

SH7262/SH7264 グループ

RJJ06B1152-0100

Rev.1.00

2010.04.15

コントローラエリアネットワーク

ダイレクトメモリアクセスコントローラを用いたデータフレーム受信設定例

要旨

本アプリケーションノートでは、コントローラエリアネットワークで受信したデータフレームを、ダイレクトメモリアクセスコントローラを使用して内蔵 RAM へ格納する設定例について説明しています。

動作確認デバイス

SH7262/SH7264

以下、総称して「SH7264」として説明します。

目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラムリスト.....	17
4. 参考ドキュメント.....	28

1. はじめに

1.1 仕様

- コントローラエリアネットワークが1回データフレームを受信するごとにダイレクトメモリアクセスコントローラを起動し高速内蔵 RAM にデータを格納します。

1.2 使用機能

- コントローラエリアネットワーク
- ダイレクトメモリアクセスコントローラ (DMAC)

1.3 適用条件

マイコン	SH7262/SH7264
動作周波数	内部クロック : 144 MHz バスクロック : 72 MHz 周辺クロック : 36 MHz
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 Release00
コンパイルオプション	High-performance Embedded Workshop でのデフォルト設定 (-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)

1.4 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。合わせて参照してください。

- SH7262/SH7264 グループ 初期設定例
- SH7262/SH7264 グループ コントローラエリアネットワーク データフレーム送信設定例
- SH7262/SH7264 グループ コントローラエリアネットワーク データフレーム受信設定例
- SH7262/SH7264 グループ コントローラエリアネットワーク リモートフレーム送信設定例
- SH7262/SH7264 グループ コントローラエリアネットワーク リモートフレーム受信設定例

1.5 "L"アクティブ端子 (信号) の表記について

端子名 (信号名) 末尾の # は "L" アクティブ端子 (信号) であることを示します。

2. 応用例の説明

本応用例では、コントローラエリアネットワークで受信したデータフレームを、ダイレクトメモリアクセスコントローラを使用して高速内蔵 RAM へ格納します。

2.1 使用機能の動作概要

(1) コントローラエリアネットワーク

SH7264 は CAN2.0B Active と ISO-11898 をサポートするコントローラエリアネットワークを 2 チャンネル内蔵しています。

コントローラエリアネットワークはプログラム可能な 31 個の送受信用メールボックスおよび 1 個の受信用メールボックス、全てのメールボックスに対応したプログラム可能な受信フィルタマスクを持ち、自由度の高い通信方法を提供します。表 1 にコントローラエリアネットワークの特長を、図 1 にブロック図を示します。また、表 2 に割り込み要因を示します。ダイレクトメモリアクセスコントローラを起動できる割り込み要因はメールボックス 0 のデータフレーム受信割り込みとリモートフレーム受信割り込みのみです。コントローラエリアネットワークについての詳細は、「SH7262 グループ、SH7264 グループ ハードウェアマニュアル コントローラエリアネットワーク」の章を参照してください。

表1 コントローラエリアネットワークの特長

項目	内容
対応規格	CAN 規格 2.0B。ビットタイミングは ISO-11898 規格に準拠
チャンネル数	2 チャンネル*
メールボックス	32 個* (プログラム可能な 31 個の送受信用メールボックスおよび 1 個の受信用メールボックス)
転送レート	最大 1Mbps
割り込み要因	16
トリガモード	イベントトリガモード タイムトリガモード
テスト機能	リスンオンリモード、エラーパッシブモードを内蔵
DMA 転送	メールボックス 0 のデータフレーム受信、またはリモートフレーム受信の割り込みで DMAC を起動可能

【注】 *2 チャンネルを組み合わせて 64 メールボックス×1 チャンネルとして使用することも可能です。

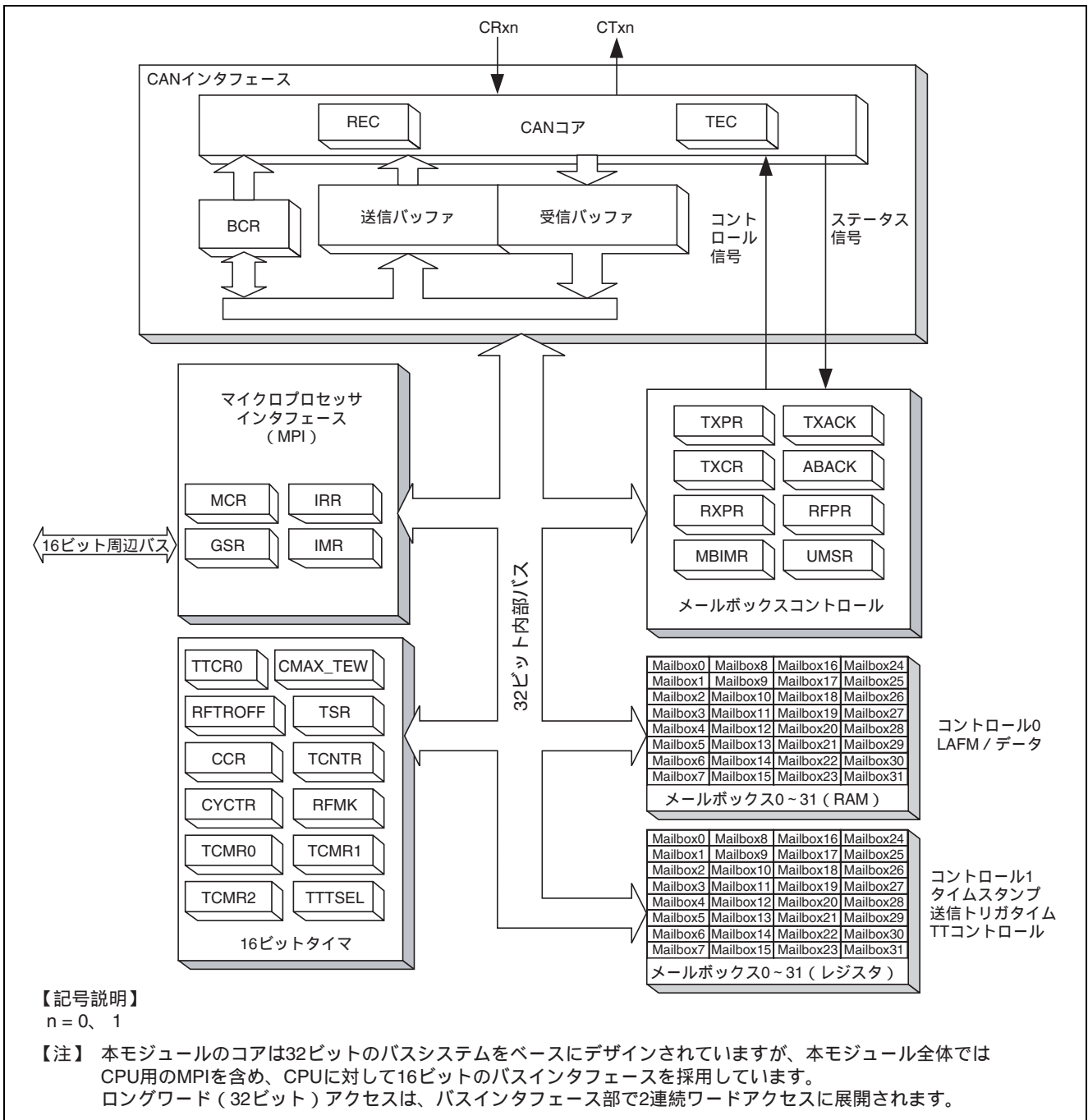


図1 コントローラエリアネットワークのブロック図（1チャンネルあたり）

表2 コントローラエリアネットワークの割り込み要因 (1チャンネルあたり)

名称	要因	割り込み フラグ	ダイレクトメモリ アクセスコントローラの 起動	
ERS	エラーパッシブ (TEC \geq 128 または REC \geq 128)	IRR5	不可	
	バスオフ (TEC \geq 256) /バスオフからの復帰	IRR6		
	エラーワーニング (TEC \geq 96)	IRR3		
	エラーワーニング (REC \geq 96)	IRR4		
OVR	リセット/ホルト/CAN スリープ遷移	IRR0		
	オーバロードフレーム送信	IRR7		
	未読メッセージのオーバーライト (オーバラン)	IRR9		
	スタートシステムマトリックス	IRR10		
	TCMR2 コンペアマッチ	IRR11		
	CAN スリープ中 CAN バス動作の検出	IRR12		
	タイマオーバラン/Next_is_Gap/メッセージエラー	IRR13		
	TCMR0 コンペアマッチ	IRR14		
	TCMR1 コンペアマッチ	IRR15		
RM0 ^{※1}	データフレーム受信	IRR1 ^{※2}		可 ^{※3}
RM1 ^{※1}	リモートフレーム受信	IRR2 ^{※2}		
SLE	メッセージの送信/送信取り消し (スロットエンプティ)	IRR8	不可	

- 【注】 ※1 RM0 はメールボックス 0 のリモートフレーム受信フラグ (RFPR0[0]) またはデータフレーム受信フラグ (RXPR0[0]) による割り込み、RM1 はメールボックス n (n=1~31) のリモートフレーム受信フラグ (RFPR1[n]) またはデータフレーム受信フラグ (RXPR1[n]) による割り込みです。
- ※2 IRR1 はメールボックス 0~31 のデータフレーム受信フラグ、IRR2 はメールボックス 0~31 のリモートフレーム受信フラグです。
- ※3 RM0 割り込みのみダイレクトメモリアクセスコントローラを起動できます。

(2) ダイレクトメモリアクセスコントローラ

ダイレクトメモリアクセスコントローラは、DACK（転送要求受け付け信号）付き外部デバイス、外部メモリ、内蔵メモリ、メモリマップト外部デバイス、および内蔵周辺モジュール間のデータ転送を CPU に代わって高速に行うことができます。

ダイレクトメモリアクセスコントローラのバスモードにはサイクルスチールモードとバーストモードがあります。

サイクルスチールモードでは、ダイレクトメモリアクセスコントローラは一回の転送単位（バイト、ワード、ロングワード、または 16 バイト単位）の転送を終了するたびにバス権を他のバスマスタに渡します。その後転送要求があれば、他のバスマスタからバス権を取り戻し、再び 1 転送単位の転送を行い、その転送を終了するとまたバス権を他のバスマスタに渡します。これを転送終了条件が満たされるまで繰り返します。

本応用例ではサイクルスチールモードを用いてコントローラエリアネットワークのメールボックス 0 に受信されたデータを高速内蔵 RAM に転送します。

表 3 にダイレクトメモリアクセスコントローラの特長を、図 2 にブロック図を示します。ダイレクトメモリアクセスコントローラについての詳細は、「SH7262 グループ、SH7264 グループ ハードウェアマニュアルダイレクトメモリアクセスコントローラ」の章を参照してください。

表3 ダイレクトメモリアクセスコントローラの特長

項目	内容
チャンネル数	16 チャンネル
アドレス空間	アーキテクチャ上は 4G バイト
転送データ単位	バイト、ワード、ロングワード、16 バイト
最大転送回数	16,777,216 (24 ビット) 回
アドレスモード	シングルアドレスモード、デュアルアドレスモード
転送要求	外部リクエスト、内蔵周辺モジュールリクエスト、オートリクエスト
バスモード	サイクルスチールモード（通常モードとインターミッテントモード）、バーストモード
優先順位	チャンネル優先順位固定モードの 2 種類から選択可能
割り込み要求	データ転送 1/2 終了、データ転送終了
リロード機能	実行中の DMA 転送と同じ設定での DMA 転送を再設定することなく繰り返し実行することができます。また、DMA 転送中にリロードレジスタをあらかじめ設定しておくことで、次回の DMA 転送を異なる設定で実行することができます。このリロード機能は、チャンネルごと、リロードレジスタごとに ON/OFF の設定が可能です。

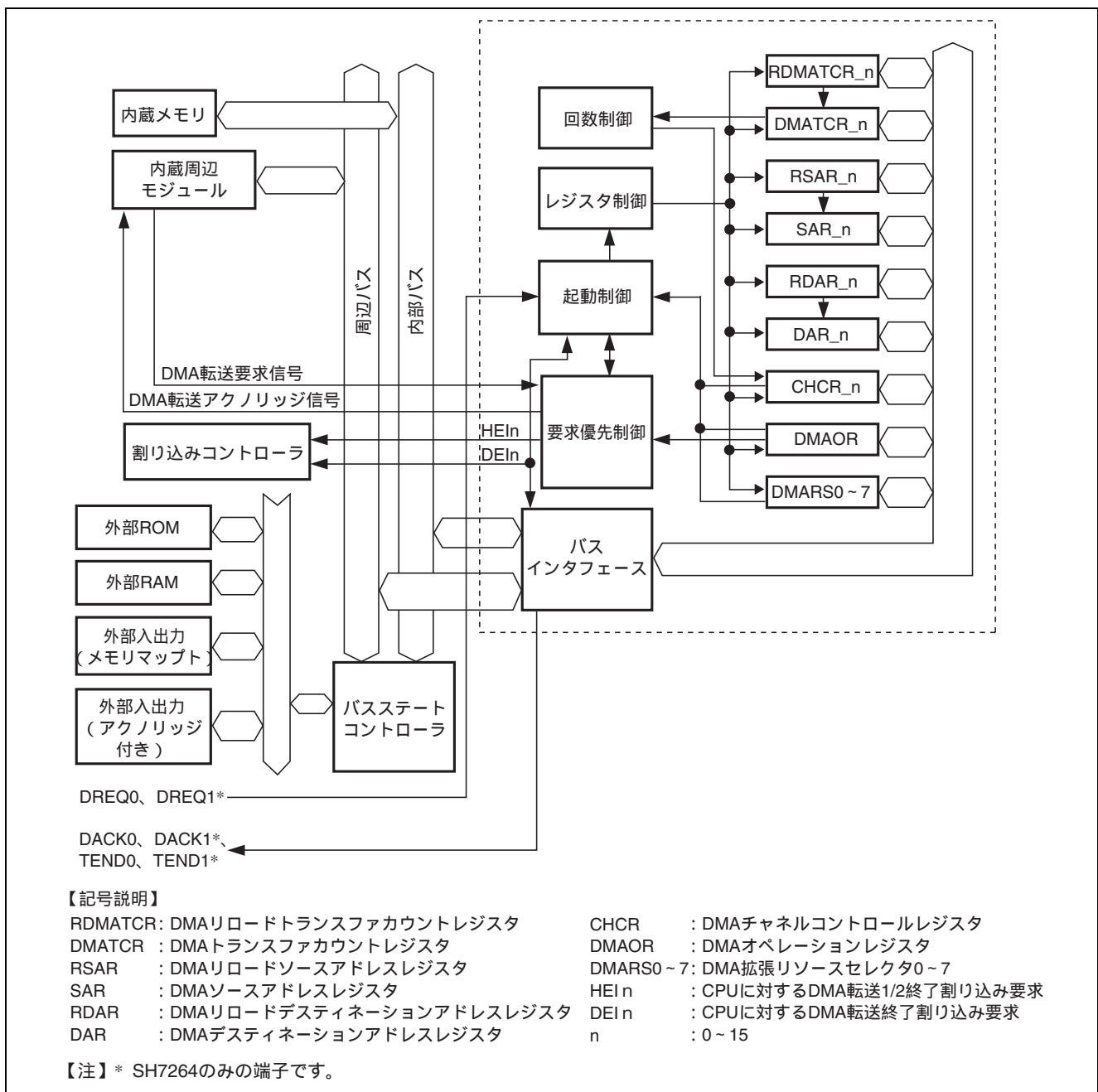


図2 ダイレクトメモリアクセスコントローラのブロック図

2.2 使用機能の設定手順

(1) コントローラエリアネットワークの設定手順

コントローラエリアネットワークの初期設定はリセットモード（コンフィギュレーションモード）で行います。初期設定終了後、リセットモードを解除することでCANバスアクティビティに参加します。コントローラエリアネットワークのデータフレーム受信割り込み（IRR1）でダイレクトメモリアクセスコントローラを起動させる場合、インタラプトマスクレジスタ（IMR）のビット1（IMR1）とメールボックスインタラプトマスクレジスタ0（MBIMR0）のビット0（MBIMR0[0]）を割り込み許可に設定します。

図3、図4にコントローラエリアネットワークの初期設定フロー例を示します。なお、各レジスタ設定の詳細は、「SH7262 グループ、SH7264 グループ ハードウェアマニュアル」を参照してください。

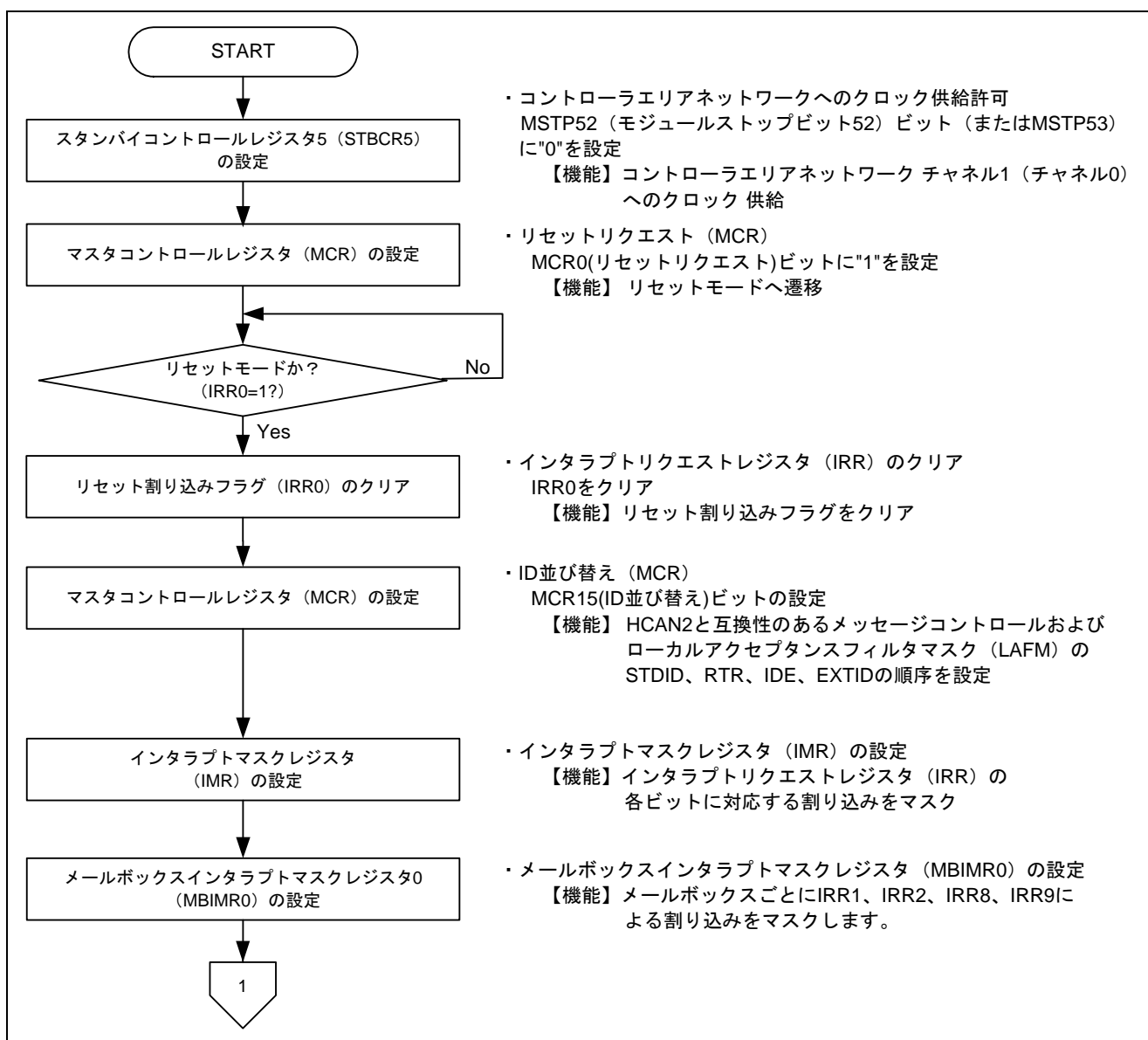


図3 コントローラエリアネットワークの初期設定フロー例（1）

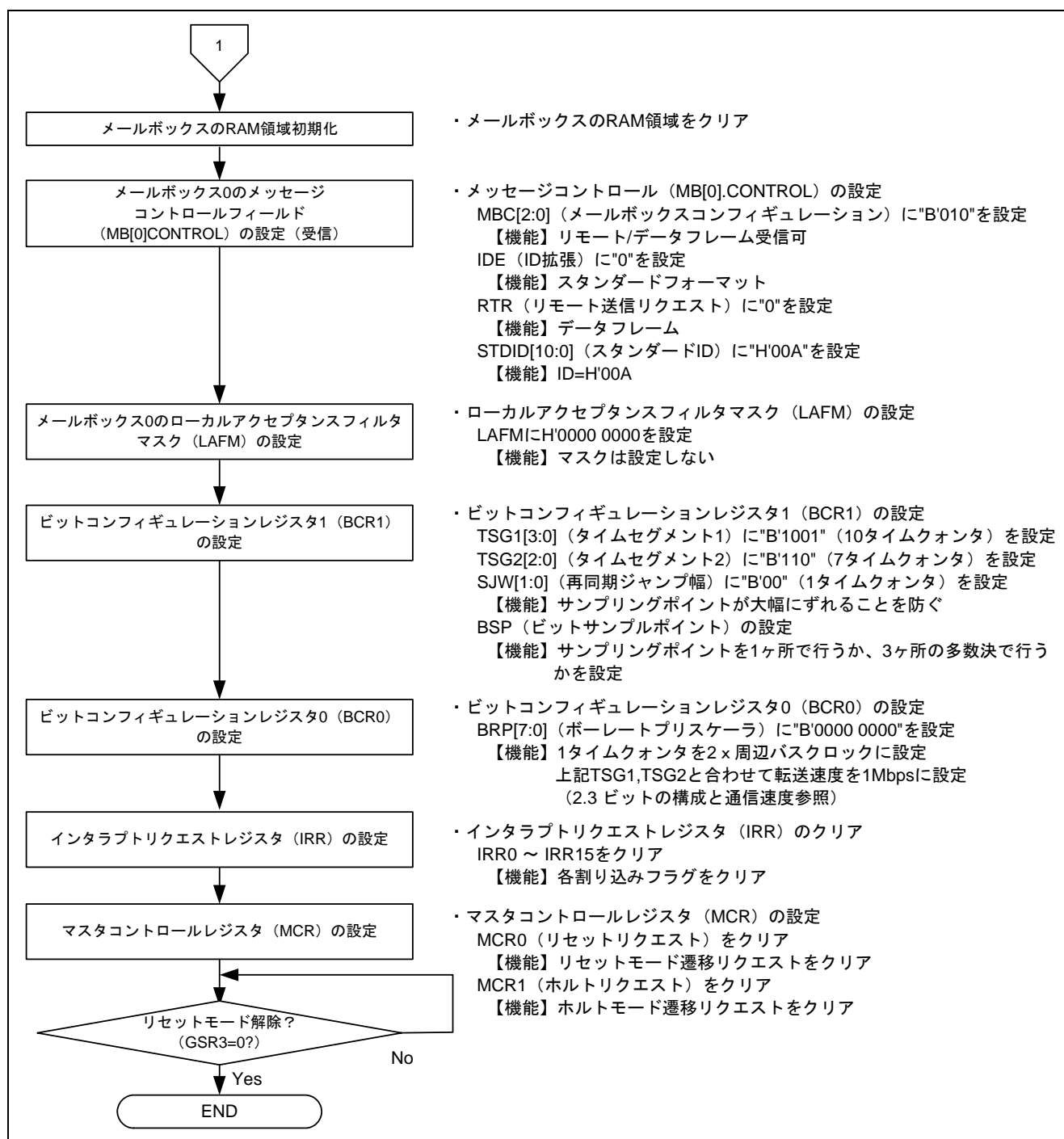


図4 コントローラエリアネットワークの初期設定フロー例 (2)

(2) ダイレクトメモリアクセスコントローラの設定手順

コントローラエリアネットワークのデータフレーム受信割り込み (IRR1) を起動要因とした場合、使用できるバスモードはサイクルスチールモードです。また、本応用例では DMA ソースアドレスレジスタと DMA トランスファカウンタレジスタに対してリロード機能を使用しています。図 5、図 6 にダイレクトメモリアクセスコントローラの初期設定フロー例を示します。なお、各レジスタ設定の詳細は、「SH7262 グループ、SH7264 グループ ハードウェアマニュアル」を参照してください。

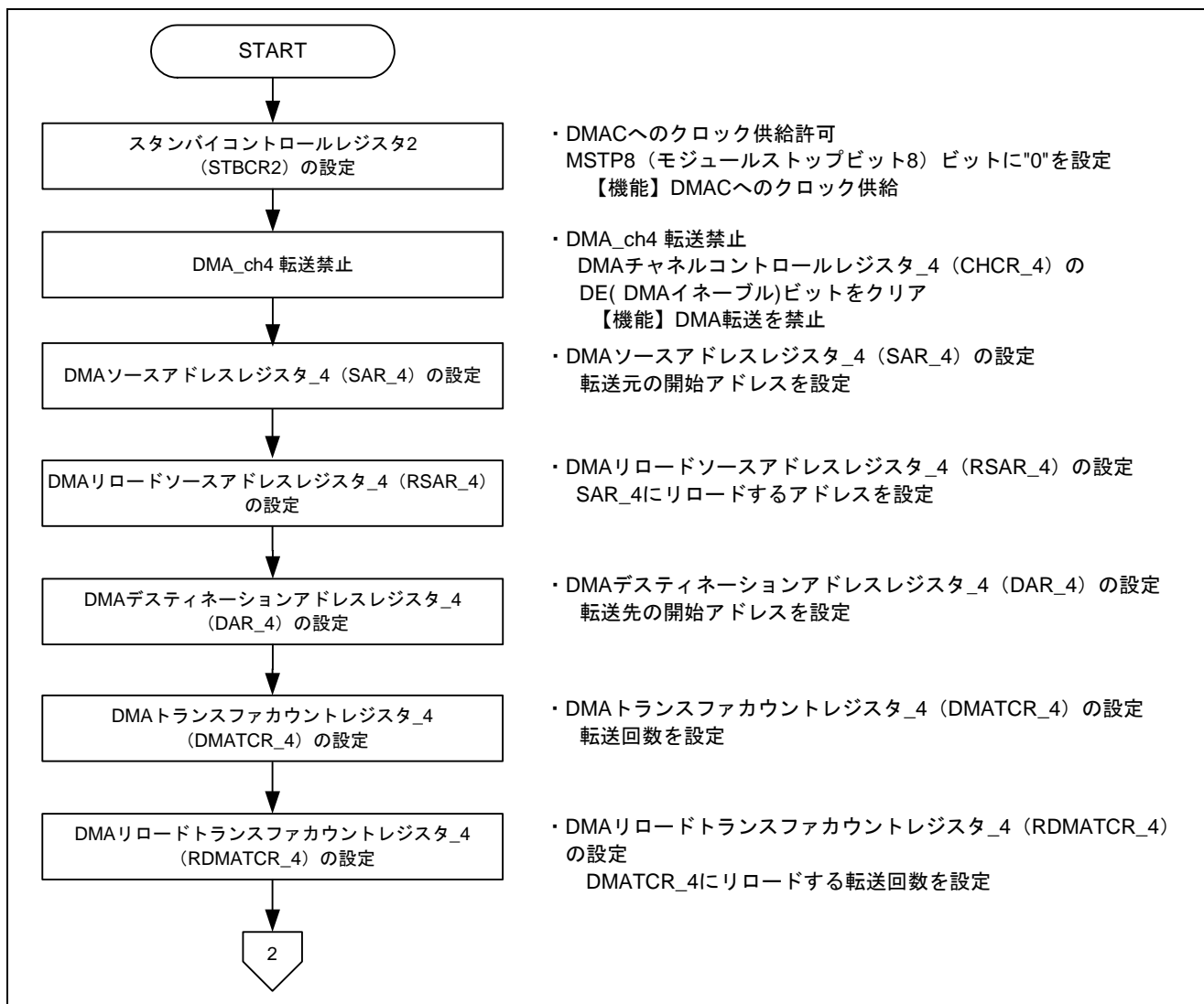


図5 ダイレクトメモリアクセスコントローラの初期設定フロー例 (1)

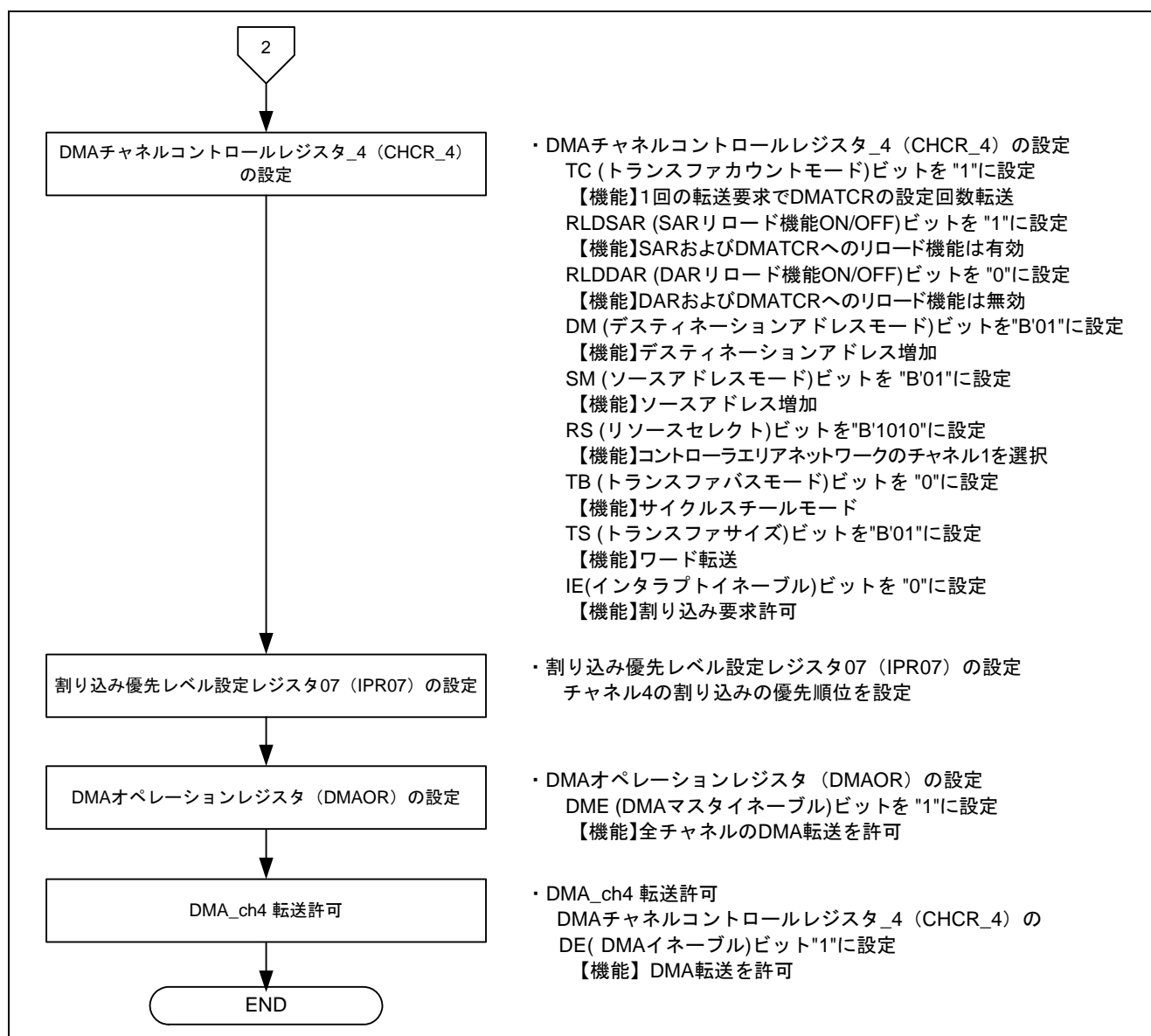


図6 ダイレクトメモリアクセスコントローラの初期設定フロー例 (2)

2.3 ビットの構成と通信速度

コントローラエリアネットワークでの1ビットは以下に示す4つのセグメントで構成されています。

- ① シンクロナイゼーションセグメント (SS)
- ② プロパゲーションタイムセグメント (PRSEG)
- ③ フェーズバッファセグメント 1 (PHSEG1)
- ④ フェーズバッファセグメント 2 (PHSEG2)

さらに、各セグメントは T_q (タイムクオンタ) と呼ばれる基準時間で構成されます。図 7 に $SS=1T_q$ 、 $PRSEG=8T_q$ 、 $PHSEG1=8T_q$ 、 $PHSEG2=8T_q$ の場合のビット構成例を示します。

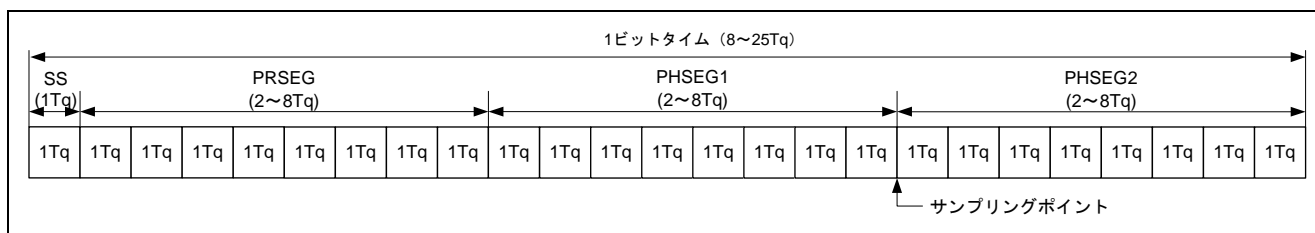


図7 1ビットの構成図

コントローラエリアネットワークでは、 $PRSEG+PHSEG1$ の T_q 数をビットコンフィギュレーションレジスタ 1 (BCR1) の $TSG1[3:0]$ に、 $PHSEG2$ の T_q 数を $TSG2[2:0]$ に設定します (設定値+1 が T_q 数)。また、 $1T_q$ に対応する周辺バスクロック数をビットコンフィギュレーションレジスタ 0 (BCR0) の $BRP[7:0]$ に設定します。

以下の説明では $BRP[7:0]$ 、 $TSG1[3:0]$ 、 $TSG2[2:0]$ はレジスタ設定値、 BRP 、 $TSEG1$ 、 $TSEG2$ 、 SJW はレジスタ設定値に対応する値を示します。レジスタ設定値に対応する値は「SH7262 グループ、SH7264 グループ ハードウェアマニュアル コントローラエリアネットワーク」の章を参照してください。

CAN では $1T_q = \frac{2 \times (BRP[7:0] + 1)}{\text{周辺バスクロック}}$ と定義しており、通信速度は以下の通り計算されます。

$$\begin{aligned} \text{通信速度} &= \frac{\text{周辺バスクロック}}{2 \times (BRP[7:0] + 1) \times (1 \text{ ビットの } T_q \text{ 数})} \\ &= \frac{\text{周辺バスクロック}}{2 \times (BRP[7:0] + 1) \times ((TSG1[3:0] + 1) + (TSG2[2:0] + 1) + 1)} \end{aligned}$$

ビットコンフィギュレーションレジスタ設定上の制限事項を以下に示します。

$$TSEG1 \text{ (Min.)} > TSEG2 \geq SJW \text{ (Max.)} \quad (SJW=1 \sim 4)$$

SJW: 再同期ジャンプ幅。位相誤差を補正するためにフェーズバッファセグメント 1 を延長、または、フェーズバッファセグメント 2 を短縮するセグメントです。

$$8 \leq TSEG1 + TSEG2 + 1 \leq 25 \text{ タイムクオンタ}$$

$$TSEG2 \geq 2$$

本参考プログラムでは、周辺バスクロック=36MHz、 $BRP[7:0]=0$ 、 $TSG1[3:0]=9$ 、 $TSG2[2:0]=6$ に設定していますので通信速度は下記のようになります。

$$\text{通信速度} = \frac{36\text{M}}{2 \times (0+1) \times ((9+1) + (6+1) + 1)} = 1\text{M} \dots 1\text{Mbps}$$

2.4 参考プログラムの動作

本参考プログラムでは、コントローラエリアネットワークのチャンネル1を使用して、メールボックス0にID : H'00A のスタンダードフォーマットのデータフレームを受信します。データフレーム受信割り込み (IRR1) を起動要因としダイレクトメモリアクセスコントローラでメールボックス0の内容を高速内蔵RAMに転送します。この時、CPU へのデータフレーム受信割り込みは発生しません。ダイレクトメモリアクセスコントローラは転送終了後、リロード機能でDMA ソースアドレスレジスタ、DMA トランスファカウントレジスタの値を初期値に再設定します。受信データを連続した領域に順に格納するため、デスティネーションアドレスレジスタはリロードしません。ダイレクトメモリアクセスコントローラの転送終了割り込み処理でTEフラグをクリアし次のデータフレーム受信割り込みによる起動に備えます。受信バッファ変数に空きがなくなったらダイレクトメモリアクセスコントローラを禁止します。

転送先の領域にキャッシュ有効領域を指定する場合は、キャッシュとメモリ間のコヒーレンシを保証するため、必要に応じてキャッシュのエントリを無効化してください。

図8に参考プログラムの動作概要を示します。

【注】 参考プログラムは送信、受信ともに実行していますが、ここでは受信部のみを説明します。

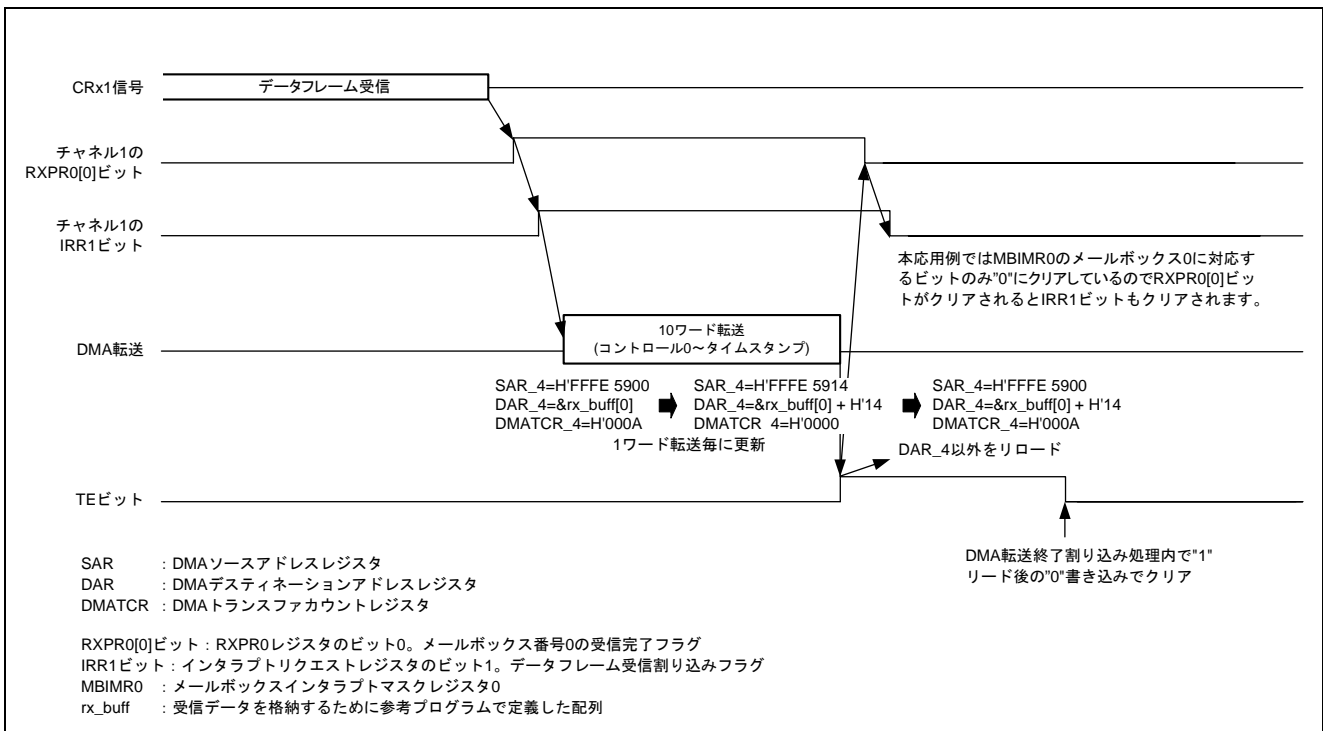


図8 参考プログラムの動作概要

2.5 参考プログラムの処理手順

表4にコントローラエリアネットワークの設定例を、表5にダイレクトメモリアクセスコントローラの設定例を示します。また、図9に本参考プログラムのフローを示します。

【注】 参考プログラムは送信、受信ともに実行していますが、ここでは受信部のみを説明します。

表4 参考プログラムのレジスタ設定

レジスタ名	アドレス	設定値	機能
スタンバイコントロール レジスタ5 (STBCR5)	H'FFFE 0410	H'FB	MSTP52="0" : コントローラエリアネットワークチャンネル1は動作
マスタコントロールレジスタ (MCR)	H'FFFE 5800	H'0001	MCR0="1" : リセットモード遷移 リクエスト
		H'8001	MCR15="1" : RCAN-ET と HCAN2 は 異なる順序
		H'8000	MCR0="0" : リセットモード解除
インタラプトマスクレジスタ (IMR)	H'FFFE 580A	H'FFFD	データフレーム受信割り込み (IRR1) 許可
メールボックスインタラプト マスクレジスタ0 (MBIMR0)	H'FFFE 5852	H'FFFE	MBIMR0[0]="0" : メールボックス0の受 信割り込みを許可
ビットコンフィギュレーション レジスタ1 (BCR1)	H'FFFE 5804	H'9600	TSG1[3:0]="B'1001" : PRSEG + PHSEG1 = 10Tq TSG2[2:0]="B'110" : PHSEG2 = 7Tq SJW="0" : SJW=1Tq BSP="0" : 1か所でビットサンプリング
ビットコンフィギュレーション レジスタ0 (BCR0)	H'FFFE 5806	H'0000	BRP[7:0]= "0" : 1Tq=2xPφ
メールボックス0メッセージ コントロールフィールド1 (MB[0].CONTROL1)	H'FFFE 5910	H'0200	MBC[2:0]="B'010" : データフレームと リモートフレームの受信可
メールボックス0メッセージ コントロールフィールド0 (MB[0].CONTROL0)	H'FFFE 5900	H'0028 0000	IDE="0" : スタンダードフォーマット RTR="0" : データフレーム STDID[10:0]= " H'00A " : スタンダード ID= H'00A
メールボックス0ローカル アクセプタンスフィルタマスク (MB[0].LAFM)	H'FFFE 5904	H'0000 0000	クリア : MASK は設定しない

表5 ダイレクトメモリアクセスコントローラの設定

レジスタ名	アドレス	設定値	機能
スタンバイコントロール レジスタ 2 (STBCR2)	H'FFFE 0018	H'00	MSTP8="0" : DMAC は動作
DMA ソースアドレスレジスタ _4 (SAR_4)	H'FFFE 1040	H'FFFE 5900	転送元の開始アドレス: メールボックス 0 の先頭アドレスを設定
DMA リロードソースアドレス レジスタ_4 (RSAR_4)	H'FFFE 1140	H'FFFE 5900	SAR_4 にリロードするアドレス: メール ボックス 0 の先頭アドレスを設定
DMA デスティネーション アドレスレジスタ_4 (DAR_4)	H'FFFE 1044	&rx_buff[0]	転送先の開始アドレス: 受信バッファの先 頭アドレスを設定
DMA トランスファカウント レジスタ_4 (DMATCR_4)	H'FFFE 1048	H'0000 000A	転送回数: 10 回
DMA リロードトランスファ カウントレジスタ_4 (RDMATCR_4)	H'FFFE 1148	H'0000 000A	DMATCR_4 にリロードする転送回数: 10 回
DMA チャンネルコントロール レジスタ_4 (CHCR_4)	H'FFFE 104C	H'0000 0000	DE = "0": DMA 転送禁止
		H'A000 5A0C	TC = "1" 1 回の DMA 要求で DMATCR0 の 設定回数転送 RLDSAR = "1": ソースアドレスリロード機 能有効 RLDDAR = "0": デスティネーションアドレ スリロード機能無効 DM = "B'01": デスティネーションアドレス増加 SM = "B'01": ソースアドレス増加 RS = "B'1010": 転送要求元をコントロー ラエリアネットワークチャンネル 1 に設定 TB = "0": サイクルスチールモード TS = "B'01": ワード転送 IE = "0": 割り込み要求許可
		H'A000 5A0D	DE = "1": DMA 転送許可
DMA オペレーションレジスタ (DMAOR)	H'FFFE 1200	H'0001	DME = 全チャンネルの DMA 転送を許可
割り込み優先レベル設定 レジスタ 07 (IPR07)	H'FFFE 0C02	H'A000	DMAC チャンネル 4 の転送終了割り込みレ ベル: レベル 10 を設定

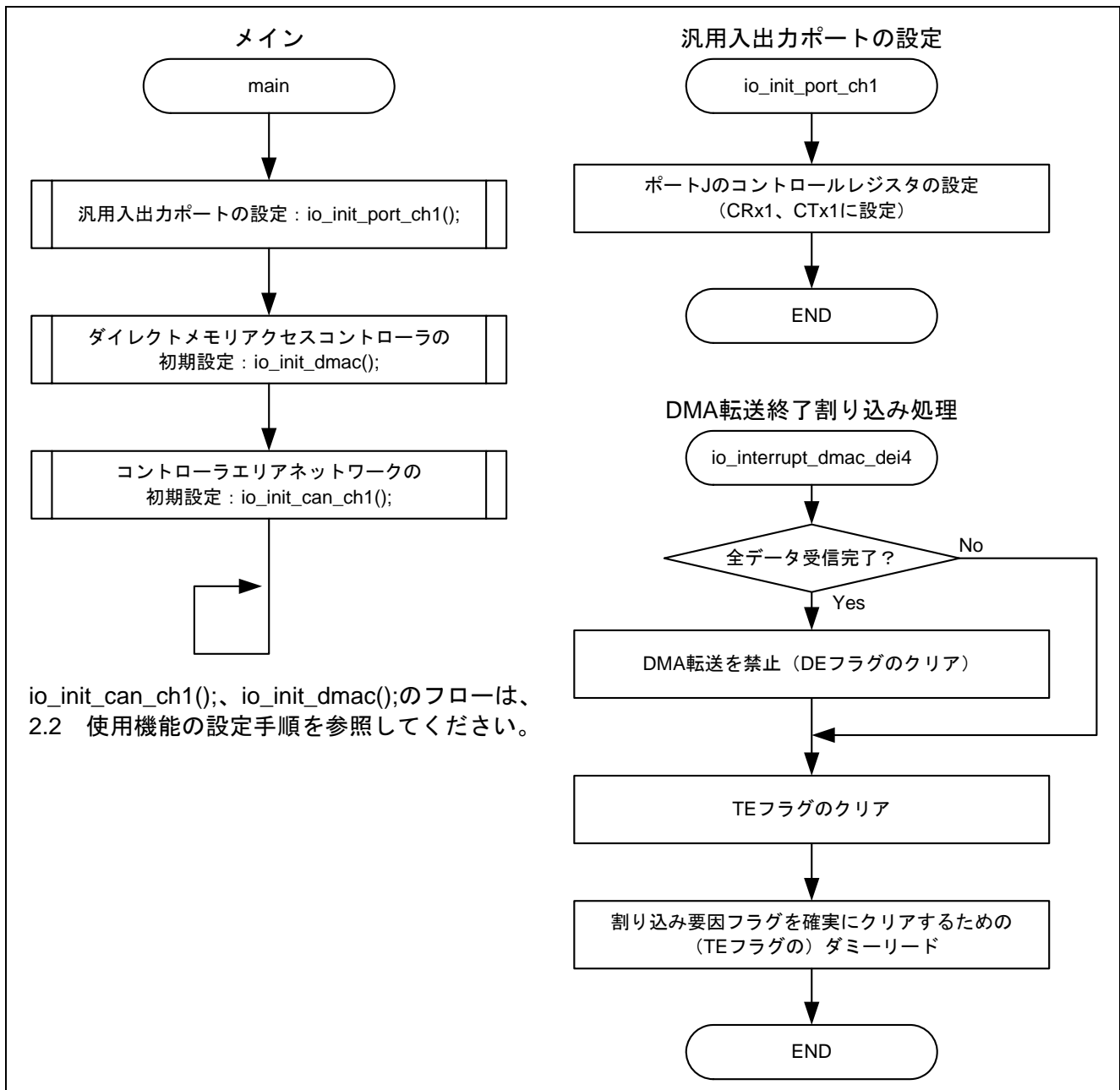


図9 参考プログラムの処理フロー

3. 参考プログラムリスト

3.1 参考プログラムについての補足

SH7264 は、製品によって大容量内蔵 RAM の容量が 1MB または 640KB と異なるため、参考プログラムのセクション配置やレジスタの設定を一部変更する必要があります。そのため本アプリケーションノートでは 1MB 用と 640KB 用の 2 つのワークスペースを用意しています。

640KB 版はライトプロテクトを解除しなければ保持用内蔵 RAM へ書き込むことができないため、640KB 版のワークスペースは、システムコントロールレジスタ 5 (SYSCR5) にライトプロテクトの解除を設定しています。

使用する製品を確認した上で、対応するワークスペースを使用してください。

3.2 サンプルプログラムリスト "main.c" (1)

```
1  /*****
2  *  DISCLAIMER
3  *
4  *  This software is supplied by Renesas Electronics Corp. and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics Corp. and is protected under
8  *  all applicable laws, including copyright laws.
9  *
10 *  THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *  REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *  INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *  PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *  DISCLAIMED.
15 *
16 *  TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *  ELECTRONICS CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *  FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *  FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *  AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *  Renesas reserves the right, without notice, to make changes to this
23 *  software and to discontinue the availability of this software.
24 *  By using this software, you agree to the additional terms and
25 *  conditions found by accessing the following link:
26 *  http://www.renesas.com/disclaimer
27 *****/
28 * (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 * "FILE COMMENT"***** Technical reference data *****
30 * System Name : SH7264 Sample Program
31 * File Name   : main.c
32 * Abstract    : DMAC+CAN Module Application (Data Frame Transmit and Receive)
33 * Version     : 1.00.01
34 * Device      : SH7262/SH7264
35 * Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *              : C/C++ compiler package for the SuperH RISC engine family
37 *              : (Ver.9.03 Release00).
38 * OS          : None
39 * H/W Platform: M3A-HS64G50(CPU board)+M3A-HS64G02(IO board)
40 * Description :
41 *****/
42 * History     : Feb.09,2010 ver.1.00.00
43 *              : Apr.15,2010 Ver.1.00.01 Changed the company name
44 * "FILE COMMENT END"*****/
45 #include "iodefine.h"      /* SH7264 iodefine */
46
```

3.3 サンプルプログラムリスト "main.c" (2)

```
47  /* ---- プロトタイプ宣言 ---- */
48  void main(void);
49  void io_interrupt_dmac_dei4(void);
50  extern void io_init_port_ch0(void);
51  extern void io_init_port_ch1(void);
52  extern void io_init_can_ch0(void);
53  extern void io_init_can_ch1(void);
54  extern void io_init_dmac(void *src, void *dst, int count);
55  extern void io_data_send_ch0(unsigned char *addr, int size);
56
57  /* ---- マクロ定義 ---- */
58  #define NUM_OF_BUFF      10          /* 受信バッファの面数 */
59
60  /* ---- 型定義 ---- */
61  typedef struct{
62      unsigned long   control0;      /* メッセージコントロールフィールド 0 */
63      unsigned long   lafm;          /* ローカルアクセプタンスフィルタマスク */
64      unsigned char   data[8];      /* データ */
65      unsigned short  control1;      /* メッセージコントロールフィールド 1 */
66      unsigned short  timestamp;     /* タイムスタンプ */
67  }MB_TYPE;
68
69  /* ---- 変数定義 ---- */
70  MB_TYPE      rx_buff[ NUM_OF_BUFF ]; /* 受信バッファ */
71  unsigned char tx_data[8] = {0xc0, 0xc1, 0xc2, 0xc3, 0xc4, 0xc5, 0xc6, 0xc7};
72                                     /* 送信データ */
73
```

3.4 サンプルプログラムリスト "main.c" (3)

```
74  /*"FUNC COMMENT"*****
75  * ID      :
76  * Outline : Sample program main
77  *-----
78  * Include : "iodefine.h"
79  *-----
80  * Declaration : void main(void);
81  *-----
82  * Description : コントローラエリアネットワーク (RCAN) の初期設定を行った後、
83  *              : チャンネル0はデータフレームを送信します。チャンネル1はデータフレームを
84  *              : 受信します。受信データ転送には、ダイレクトメモリアクセスコントローラ
85  *              : (DMAC)を使用します。
86  *-----
87  * Argument  : void
88  *-----
89  * Return Value : void
90  *-----
91  * Note      :
92  *"FUNC COMMENT END"*****/
93  void main(void)
94  {
95      int i;
96      unsigned char *p;
97      int idx, sz;
98
99      /* ==== 汎用入出力ポートの初期設定 ==== */
100     io_init_port_ch1();
101     io_init_port_ch0();
102
103     /* ==== DMACの初期設定(チャンネル1) ==== */
104     io_init_dmac( &RCAN1.MB[0], rx_buff, sizeof(MB_TYPE) / sizeof(short) );
105                 /* チャンネル1-メールボックス0の受信データを内蔵RAMに転送 */
106
107     /* ==== CANの初期設定 ==== */
108     io_init_can_ch1();
109     io_init_can_ch0();
110
111     /* ==== データ送信(チャンネル0) ==== */
112     for( i=0; i<NUM_OF_BUFF; i++ ){
113         io_data_send_ch0( tx_data, 8 );
114     }
115
116     while(1){
117         /* wait */
118     }
119 }
```

3.5 サンプルプログラムリスト "main.c" (4)

```
120  /*"FUNC COMMENT"*****
121  * ID          :
122  * Outline     : DMAC setting
123  * -----
124  * Include     : "iodefine.h"
125  * -----
126  * Declaration : void io_init_dmac(void *src, void *dst, int count);
127  * -----
128  * Description : ダイレクトメモリアクセスコントローラ(DMAC)の初期設定を行います。
129  *             : サイクルスチールモード、デュアルアドレスモード、
130  *             : 内蔵周辺モジュールリクエスト(RCAN)、転送データ長:ワード、
131  *             : ソース:RCAN メールボックス 0、デスティネーション:内蔵RAMを設定します。
132  *             : ソースアドレスに対しては、リロード機能も使用します。
133  * -----
134  * Argument    : void *src ; I : 転送元アドレス
135  *             : void *dst ; O : 転送先アドレス
136  *             : int count ; I : 転送回数
137  * -----
138  * Return Value : void
139  * -----
140  * Note        :
141  * "FUNC COMMENT END"*****/
142  void io_init_dmac(void *src, void *dst, int count)
143  {
144  /* ==== スタンバイコントロールレジスタ 2 の設定 ==== */
145  CPG.STBCR2.BIT.MSTP8 = 0;          /* DMAC のモジュールスタンバイ解除 */
146
147  /* ====DMA_ch4 転送禁止 ==== */
148  DMAC.CHCR4.BIT.DE = 0x0;          /* DMA 禁止 */
149
150  /* ====DMA ソースアドレスレジスタ_4 (SAR_4) の設定 ==== */
151  DMAC.SAR4.LONG = (unsigned long)src;
152
153  /* ====DMA リロードソースアドレスレジスタ_4 (RSAR_4) の設定 ==== */
154  DMAC.RSAR4.LONG = (unsigned long)src;
155
156  /* ====DMA デスティネーションアドレスレジスタ_4 (DAR_4) の設定 ==== */
157  DMAC.DAR4.LONG = (unsigned long)dst;
158
159  /* ====DMA トランスファカウンタレジスタ_4 (DMATCR_4) の設定 ==== */
160  DMAC.DMATCR4.LONG = count;
161
162  /* ====DMA リロードトランスファカウンタレジスタ_4 (RDMATCR_4) の設定 ==== */
163  DMAC.RDMATCR4.LONG = count;
164
```

3.6 サンプルプログラムリスト "main.c" (5)

```

165  /* ====DMA チャンネルコントロールレジスタ_4 (CHCR_4) の設定 ==== */
166  DMAC.CHCR4.LONG = 0xa0005a0c;
167      /*
168          bit31   : TC DMATCR 転送 : 1----- DMATCR の設定回数転送
169          bit30   : reserve 0
170          bit29   : RLDSAR ON : 1----- リロード機能有効 (RSAR)
171          bit28   : RLDDAR ON : 0----- リロード機能無効 (RDAR)
172          bit27   : reserve 0
173          bit26   : DAF : 0----- 未使用
174          bit25   : SAF : 0----- 未使用
175          bit24   : reserve 0
176          bit23   : DO over run0 : 0----- 未使用
177          bit22   : TL TEND low active : 0---- 未使用
178          bit21   : reserve 0
179          bit20   : TEMASK :0----- 未使用
180          bit19   : HE :0----- 未使用
181          bit18   : HIE :0----- 未使用
182          bit17   : AM :0----- 未使用
183          bit16   : AL :0----- 未使用
184          bit15-14 : DM1:0 DM0:1----- デスティネーションアドレス増加
185          bit13-12 : SM1:0 SM0:1----- ソースアドレス増加
186          bit11-8  : RS : B'1010----- 転送要求元は RCAN のチャンネル 1
187          bit7     : DL : DREQ level : 0 ----- 未使用
188          bit6     : DS : DREQ select :0 Low level 未使用
189          bit5     : TB :cycle :0----- サイクルスチール
190          bit4-3   : TS : transfer size :B'01--- ワード単位転送
191          bit2     : IE : interrupt enable :1--- 割り込み許可
192          bit1     : TE : transfer end :0----- TE フラグのクリア
193                                     (1 リード後の 0 クリアのみ有効)
194          bit0     : DE : DMA enable bit :0----- DMA 転送禁止
195      */
196  /* =====割り込み優先レベル設定レジスタ 07 (IPR07) の設定===== */
197  INTC.IPR07.BIT._DMAC4 = 0xa;
198
199  /* ----DMA オペレーションレジスタ (DMAOR) の設定---- */
200  DMAC.DMAOR.WORD |= 0x0007; /* DME ビットをセット。アドレスエラーフラグ、*/
201                          /* NMI フラグのクリア防止のため、AE ビット、*/
202                          /* NMIF フラグには 1 をライト*/
203  /* =====DMA_ch4 転送許可===== */
204  DMAC.CHCR4.BIT.DE = 0x1;
205
206  }

```

3.7 サンプルプログラムリスト "main.c" (6)

```
207 /*"FUNC COMMENT"*****
208 * ID :
209 * Outline : Sample program main
210 *-----
211 * Include : "iodefine.h"
212 *-----
213 * Declaration : void io_interrupt_dmac_dei4(void);
214 *-----
215 * Description : DMAC チャンネル 4 の転送完了割り込みを処理します。
216 * : リロードレジスタを使用して繰り返し受信します。
217 * : バッファサイズ分のデータを転送したら DMA を停止します。
218 *-----
219 * Argument : void
220 *-----
221 * Return Value : void
222 *-----
223 * Note :
224 *"FUNC COMMENT END"*****/
225 void io_interrupt_dmac_dei4( void )
226 {
227     volatile int dummy;
228
229     /* ==== バッファサイズ分のデータを転送したら DMA を停止する ==== */
230     if( DMAC.DAR4.LONG >= (unsigned long)&rx_buff[ NUM_OF_BUFF ] ){
231         DMAC.CHCR4.BIT.DE = 0; /* 転送禁止 */
232     }
233     /* ==== TE フラグのクリア ==== */
234     DMAC.CHCR4.BIT.TE = 0;
235
236     /* ==== 割り込み要因フラグを確実にクリアするためのダミーリード ==== */
237     dummy=DMAC.CHCR4.LONG;
238
239 }
240 /* End of File */
241
```

3.8 サンプルプログラムリスト "can1.c" (1)

```
1  /*****
2  *  DISCLAIMER
3  *
4  *  This software is supplied by Renesas Electronics Corp. and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics Corp. and is protected under
8  *  all applicable laws, including copyright laws.
9  *
10 *  THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 *  REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 *  INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 *  PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 *  DISCLAIMED.
15 *
16 *  TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 *  ELECTRONICS CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 *  FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 *  FOR ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 *  AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 *
22 *  Renesas reserves the right, without notice, to make changes to this
23 *  software and to discontinue the availability of this software.
24 *  By using this software, you agree to the additional terms and
25 *  conditions found by accessing the following link:
26 *  http://www.renesas.com/disclaimer
27 *****/
28 * (C) 2010 Renesas Electronics Corporation. All rights reserved.
29 * "FILE COMMENT"***** Technical reference data *****
30 * System Name : SH7264 Sample Program
31 * File Name   : can1.c
32 * Abstract    : DMAC+CAN Module Application (Data Frame Receive)
33 * Version     : 1.00.01
34 * Device      : SH7262/SH7264
35 * Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *              : C/C++ compiler package for the SuperH RISC engine family
37 *              : (Ver.9.03 Release00).
38 * OS          : None
39 * H/W Platform: M3A-HS64G50(CPU board)+M3A-HS64G02(IO board)
40 * Description :
41 *****/
42 * History     : Feb.09,2010 ver.1.00.00
43 *              : Apr.15,2010 Ver.1.00.01 Changed the company name
44 * "FILE COMMENT END"*****
45 #include "iodefine.h"      /* SH7264 iodefine */
46
47 /* ---- prototype declaration ---- */
48 void io_init_port_ch1(void);
49 void io_init_can_ch1(void);
50
51 /* ---- symbol definition ---- */
52 #define CAN_IRR0 0x0001
53
```


3.9 サンプルプログラムリスト "can1.c" (2)

```
54  /*"FUNC COMMENT"*****
55  * ID          :
56  * Outline     : 汎用入出力ポートの初期設定
57  *-----
58  * Include     : "iodefine.h"
59  *-----
60  * Declaration : void io_init_port_ch1(void);
61  *-----
62  * Description : 端子機能の設定を行います
63  *             : CRx1 input、CTx1 output
64  *-----
65  * Argument    : void
66  *-----
67  * Return Value : void
68  *-----
69  * Note        :
70  *"FUNC COMMENT END"*****/
71 void io_init_port_ch1(void)
72 {
73     /* ==== Setting of PORT ==== */
74     PORT.PJCR0.BIT.PJ2MD = 0x1;    /* Set CTx1 */
75     PORT.PJCR0.BIT.PJ3MD = 0x1;    /* Set CRx1 */
76 }
77
```

3.10 サンプルプログラムリスト "can1.c" (3)

```
78  /*"FUNC COMMENT"*****
79  * ID      :
80  * Outline : CANの初期設定
81  *-----
82  * Include : "iodefine.h"
83  *-----
84  * Declaration : void io_init_can_ch1(void);
85  *-----
86  * Description : コントローラエリアネットワーク (RCAN)チャンネル1の初期設定を行います
87  *              : 転送レートは1Mbpsに設定しています。メールボックス0の受信データを
88  *              : ダイレクトメモリアクセスコントローラで転送できるように割り込みマスク
89  *              : を許可します。
90  *-----
91  * Argument  : void
92  *-----
93  * Return Value : void
94  *-----
95  * Note      :
96  *"FUNC COMMENT END"*****/
97 void io_init_can_ch1(void)
98 {
99     int i,j;
100
101     /* ==== スタンバイコントロールレジスタ5の設定 ==== */
102     CPG.STBCR5.BIT.MSTP52 = 0;          /* RCANチャンネル1のモジュールスタンバイ解除 */
103
104     /* ==== マスタコントロールレジスタの設定 ==== */
105     RCAN1.MCR.WORD |= 0x0001;          /* リセットモードの設定 */
106     while((RCAN1.IRR.WORD & CAN_IRR0) != CAN_IRR0){
107         /* リセットモード移行完了待ち */
108     }
109     /* ==== IRR = 1, GSR = 1 (自動的に設定) ==== */
110
111     /* ---- リセット割り込みフラグのクリア ---- */
112     RCAN1.IRR.WORD = 0x0001;
113
114     /* ---- マスタコントロールレジスタの設定 ---- */
115     RCAN1.MCR.WORD |= 0x8000;          /* CAN2と別フォーマット */
116
117     /* ---- インタラプトマスクレジスタの設定 ---- */
118     RCAN1.IMR.WORD = 0xffffd;
119
120     /* ---- メールボックスインタラプトマスクレジスタ0の設定 ---- */
121     RCAN1.MBIMR0.BIT.MB0 = 0;
122
```

3.11 サンプルプログラムリスト "can1.c" (4)

```
123  /* ----メールボックスのRAM領域をクリア ---- */
124  for(i = 0; i < 32; i++){
125      RCAN1.MB[i].CONTROL0.LONG = 0x00000000;
126      RCAN1.MB[i].LAFM.LONG = 0x00000000;
127      for(j = 0; j < 8; j++){
128          RCAN1.MB[i].MSG_DATA[j] = 0x00;
129      }
130  }
131  /* ---- メールボックス0の設定 ---- */
132  RCAN1.MB[0].CONTROL1.WORD = 0x0200;          /* MBC=2, dlc=0 */
133  RCAN1.MB[0].CONTROL0.LONG = 0x00280000;     /* standard data frame, id=0x00a */
134  RCAN1.MB[0].LAFM.LONG = 0x00000000;
135  for(i = 0; i < 8; i++){                      /* データクリア */
136      RCAN1.MB[0].MSG_DATA[i] = 0x00;
137  }
138  /* ---- ビットコンフィギュレーションレジスタの設定 ---- */
139  RCAN1.BCR1.WORD = 0x9600;                    /* tsg1=9(10bit),tsg2=6(7bit),sjw=0(1bit),bsp=0 */
140  RCAN1.BCR0.WORD = 0x0000;                    /* 1 Mbps */
141
142  /* ---- インタラプトリクエストレジスタの設定 ---- */
143  RCAN1.IRR.WORD = 0xffff;
144
145  /* ---- マスタコントロールレジスタの設定 ---- */
146  RCAN1.MCR.WORD &= 0xf8fc;                    /* MCR0,MCR1 クリア */
147  while( (RCAN1.GSR.WORD & 0x0008) != 0x0000 ){
148      /* reset state is end */
149  }
150 }
151
152 /* End of File */
153
```

4. 参考ドキュメント

- ソフトウェアマニュアル
SH-2A、SH2A-FPU ソフトウェアマニュアル Rev.3.00
(最新版をルネサスエレクトロニクスのホームページから入手してください。)
- ハードウェアマニュアル
SH7262 グループ、SH7264 グループ ハードウェアマニュアル Rev.2.00
(最新版をルネサスエレクトロニクスの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>

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。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.04.15	—	初版発行

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くと、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. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連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の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。
5. 輸出に際しては、「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きを行ってください。本資料に記載されている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の目的で使用しないでください。また、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に使用することができません。
6. 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。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されてい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生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
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
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、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社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。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、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、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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。
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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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/inquiry>