域に配置した場合(サンプルコード 2)のメモリマップ

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	R5F52108ADFP (RX210 グループ)
動作周波数	<ul style="list-style-type: none"> メインクロック: 20MHz PLL: 100MHz (メインクロック 2 分周 10 通倍) システムクロック (ICLK): 50MHz (PLL 2 分周) 周辺モジュールクロック B (PCLKB): 25MHz (PLL 4 分周)
動作電圧	5.0V
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=rx200 -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 1.2A
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 RX210 (製品型名: R0K505210C000BE)

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。

- RX210グループ 初期設定例 Rev.2.00 (R01AN1002JJ)
- RX21A グループ 初期設定例 Rev.1.10 (R01AN1486JJ)
- RX220 グループ 初期設定例 Rev.1.10 (R01AN1494JJ)

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. 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。

使用端子は 100 ピン版の製品を想定しています。100 ピン版未満の製品を使用する場合は、使用する製品に合わせて端子を選択してください。

表4.1 使用端子と機能

端子名	入出力	内容
P14	出力	LED0 点灯/消灯用データ出力
P15	出力	LED1 点灯/消灯用データ出力
P16	出力	LED2 点灯/消灯用データ出力
P17	出力	LED3 点灯/消灯用データ出力
PA0~PA7	出力	パターン出力

5. ソフトウェア説明

あらかじめ ROM 領域に LED 出力パターン của データテーブル、および PORTA 出力パターン của データテーブルを配置しておきます。

CMT0 のコンペアマッチ割り込み(CMI0)要求が発生すると、DTC が起動して、PORT1 に LED 出力パターンを転送します。その後、CMI0 割り込み要求が発生するごとにデータテーブルの値を順次転送します。

また、CMT3 のコンペアマッチ割り込み(CMI3)要求が発生すると、DTC が起動して、PORTA に PORTA 出力パターンを転送します。その後、CMI3 割り込み要求が発生するごとにデータテーブルの値を順次転送します。

使用する周辺機能の設定を以下に示します。

<CMT>

- カウントクロック : PCLKB/512
- CMT0 コンペアマッチ周期 : 500ms
- CMT3 コンペアマッチ周期 : 250ms

<DTC>

- 起動要因 : CMT0 および CMT3 のコンペアマッチ割り込み
- DTC アドレスモード : フルアドレスモード

[CMI0 割り込み要求による DTC 転送設定]

- 転送モード : リピート転送モード
- 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 : ROM 領域 (LED 出力データ)
- 転送先アドレッシングモード : DAR レジスタはアドレス固定
- 転送先アドレス : PORT1(LED)
- データ転送単位 : 8 ビット
- 転送回数 : 4 回
- チェーン転送 : 禁止

[CMI3 割り込み要求による DTC 転送設定]

- 転送モード : リピート転送モード
- 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 : ROM 領域 (PORTA 出力データ)
- 転送先アドレッシングモード : DAR レジスタはアドレス固定
- 転送先アドレス : PORTA
- データ転送単位 : 8 ビット
- 転送回数 : 8 回
- チェーン転送 : 禁止

5.1 動作概要

- (1) 初期設定後、CMSTR0.STR0 ビットおよび、CMSTR1.STR3 ビットを“1”にすると、CMT0、CMT3 のカウントが始まります。
- (2) CMI3 割り込み要求が発生すると、DTC 転送にて、PORTA.PODR レジスタに PORTA 出力データの 1 番目(01h)を書き込みます。
以降、CMI3 割り込み要求が発生するごとに、DTC 転送にて、PORTA.PODR レジスタに PORTA 出力データが順次転送されます。
- (3) CMI0 割り込み要求が発生すると、DTC 転送にて、PORT1.PODR レジスタに LED 出力データの 1 番目(LED0 が点灯)を書き込みます。
以降、CMI0 割り込み要求が発生するごとに、DTC 転送にて、PORT1.PODR レジスタに LED 出力データが順次転送され、LED に出力されます。
- (4) CMI3 割り込み要求による DTC 転送が転送回数分(8 回)発生したあと、再度 CMI3 割り込み要求が発生すると、PORTA 出力データの 1 番目から DTC 転送を行います。
- (5) CMI0 割り込み要求による DTC 転送が転送回数分(4 回)発生したあと、再度 CMI0 割り込み要求が発生すると、LED 出力データの 1 番目から DTC 転送を行います。

図 5.1 に動作タイミング図を示します。図の(1)~(5)は、上記(1)~(5)の動作タイミングを示します。

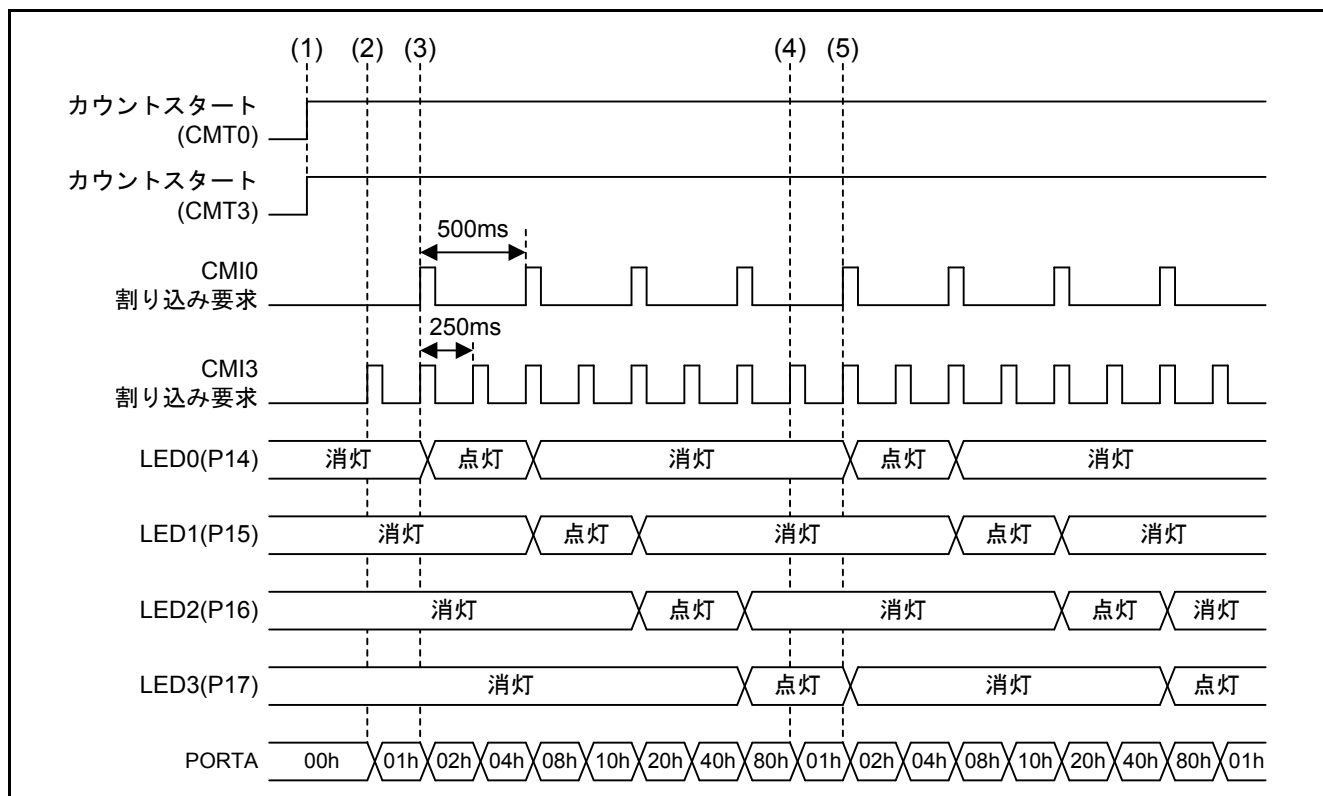


図5.1 動作タイミング図

<システムに組み込む際の注意>

本アプリケーションノートのサンプルコードを、実際のシステムに組み込んで使用する際、次のような現象に注意してください。

- DTC のデータ転送単位が 8 ビットなので、CMI0 割り込み要求による DTC 転送にて、PORT1.PODR レジスタに LED0~LED3(P14~P17)への出力データを書き込む際、P10~P13 に対しても“0”が書き込まれます。そのため、P10~P13 は出力ポートとして使用しないでください。

5.2 セクション構成

表 5.1にサンプルコード 1 で変更するセクション情報を示します。

セクションの追加/変更および削除の方法は、最新の RX ファミリー C/C++コンパイラパッケージ ユーザーズマニュアルを参照してください。

表5.1 サンプルコード 1 で変更するセクション情報

セクション名	変更	アドレス	内容
DTC_SECTION	追加	FFFF 7000h	DTC ベクタテーブル(ROM 領域)

5.3 ファイル構成

表 5.2にサンプルコード 1 で使用するファイルを、表 5.3にサンプルコード 2 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5.2 サンプルコード 1 で使用するファイル

ファイル名	概要	備考
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_vector_table.c	DTC ベクタテーブル	

表5.3 サンプルコード 2 で使用するファイル

ファイル名	概要	備考
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	

5.4 オプション設定メモリ

表 5.4にサンプルコード 1,2 共通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5.4 サンプルコード 1,2 共通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

5.5 定数一覧

表 5.5 にサンプルコード 1 で使用する定数を、表 5.6 にサンプルコード 2 で使用する定数を示します。

表5.5 サンプルコード 1 で使用する定数

定数名	設定値	内容
P_LED	PORT1.PODR.BYTE	LED 出力端子
LED_OFF	F0h	LED0,1,2,3 全消灯
LED0_ON	E0h	LED0 点灯、LED1,2,3 消灯
LED1_ON	D0h	LED1 点灯、LED0,2,3 消灯
LED2_ON	B0h	LED2 点灯、LED0,1,3 消灯
LED3_ON	70h	LED3 点灯、LED0,1,2 消灯
PD_LED	PORT1.PDR.BYTE	LED 出力端子の方向制御ビット
LED_OUTPUT	F0h	LED 出力設定
ADDR_CMT0_DTC_TBL	0000 2020h	CMI0 割り込み要求による DTC の転送情報先頭番地
ADDR_CMT3_DTC_TBL	0000 2030h	CMI3 割り込み要求による DTC の転送情報先頭番地

表5.6 サンプルコード 2 で使用する定数

定数名	設定値	内容
P_LED	PORT1.PODR.BYTE	LED 出力端子
LED_OFF	F0h	LED0,1,2,3 全消灯
LED0_ON	E0h	LED0 点灯、LED1,2,3 消灯
LED1_ON	D0h	LED1 点灯、LED0,2,3 消灯
LED2_ON	B0h	LED2 点灯、LED0,1,3 消灯
LED3_ON	70h	LED3 点灯、LED0,1,2 消灯
PD_LED	PORT1.PDR.BYTE	LED 出力端子の方向制御ビット
LED_OUTPUT	F0h	LED 出力設定
ADDR_CMT0_DTC_TBL	0000 2020h	CMI0 割り込み要求による DTC の転送情報先頭番地
ADDR_CMT3_DTC_TBL	0000 2030h	CMI3 割り込み要求による DTC の転送情報先頭番地
ADDR_DTC_VECT_TOP	0000 2000h	DTC ベクタアドレスの先頭番地
ADDR_CMT0_DTC_VECT	0000 2070h	CMT0(CMI0)の DTC ベクタアドレス
ADDR_CMT1_DTC_VECT	0000 2074h	CMT1(CMI1)の DTC ベクタアドレス
ADDR_CMT2_DTC_VECT	0000 2078h	CMT2(CMI2)の DTC ベクタアドレス
ADDR_CMT3_DTC_VECT	0000 207Ch	CMT3(CMI3)の DTC ベクタアドレス

5.6 構造体/共用体一覧

図 5.2にサンプルコード 1,2 共通で使用する構造体/共用体を示します。

ビットフィールドのメンバは#pragma のビットフィールドの並び順指定の機能を使用して上位ビットから割り付けています。詳細は最新の RX ファミリー C/C++コンパイラパッケージ ユーザーズマニュアルを参照してください。

```

/* **** DTC Transfer information data **** */
#pragma bit_order left
#pragma unpack
struct st_dtc_full{
    union{
        unsigned long LONG;
        struct{
            unsigned long MRA_MD          :2;
            unsigned long MRA_SZ          :2;
            unsigned long MRA_SM          :2;
            unsigned long                 :2;
            unsigned long MRB_CHNE        :1;
            unsigned long MRB_CHNS        :1;
            unsigned long MRB_DISEL       :1;
            unsigned long MRB_DTS         :1;
            unsigned long MRB_DM          :2;
            unsigned long                 :2;
            unsigned long                 :16;
        }BIT;
    }MR;
    void * SAR;
    void * DAR;
    struct{
        unsigned long CRA:16;
        unsigned long CRB:16;
    }CR;
};
#pragma bit_order
#pragma packoption

```

図5.2 サンプルコード 1,2 共通で使用する構造体/共用体

5.7 変数一覧

表 5.7にサンプルコード 1 で使用するグローバル変数を、表 5.8にサンプルコード 2 で使用するグローバル変数を、表 5.9にサンプルコード 1 で使用する const 型変数を、表 5.10にサンプルコード 2 で使用する const 型変数を示します。

表5.7 サンプルコード 1 で使用するグローバル変数

型	変数名	内容	使用関数
st_dtc_full	cmt0_dtc_tbl	CMI0 割り込み要求による DTC の転送 情報テーブル	dtc_init
st_dtc_full	cmt3_dtc_tbl	CMI3 割り込み要求による DTC の転送 情報テーブル	dtc_init

表5.8 サンプルコード2で使用するグローバル変数

型	変数名	内容	使用関数
st_dtc_full	cmt0_dtc_tbl	CMI0 割り込み要求による DTC の転送情報テーブル	dtc_init
st_dtc_full	cmt3_dtc_tbl	CMI3 割り込み要求による DTC の転送情報テーブル	dtc_init
void *	cmt0_dtc_vector	CMT0(CMI0)の DTC ベクタ	dtc_init
void *	cmt1_dtc_vector	CMT1(CMI1)の DTC ベクタ	dtc_init
void *	cmt2_dtc_vector	CMT2(CMI2)の DTC ベクタ	dtc_init
void *	cmt3_dtc_vector	CMT3(CMI3)の DTC ベクタ	dtc_init

表5.9 サンプルコード1で使用する const 型変数

型	変数名	内容	使用関数
unsigned char	dtc_output_data0	LED 出力データ(CMI0 割り込み要求による DTC 転送元)	dtc_init
unsigned char	dtc_output_data1	PORTA 出力データ(CMI3 割り込み要求による DTC 転送元)	dtc_init
void *	Relocatable_Vectors	DTC ベクタテーブル	dtc_init

表5.10 サンプルコード2で使用する const 型変数

型	変数名	内容	使用関数
unsigned char	dtc_output_data0	LED 出力データ(CMI0 割り込み要求による DTC 転送元)	dtc_init
unsigned char	dtc_output_data1	PORTA 出力データ(CMI3 割り込み要求による DTC 転送元)	dtc_init

5.8 関数一覧

表 5.11にサンプルコード 1,2 共通で使用する関数を示します。

表5.11 サンプルコード 1,2 共通で使用する関数

関数名	概要
main	メイン処理
port_init	ポート初期設定
R_INIT_StopModule	リセット後に動作している周辺機能の停止
R_INIT_NonExistentPort	存在しないポートの初期設定
R_INIT_Clock	クロック初期設定
peripheral_init	周辺機能初期設定
lpc_init	消費電力低減機能初期設定
cmt_init	CMT 初期設定
dtc_init	DTC 初期設定

5.9 関数仕様

サンプルコード 1,2 共通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	初期設定後、CMT のカウントを開始します。
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、各グループのアプリケーションノート「初期設定例」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	100 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、100 ピン版(PIN_SIZE=100)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、各グループのアプリケーションノート「初期設定例」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、各グループのアプリケーションノート「初期設定例」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

lpc_init	
概要	消費電力低減機能初期設定
ヘッダ	なし
宣言	void lpc_init(void)
説明	使用する周辺機能のモジュールストップ状態を解除します。
引数	なし
リターン値	なし

cmt_init	
概要	CMT 初期設定
ヘッダ	なし
宣言	void cmt_init(void)
説明	CMT を使用するための設定を行います。
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	void dtc_init(void)
説明	DTC を使用するための設定を行います。
引数	なし
リターン値	なし

5.10 フローチャート

5.10.1 メイン処理

図 5.3にメイン処理のフローチャートを示します。

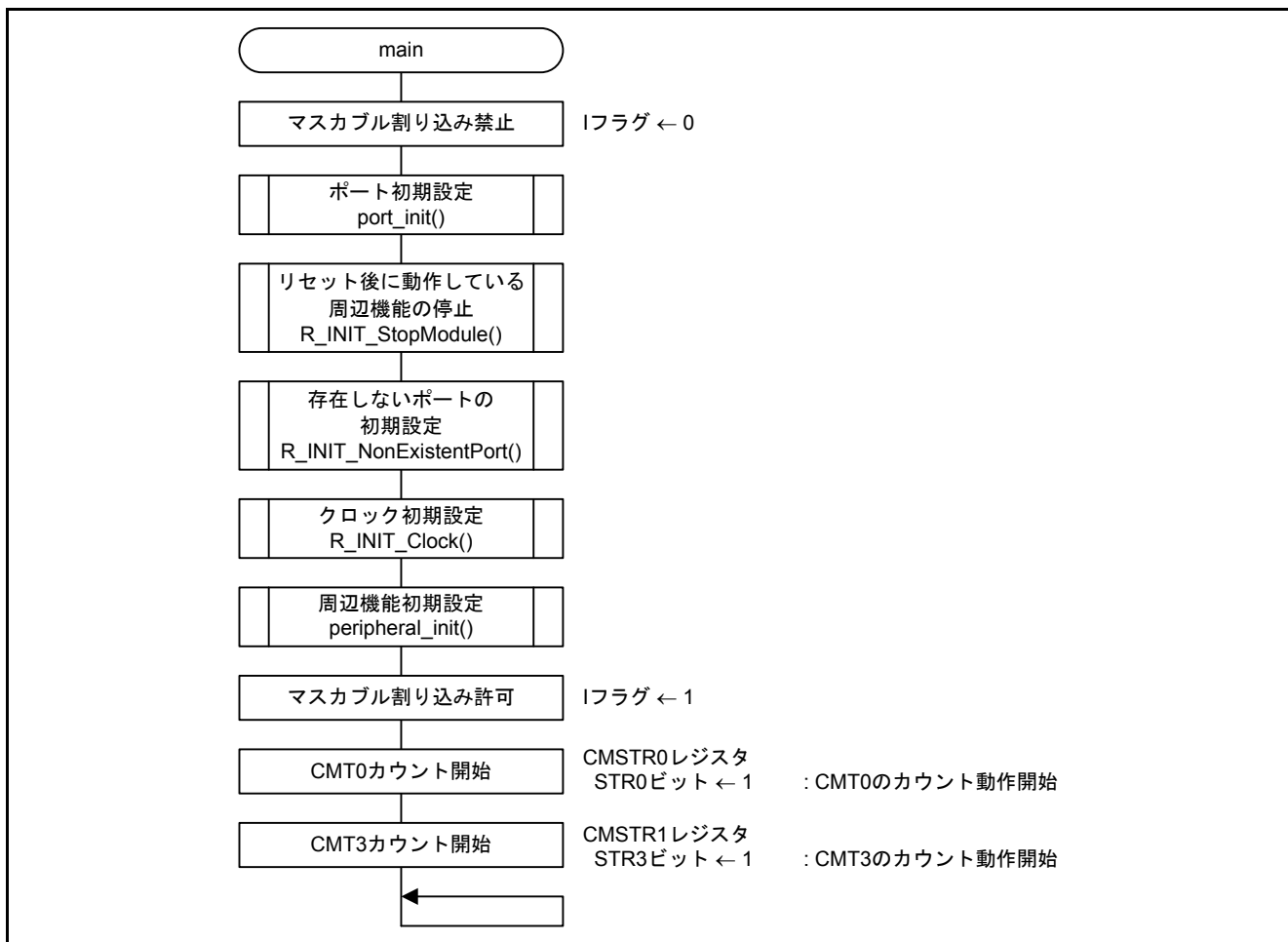


図5.3 メイン処理

5.10.2 ポート初期設定

図 5.4にポート初期設定のフローチャートを示します。

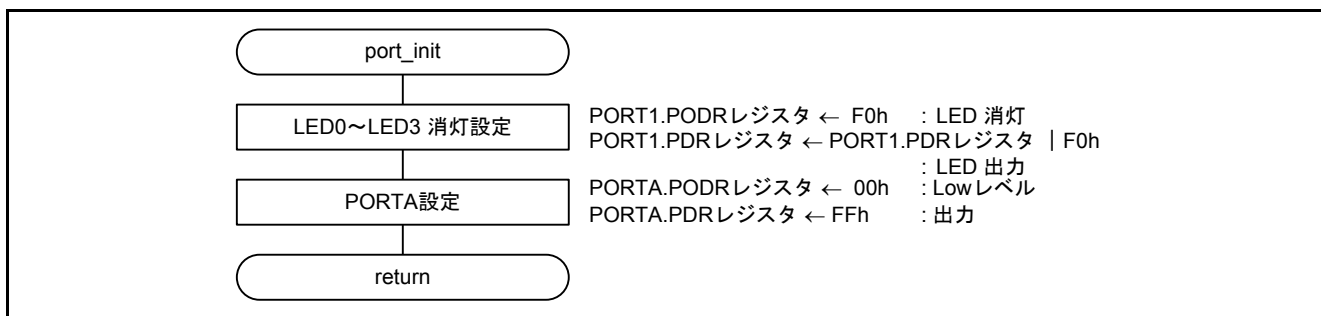


図5.4 ポート初期設定

5.10.3 周辺機能初期設定

図 5.5に周辺機能初期設定のフローチャートを示します。

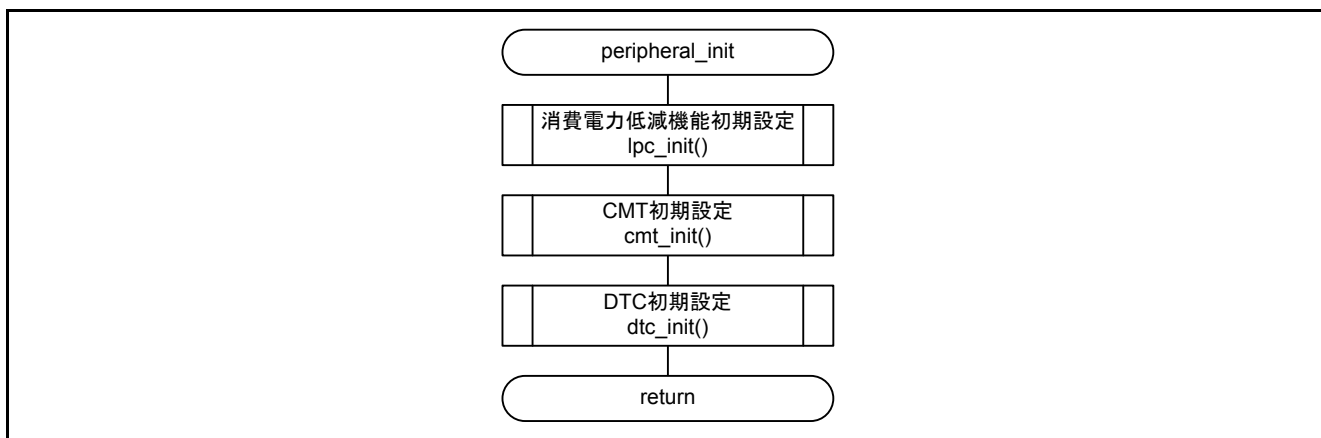


図5.5 周辺機能初期設定

5.10.4 消費電力低減機能初期設定

図 5.6に消費電力低減機能初期設定のフローチャートを示します。

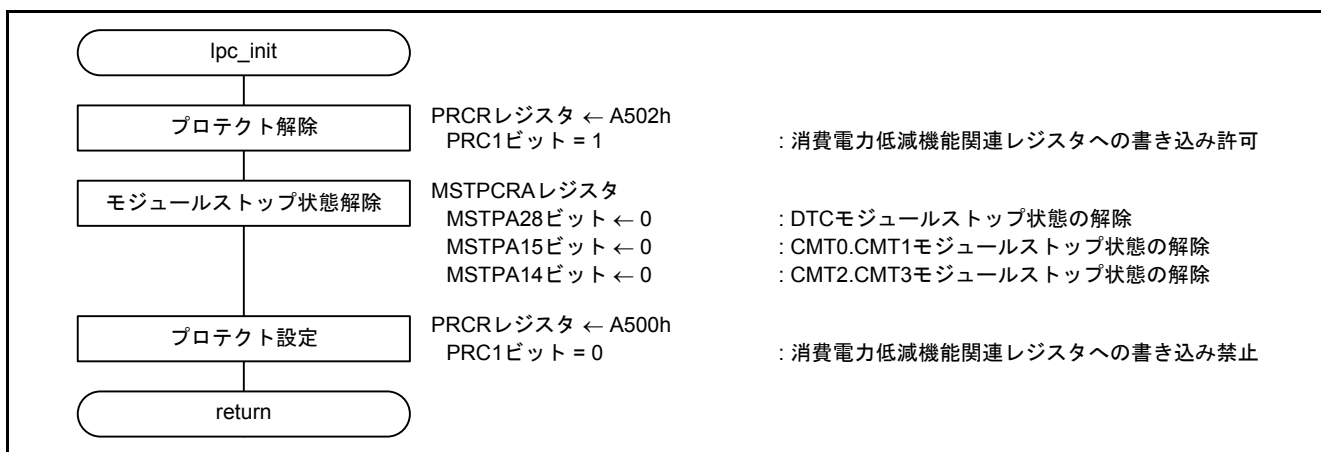


図5.6 消費電力低減機能初期設定

5.10.5 CMT 初期設定

図 5.7にCMT 初期設定のフローチャートを示します。

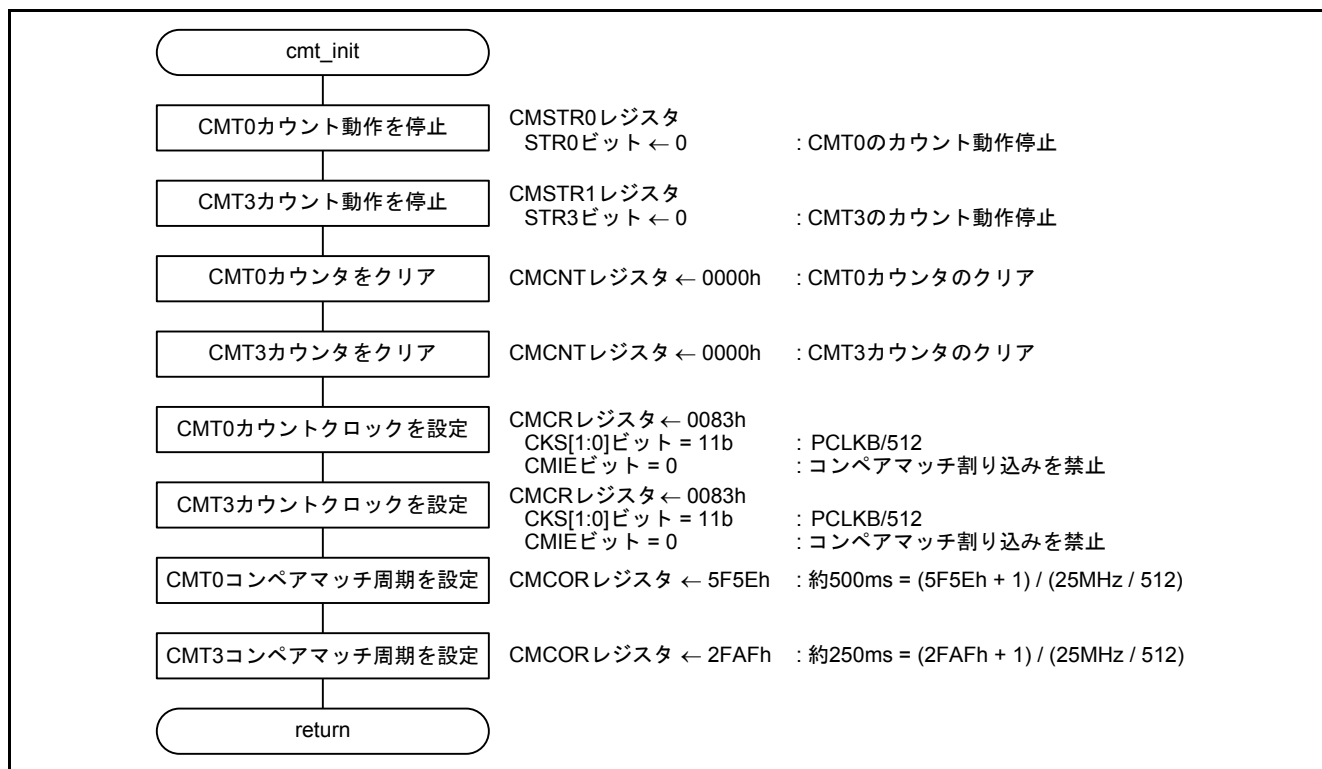


図5.7 CMT 初期設定

5.10.6 DTC 初期設定

図 5.8にDTC 初期設定のフローチャートを示します。



図5.8 DTC 初期設定

6. RX21A、RX220 グループ 初期設定例 アプリケーションノートとの組み合わせ方

本アプリケーションノートのサンプルコードは、RX210 グループで動作することを確認しています。RX21A グループや RX220 グループで動作させるには、それぞれの初期設定例のアプリケーションノートと組み合わせてください。

手順は、初期設定例のアプリケーションノート 「5. RX210 グループのアプリケーションノートを RX21A グループに適用する方法」、 「4. RX210 グループのアプリケーションノートを RX220 グループに適用する方法」を参照ください。

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

RX210グループ ユーザーズマニュアル ハードウェア編 Rev.1.50 (R01UH0037JJ)

RX21A グループ ユーザーズマニュアル ハードウェア編 Rev.1.00 (R01UH0251JJ)

RX220 グループ ユーザーズマニュアル ハードウェア編 Rev.1.10 (R01UH0292JJ)

(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 ファミリー 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>

お問い合わせ先

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

改訂記録	RX210、RX21A、RX220 グループ DTC ベクタテーブルの効率的な配置方法
------	--

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2013.03.01	—	初版発行
1.01	2014.07.01	1	対象デバイスに RX21A、RX220 グループを追加
		5	関連アプリケーションノートに RX21A、RX220 グループ 初期設定例のアプリケーションノートを追加
		14、15	参照するアプリケーションノートを各グループのアプリケーションノート初期設定例に変更
		20	RX21A、RX220 グループ 初期設定例と組み合わせる方法の参照先を追加
		21	参考ドキュメントに RX21A、RX220 グループのユーザーズマニュアルを追加

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、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/>