

# SH7254R グループ

R01AN1170JJ0100

Rev.1.00

2012.05.08

## A-DMAC を使用した RCAN-TL1 動作例

### 要旨

本アプリケーションノートは、専用ダイレクトメモリアクセスコントローラ(A-DMAC)を使用したコントローラエリアネットワーク(RCAN-TL1)動作例をまとめたものです。A-DMAC を用いることによって、CPU を介さずに内蔵周辺モジュールと内蔵 RAM 間のデータ転送を行うことが可能となります。

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

### 動作確認デバイス

SH72546R

### 適用条件

- ・ 統合開発環境 : ルネサス エレクトロニクス製  
High-performance Embedded Workshop Ver.4.09.00
- ・ 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.9.04.00 Release 00
- ・ 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 -object="\$(CONFIGDIR)¥\$(FILELEAF).obj"  
-debug -gbr=auto -chginpath -errorpath -global\_volatile=0  
-opt\_range=all -infinite\_loop=0 -del\_vacant\_loop=0 -struct\_alloc=1  
-nologo

### 目次

1. A-DMAC を使用した RCAN のデータフレーム送受信 .....	2
1.1 仕様 .....	2
1.2 使用機能説明 .....	4
1.3 動作説明 .....	9
1.4 ソフトウェア説明 .....	11
1.5 フローチャート .....	15
1.6 サンプルプログラム .....	23

## 1. A-DMAC を使用した RCAN のデータフレーム送受信

### 1.1 仕様

図 1 に示すように SH72546R を 2 個使用した、データフレームの送受信を行います。

#### (1)送受信 RCAN 共通仕様

- ・ 通信速度は 1Mbps (Pφ=40MHz 時)とします。

#### (2)送信側 RCAN 仕様

- ・ データフレームの送信はメールボックス 1~4 を用い、メッセージ送信プライオリティでメールボックス番号が大きい方から順に送信するように設定します。
- ・ データフレームで送信されるデータ長は 8 バイトとし、スタンダードフォーマットのデータフレームを 4 回送信します。各メールボックスの ID およびデータは表 1 のとおりとします。

#### (3)送信側 A-DMAC 仕様

- ・ 送信待ちレジスタに転送するメールボックス番号をソフトウェアで設定した後、転送許可状態とすると A-DMAC はエイリアス領域に書き込んだデータをメールボックスに転送します。
- ・ 1 つのメールボックスに対して、20 バイトのエイリアス領域が割り当てられています。
- ・ A-DMAC による転送が完了した後、割り込みコントローラに転送完了を通知して転送完了割り込みが発生し、データフレームの送信を開始します。

#### (4)受信側 RCAN 仕様

- ・ メッセージの受信はメールボックス 0 を用います。フィルタマスク機能を用いてメールボックス 1~4 の ID のメッセージを受信するように設定します。

#### (5)受信側 A-DMAC 仕様

- ・ メールボックスフルの通知を A-DMAC の転送要求として用い、A-DMAC はメールボックス 0 のメッセージデータをエイリアス領域に転送します。
- ・ データフレーム受信割り込みを用いて、エイリアス領域に転送されたデータを RAM に退避していません。

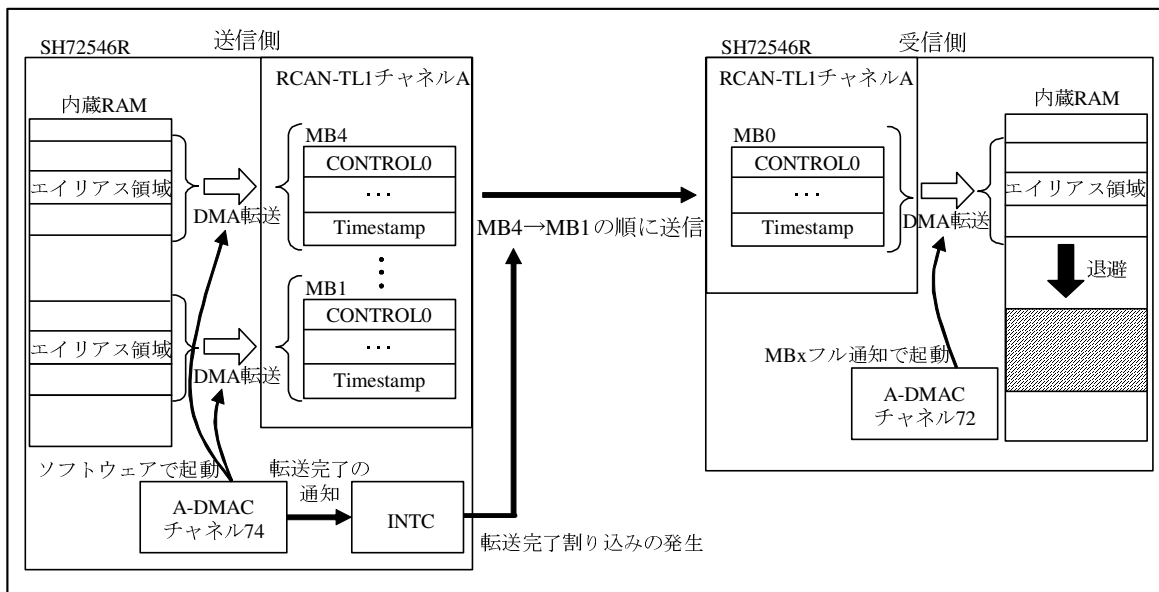


図 1 通信仕様

表 1 メッセージ仕様

メールボックス	スタンダード ID	データ
メールボックス 1	H'0AA	H'22 44 66 88 AA CC EE 00(8byte)
メールボックス 2	H'1BB	H'11 33 55 77 99 BB DD FF(8byte)
メールボックス 3	H'2CC	H'11 22 33 44 55 66 77 88(8byte)
メールボックス 4	H'3DD	H'88 99 AA BB CC DD EE FF(8byte)

## 1.2 使用機能説明

### 1.2.1 機能説明

以下に、A-DMAC および RCAN-TL1 の使用機能について説明を記します。

#### A-DMAC 使用機能

- 専用ダイレクトアクセスコントローラ(A-DMAC)

A-DMAC は、対応する内蔵周辺モジュール(本タスク例では RCAN-TL1)と内蔵 RAM 間のデータ転送を CPU に代わって高速で行う機能です。A-DMAC を使用すると、CPU の負担を減らすとともに LSI の動作効率を上げることができます。本タスク例で用いた RCAN 用チャネルは、RCAN モジュールのメールボックス(MBx)と内蔵 RAM 間の転送を行います。Ch72 はメールボックスから内蔵 RAM への受信転送を行い、Ch74 は内蔵 RAM からメールボックスへの送信転送を行います。また Ch72 は RCAN からのメールボックスフルの通知を転送要求として用いており、Ch74 はソフトウェアにより転送が起動されます。

- エイリアス領域

A-DMAC によるデータ転送で転送元となる内蔵 RAM 上の領域をエイリアス領域と呼びます。エイリアス領域の先頭アドレスは変更することも可能で、本タスク例では H'FFF88000 としています。エイリアス領域の先頭アドレスを基準としたオフセットは RCAN 用チャネルでは固定であり、Ch72 では H'FFF88200～H'FFF88213 までを MB0 から転送するエイリアス領域として使用しています。また Ch74 では H'FFF88220～H'FFF88233 までを MB1 に転送するエイリアス領域として用いており、同様に H'FFF88240～H'FFF88253、H'FFF88260～H'FFF88273、H'FFF88280～H'FFF88293 のエイリアス領域に書き込んだデータはそれぞれ MB2、MB3、MB4 に転送されます。1つのメールボックスあたりロングワード単位で5回転送し、合計20バイトが転送されます。

- 転送完了割り込み

A-DMAC によるデータ転送が完了し、A-DMAC トランスファエンドレジスタ(ADMATE)の TE ビットが1にセットされるタイミングで、INTC に転送完了を通知し転送完了割り込みが発生します。ADMATE の TE ビットをクリアすることで、転送完了割り込みはクリアされます。また転送完了割り込みは、A-DMAC 割り込みコントロールレジスタ(ADMAIE)の各チャネルに対応した IE ビットを0に設定することで禁止することができます。

- A-DMAC オペレーションレジスタ(ADMAOR)

ADMAOR は、全てのチャンネルの DMA 転送を許可または禁止するレジスタです。DME ビットおよび各チャンネルの A-DMAC イネーブルレジスタ (ADMADE) の DE ビットを 1 に設定すると、DMA 転送が許可されます。また、DME ビットをクリアすると、全てのチャンネルの DMA 転送が中断されます。

- A-DMAC エイリアススペースレジスタ(ADMAABR)

ADMAABR は、A-DMAC 用のエイリアス領域の先頭アドレスを指定するレジスタです。

- A-DMAC 割り込みコントロールレジスタ(ADMAIE)

ADMAIE は、チャンネルごとの CPU への割り込みを許可または禁止するレジスタです。IE ビットを 1 に設定した場合、対応するチャンネルの A-DMAC トランスファエンドレジスタ(ADMATE)の TE ビットがセットされると転送完了割り込みを要求します。

- A-DMAC トランスファエンドレジスタ(ADMATE)

ADMATE は、対応するチャンネルごとの DMA 転送の状態を示すレジスタです。ADMATE の TE ビットは、設定された回数の DMA 転送が終了すると 1 にセットされます。

- A-DMAC イネーブルレジスタ(ADMADE)

ADMADE は、ATU-III (タイマ A、C、F) 用チャンネル、RCAN 用チャンネルの DMA 転送を許可または禁止するレジスタです。各チャンネルに対応した DE ビットおよび ADMAOR の DME ビットを 1 に設定すると、DMA 転送を許可します。

- A-DMAC 受信待ちレジスタ(ADMARVPR)

ADMARVPR レジスタは、RCAN モジュールの各メールボックスに対応したフラグを持ち、RCAN からの転送要求に応じてメールボックスのデータをエイリアス領域に転送が完了した際に、メールボックスに対応するフラグが 1 にセットされます。

- A-DMAC 送信待ちレジスタ(ADMATVPR)

ADMATVPR レジスタは、RCAN モジュールの各メールボックスに対応したフラグを持ち、エイリアス領域からメールボックスに転送するメールボックス番号をソフトウェアから指定します。転送するメールボックスに対応するビットが 1 に設定されると、転送許可状態でエイリアス領域からメールボックスへの転送を行います。転送が完了した際は、メールボックスに対応するフラグがハードウェアにより 0 にクリアされます。

### RCAN-TL1 使用機能

#### (a) メールボックス

- メッセージコントロール 0(CONTROL0)

CONTROL0 は、スタンダードフォーマット/エクステンデッドフォーマットの選択、データフレーム/リモートフレームの選択、スタンダード ID(STDID)、エクステンデッド ID(EXTID)の設定を行うレジスタです。

- ローカルアクセプタンスフィルタ(LAFM)

LAFM は IDE、STDID、EXTID のフィルタマスク設定を行うレジスタです。1つのビットがフィルタマスクに設定されている場合、受信メッセージの ID とメールボックスに設定された ID の比較を行う際、受信メッセージの ID に対応するビットは無視されます。またビットがクリアされている場合、受信メッセージの ID に対応するビットは、格納するメールボックスに設定された ID と一致しなくてはなりません。フィルタマスクを使用しない場合は全てのビットを 0 にする必要があります。

- メッセージデータ(DATA)

DATA は、送受信される RCAN-TL1 のメッセージを格納します。

- メッセージコントロール 1(CONTROL1)

CONTROL1 は、ニューメッセージコントロール、データフレーム自動送信、自動再送信無効、メールボックスコンフィギュレーション、送信するデータのバイト数を設定するレジスタです。

#### (b) コントロールレジスタ

- マスタコントロールレジスタ(MCR)

MCR は RCAN-TL1 の動作を制御するレジスタで、ID 並び替えの順序、自動バスオフホルト、バスオフ時ホルト、テストモード、CAN スリープモード、自動ウェイクモード、メッセージ送信プライオリティ、ホルトリクエスト、リセットリクエストの設定を行います。

- ジェネラルステータスレジスタ(GSR)

GSR は RCAN-TL1 の状態を示すレジスタで、ホルト/スリープ、リセット状態、エラーパッシブ、エラーワーニング、バスオフ状態の判別が可能です。

- ビットコンフィギュレーションレジスタ (BCR)

BCR は CAN のビットタイミングパラメータとボーレートプリスケアラを設定するレジスタで、周辺バスクロックに対するビットレートを決定します。

- インタラプトリクエストレジスタ(IRR)

IRR は、各種割り込み要因のステータスフラグで構成されています。

- インタラプトマスクレジスタ(IMR)

IMR は、IRR の各ビットに対応する割り込み要求をマスクするレジスタです。ビットを 1 に設定すると、IRR の対応するビットがセットされてもその割り込み信号は生成されません。

## (c) メールボックスレジスタ

## ● 送信待ちレジスタ(TXPR)

TXPR は、各ビットに対応するメールボックスのメッセージを送信待ち状態にするレジスタです。複数のビットがセットされた場合の送信順序は、MCR2 ビットの設定によりメッセージID 優先順、またはメールボックス番号順に送信されます。

## ● データフレーム受信完了レジスタ(RXPR)

RXPR は、受信用に設定されたメールボックスがデータフレームを受信したことを示すフラグで構成されたレジスタです。データフレームが正常に受信メールボックスに格納されると、RXPR の各メールボックスに対応するビットが 1 にセットされます。

## ● メールボックスインタラプトマスクレジスタ(MBIMR)

MBIMR は、メールボックスの動作に関連する割り込み要求(IRR1：データフレーム受信割り込み、IRR2：リモートフレーム受信割り込み、IRR8：メールボックスエンプティ割り込み、IRR9：メッセージオーバーラン/オーバーライト割り込み)をマスクするレジスタです。各メールボックスに対応するビットに 1 を書き込むことでマスクが設定されます。メールボックスが受信に設定されている場合、受信割り込みフラグ(IRR1、IRR2、IRR9)による割り込みをマスクします。メールボックスが送信に設定されている場合は、送信や送信アポート(IRR8)による割り込みをマスクします。

## 1.2.2 機能割り付け

表 2 に本タスク例の機能割り付けを示します。

表 2 機能割り付け

A-DMAC		機能
レジスタ	ADMAOR	全てのチャンネルの DMA 転送を許可または禁止します。
	ADMAABR	エイリアス領域の先頭アドレスを指定します。
	ADMAIE	チャンネルごとに CPU への転送完了割り込みを許可または禁止します。
	ADMATE	RCAN 用チャンネルの DMA 転送の状態を示します。
	ADMADE	ATU-Ⅲチャンネル、RCAN 用チャンネルの DMA 転送を許可または禁止します。
	ADMARVPR	メールボックスからエイリアス領域の転送が完了した際にフラグがセットされます。
	ADMATVPR	エイリアス領域からメールボックスに転送するメールボックス番号を指定します。
RCAN-TL1		機能
メールボックス	CONTROL0	メールボックスのフォーマット、フレームの種類、ID を設定します。
	LAFM	メールボックスの IDE、STDID、EXTID のフィルタマスク機能を設定します。
	DATA	送受信される CAN メッセージを格納します。
	CONTROL1	データフレームで送信するデータのバイト数を設定します。
コントロールレジスタ	MCR	RCAN-TL1 の動作モードを制御します。
	GSR	RCAN-TL1 の状態を示します。
	BCR	周辺バスクロックに対するビットレートを設定します。
	IRR	各種割り込み要因のステータスフラグです。
	IMR	IRR の各ビットに対応する割り込み要因をマスクします。
メールボックスレジスタ	TXPR	メールボックスに格納したメッセージを送信待ち状態にします。
	RXPR	メールボックスがデータフレームを正常に受信したことを示します。
	MBIMR	各メールボックスの割り込み要求を許可します。
PFC レジスタ	PJCR1	端子機能を設定します。
INTC レジスタ	IPR28	RCANA の割り込み優先レベルを設定します。
	IPR29	データ転送完了割り込みの優先レベルを設定します。

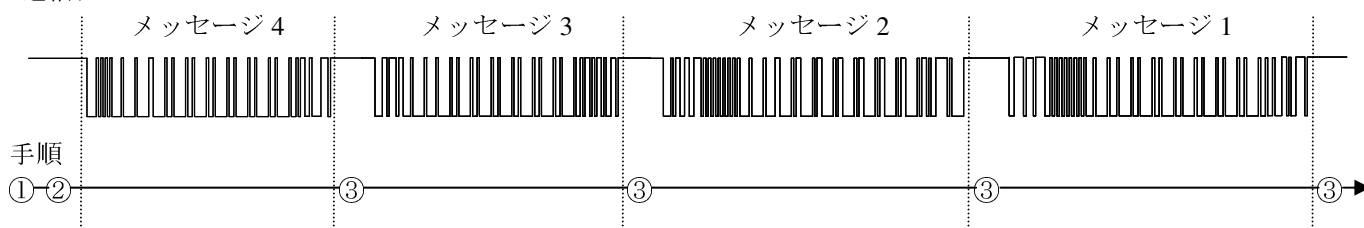


## 1.3 動作説明

図 2、3 に送信時ならびに受信時の動作原理を示します。以下に示すようなハードウェア処理およびソフトウェア処理により、A-DMAC および RCAN-TL1 を用いてデータフレームの送受信を行います。

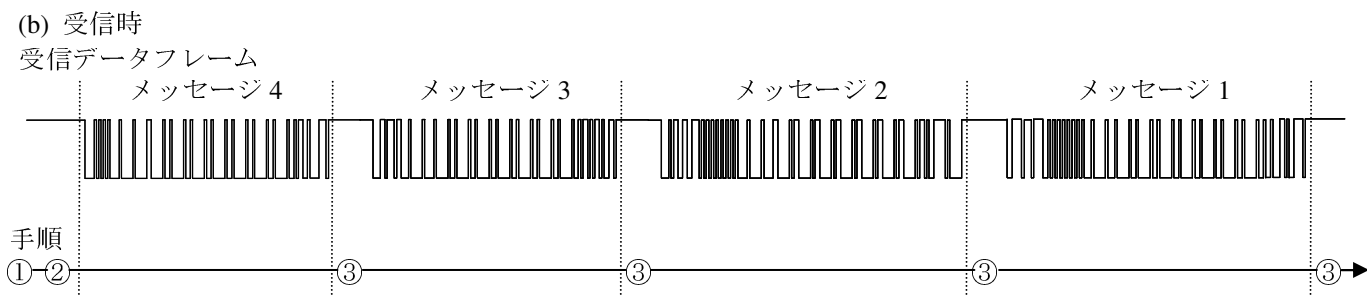
## (a) 送信時

送信データフレーム



	①RCAN 初期設定	②RCAN 用 ADMAC の設定	③送信処理
送信側 ハードウェア 処理	(a)リセットモードに遷移	(a)通常動作状態に設定 (b)エイリアス領域から MB1-4 に メールボックスの設定値を転送 (c)転送完了した MBx に対応する ADMATVPR のビットがクリア (d)ADMATE の TE ビットをセット (e)データ転送完了割り込みの発生	(a) ACK の受信 (b)送信完了フラグのセット
送信側 ソフトウェア 処理	(a)RCAN 送受信端子の設定 (b)割り込み優先レベルの設定 (c)リセットリクエストの設定 (d)メッセージ送信順序の設定 (e)割り込みの許可 (f)メールボックスの初期化 (g)ビットレートの設定 (g)リセットモードの解除	(a)エイリアス領域の先頭番地設定 (b)MB1-4 に転送する送信データ設定 (c)MB1-4 への DMA 転送要求 (d)転送完了割り込みの許可 (e)DMA 転送の許可 (f)MB1-4 を送信待ち状態に設定	

図 2 A-DMAC を用いた RCAN データフレーム送信時の動作



	①RCAN 初期設定	②RCAN 用 A-DMAC の設定	③受信処理
受信側 ハードウェア 処理	(a)リセットモードに遷移	(a)通常動作状態に設定	(a)ACK の送信 (b)受信したメッセージを MB0 に格納 (c)受信完了フラグのセット (d)データフレーム受信割り込みの発生 (e)MB0 に受信したデータを エイリアス領域に転送 (f)ADMARVPR のフラグビットを 1 にセット (g)受信完了フラグのクリア
受信側 ソフトウェア 処理	(a)RCAN 送受信端子の設定 (b)割り込み優先レベルの設定 (c)リセットリクエストの設定 (d)割り込みの許可 (e)メールボックスの初期化 (f)受信用メールボックスの設定 (g)ビットレートの設定 (h)リセットモードの解除	(a)エイリアス領域の先頭番地設定 (b) DMA 転送の許可	(a)エイリアス領域に転送された ID とデータを RAM に格納 (b)ADMARVPR のフラグビットを クリア

図 3 A-DMAC を用いた RCAN データフレーム受信時の動作

## 1.4 ソフトウェア説明

## 1.4.1 モジュール説明

本タスク例のモジュールを表 3 に示します。

表 3 モジュール説明

モジュール名	ラベル名	機能
RCAN 送受信端子設定	init_io_RCAN	RCAN-TL1 の送受信端子を設定します。
SH72546R 割り込み設定	init_INTC	RCANA と A-DMAC の割り込み優先レベルを設定します。
RCANA 初期設定	init_RCANA	RCAN-TL1 のリセットリクエストの設定、各割り込み要求の許可、メールボックスの初期化、リセット解除を行います。
RCAN 用 A-DMAC 初期設定	init_admac_rcan	エイリアス領域の先頭アドレスの設定、転送完了割り込みおよび DMA 転送の許可を行います。
A-DMAC 転送完了割り込み	int_te74	A-DMAC によるデータ転送完了後に ADMATE の TE ビットが 1 にセットされると発生し、メッセージの送信を行います。
データフレーム受信割り込み	int_rma0	メールボックス 0 からエイリアス領域に転送された受信データと ID を RAM に格納します。
エラー割り込み	int_ersa	RCAN-TL1 が受信または送信エラーカウンタによってエラーワーニング状態、エラーパッシブ状態になった場合に発生します。

## 1.4.2 使用変数の説明

本タスク例の使用変数を表 4 に示します。

表 4 使用変数の説明

ラベル名	機能	モジュール名
MBbuff_chA_MB1 MBbuff_chA_MB2 MBbuff_chA_MB3 MBbuff_chA_MB4	メールボックス 1~4 にデータを転送するエイリアス領域を表しています。H'FFF88220~H'FFF88233 までをメールボックス 1 に転送するエイリアス領域として用いており、同様に H'FFF88240~H'FFF88253、H'FFF88260~H'FFF88273、H'FFF88280~H'FFF88293 のエイリアス領域に保存したデータはそれぞれメールボックス 2、3、4 に転送されます。	RCAN 用 A-DMAC 初期設定
MBbuff[4].ID	エイリアス領域に転送されたメッセージの ID を格納する RAM エリアです。	データフレーム 受信割り込み
MBbuff[4].DATA	エイリアス領域に転送されたメッセージのデータを格納する RAM エリアです。	
Rcv_cnt	データフレーム受信割り込みが発生するごとにカウントアップしていき、受信メッセージ数をカウントするフラグになります。	

## 1.4.3 使用内部レジスタの説明

本タスク例で用いた送信側と受信側の使用内部レジスタをそれぞれ表 5 と表 6 に示します。

表 5 使用内部レジスタの説明 (送信側)

レジスタ名	機 能	設定値	モジュール名
PORTJ.CR1	ポート J 1 端子を入力端子 CRx_A、ポート J0 端子を出力端子 CTx_A に設定します。	0x000A	RCAN 送受信端子設定
IPR28	RCANA の割り込みの優先レベルを 14 に設定します。	0xE000	SH72546R 割り込み設定
IPR29	A-DMAC の転送完了割り込みの優先レベルを 15 に設定します。	0xF000	
MCR	RCAN-TL1 のリセットリクエストを設定します。	0x0001	RCANA 初期設定
GSR	RCAN-TL1 がリセット状態であることを確認します。	—	
IRR	RCAN-TL1 がリセットモードに遷移したことを確認します。	—	
IRR	リセット/ホルト/スリープ割り込みフラグをクリアします。	0x0001	
MCR	ID 並びを HCAN2 と異なる順序、メッセージをメールボックス番号の大きい方から順に送信するように設定します。	0x8040	
IMR	エラーパッシブ割り込み、送信エラーカウンタワーニング割り込みを許可します。	0xFFD7	
BCR1	Pφ=40MHz 時、1Mbps に設定します。 (TSEG1=5(6tq)、TSEG2=2(3tq)、SJW=0、BSP=0、BRP=1)	0x5200	
BCR0		0x0001	
MCR	RCAN-TL1 のリセットモードリクエストをクリアします。	0xFFFFE	
GSR	RCAN-TL1 のリセットが解除されたことを確認します。	—	
ADMAABR	エイリアス領域の先頭アドレスを H'FFF88000 に指定します。	0x01	RCAN 用 A-DMAC 初期設定
MBbuff_chA_MB1.ID	メールボックス 1 をスタンダードフォーマット、データフレームに設定します。また、スタンダード ID(H'0AA)を設定します。スタンダード ID は、MBbuff_chA_MB1.ID ロングワードデータの 18~28 番目の 10 ビットです。	0x02A80000	
MBbuff_chA_MB1.CONTROL1	メールボックス 1 を送信用、送信データを 8 バイトに設定します。	0x0008	
MBbuff_chA_MB1.DATA	メールボックス 1 の送信データ(H'22 44 66 88 AA CC EE 00)を設定します。	0x22446688 0x AACCEE00	
MBbuff_chA_MB2.ID	メールボックス 2 をスタンダードフォーマット、データフレームに設定します。また、スタンダード ID(H'1BB)を設定します。スタンダード ID は、MBbuff_chA_MB2.ID ロングワードデータの 18~28 番目の 10 ビットです。	0x06EC0000	
MBbuff_chA_MB2.CONTROL1	メールボックス 2 を送信用、送信データを 8 バイトに設定します。	0x0008	
MBbuff_chA_MB2.DATA	メールボックス 2 の送信データ(H'11 33 55 77 99 BB DD FF)を設定します。	0x11335577 0x99BDDFF	
MBbuff_chA_MB3.ID	メールボックス 3 をスタンダードフォーマット、データフレームに設定します。また、スタンダード ID(H'2CC)を設定します。スタンダード ID は、MBbuff_chA_MB3.ID ロングワードデータの 18~28 番目の 10 ビットです。	0x0B300000	

MBbuff_chA_MB3.CONTROL1	メールボックス 3 を送信用、送信データを 8 バイトに設定します。	0x0008	
MBbuff_chA_MB3.DATA	メールボックス 3 の送信データ(H'11 22 33 44 55 66 77 88)を設定します。	0x11223344 0x55667788	
MBbuff_chA_MB4.ID	メールボックス 4 をスタンダードフォーマット、データフレームに設定します。また、スタンダード ID(H'3DD)を設定します。スタンダード ID は、MBbuff_chA_MB4.ID ロングワードデータの 18~28 番目の 10 ビットです。	0x0F740000	RCAN 用 A-DMAC 初期設定
MBbuff_chA_MB4.CONTROL1	メールボックス 4 を送信用、送信データを 8 バイトに設定します。	0x0008	
MBbuff_chA_MB4.DATA	メールボックス 4 の送信データ(H'88 99 AA BB CC DD EE FF)を設定します。	0x8899AABB 0xCCDDEEFF	
ADMATVPR1	エイリアス領域からメールボックス 1~4 にデータ転送するように設定します。転送が完了すると 0 にクリアされます。	0x001E	
ADMAIE9	RCAN 用チャネル(Ch74)の CPU への転送完了割り込みを許可します。	0x04	
ADMADE7	RCAN 用チャネル(Ch74)の DMA 転送を許可します。	0x04	A-DMAC 転送完了 割り込み
ADMAOR	全てのチャネルの DMA 転送を許可します。	0x01	
ADMAIE9	RCAN 用チャネル(Ch74)の CPU への転送完了割り込みを禁止します。	0x00	
ADMADE7	RCAN 用チャネル(Ch74)の DMA 転送を禁止します。	0x00	
ADMATE2	メールボックス 1~4 への転送が完了すると 1 にセットされます。	—	
TXPR	メールボックス 1~4 を送信待ち状態に設定します。	0x0000001E	エラー 割り込み
IRR	エラーパッシブ割り込みフラグ、送信エラーワーニング割り込みフラグをクリアします。(クリア条件: 1 ライト)	0x0000	

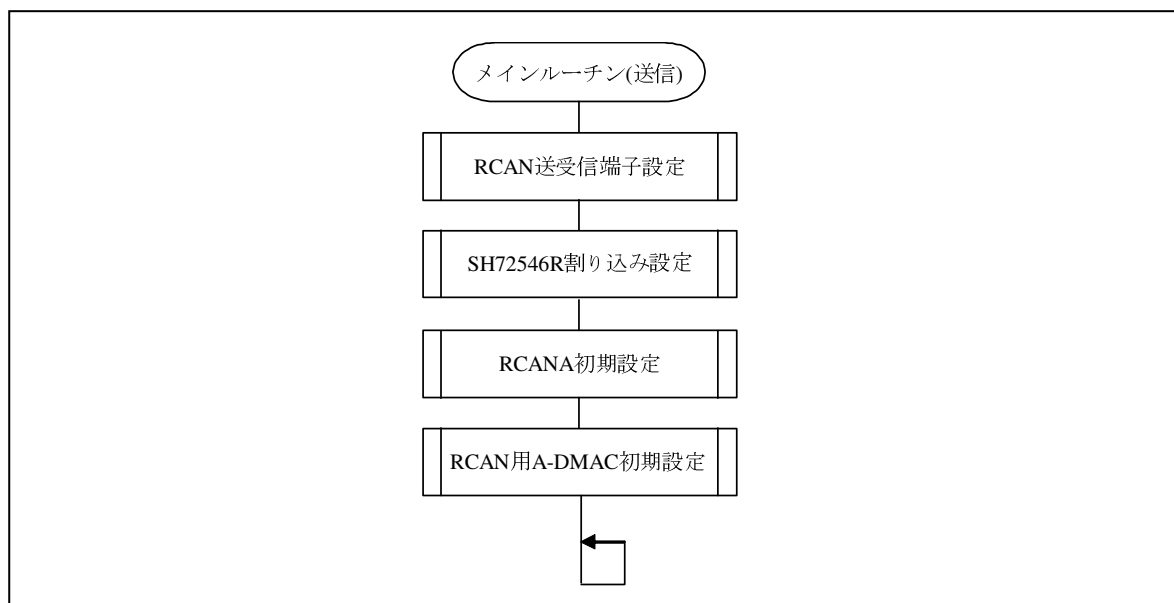
表 6 使用内部レジスタの説明 (受信側)

レジスタ名	機能	設定値	モジュール名
PORTJ.CR1	ポート J1 端子を入力端子 CRx_A、ポート J0 端子を出力端子 CTx_A に設定します。	0x000A	RCAN 送受信端子設定
IPR28	RCANA の割り込みの優先レベルを 15 に設定します。	0xF000	SH72546R 割り込み設定
MCR	RCAN-TL1 のリセットリクエストを設定します。	0x0001	RCANA 初期設定
GSR	RCAN-TL1 がリセット状態であることを確認します。	—	
IRR	RCAN-TL1 がリセットモードに遷移したことを確認します。	—	
IRR	リセット/ホルト/スリープ割り込みフラグをクリアします。	0x0001	
MCR	ID 並びを HCAN2 と異なる順序に設定します。	0x8000	
IMR	エラーパッシブ割り込み、受信エラーカウンタワーニング割り込み、データフレーム受信割り込みを許可します。	0xFFCD	
MBIMR	メールボックス 0 を受信割り込み許可に設定します。	0xFFFFE	
MSG[0].CONTROL0	メールボックス 0 をスタンダードフォーマット、データフレームに設定します。	0x00000000	
MSG[0].LAFM	メールボックス 0 のスタンダード ID についてフィルタマスク機能を有効にします。	0x1FFC0000	
MSG[0].CONTROL1	メールボックス 0 を受信用に設定します。	0x0200	
BCR1	Pφ=40MHz 時、1Mbps に設定します。 (TSEG1=5(6tq)、TSEG2=2(3tq)、SJW=0、BSP=0、BRP=1)	0x5200	
BCR0		0x0001	
MCR	RCAN-TL1 のリセットモードリクエストをクリアします。	0xFFFFE	
GSR	RCAN-TL1 のリセットが解除されたことを確認します。	—	
ADMARVPR1	メールボックス 0 のデータがエイリアス領域に転送されると対応するビットが 1 にセットされるのでクリアします。	0x0000	データフレーム受信割り込み
IRR	エラーパッシブ割り込みフラグ、受信エラーワーニング割り込みフラグをクリアします。(クリア条件: 1 ライト)	0x0000	エラー割り込み

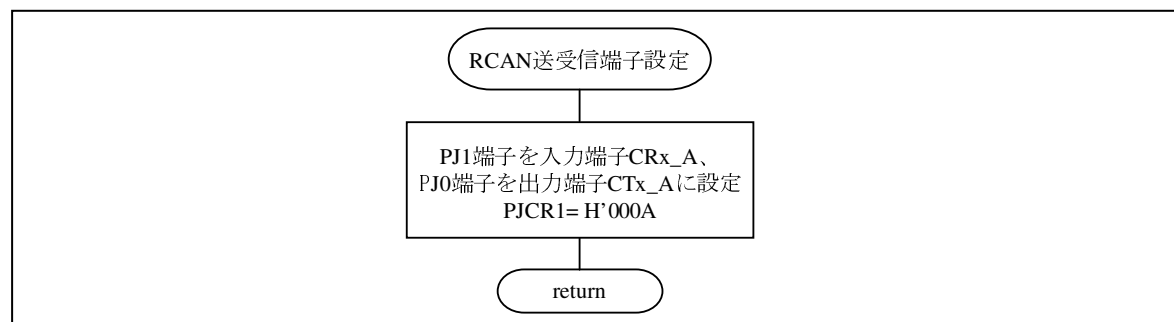
## 1.5 フローチャート

## 1.5.1 送信側フローチャート

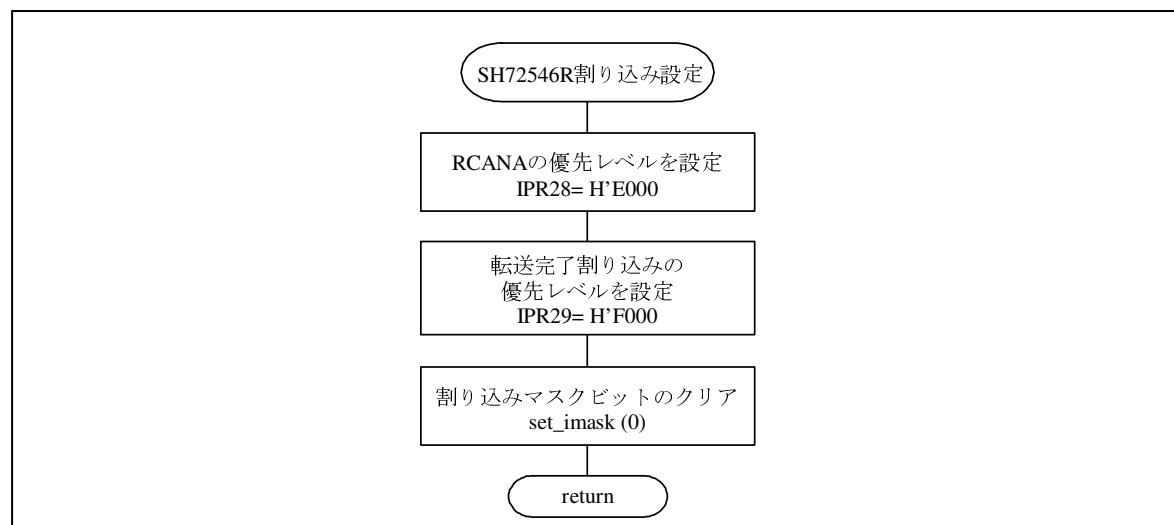
## (a) メインルーチン



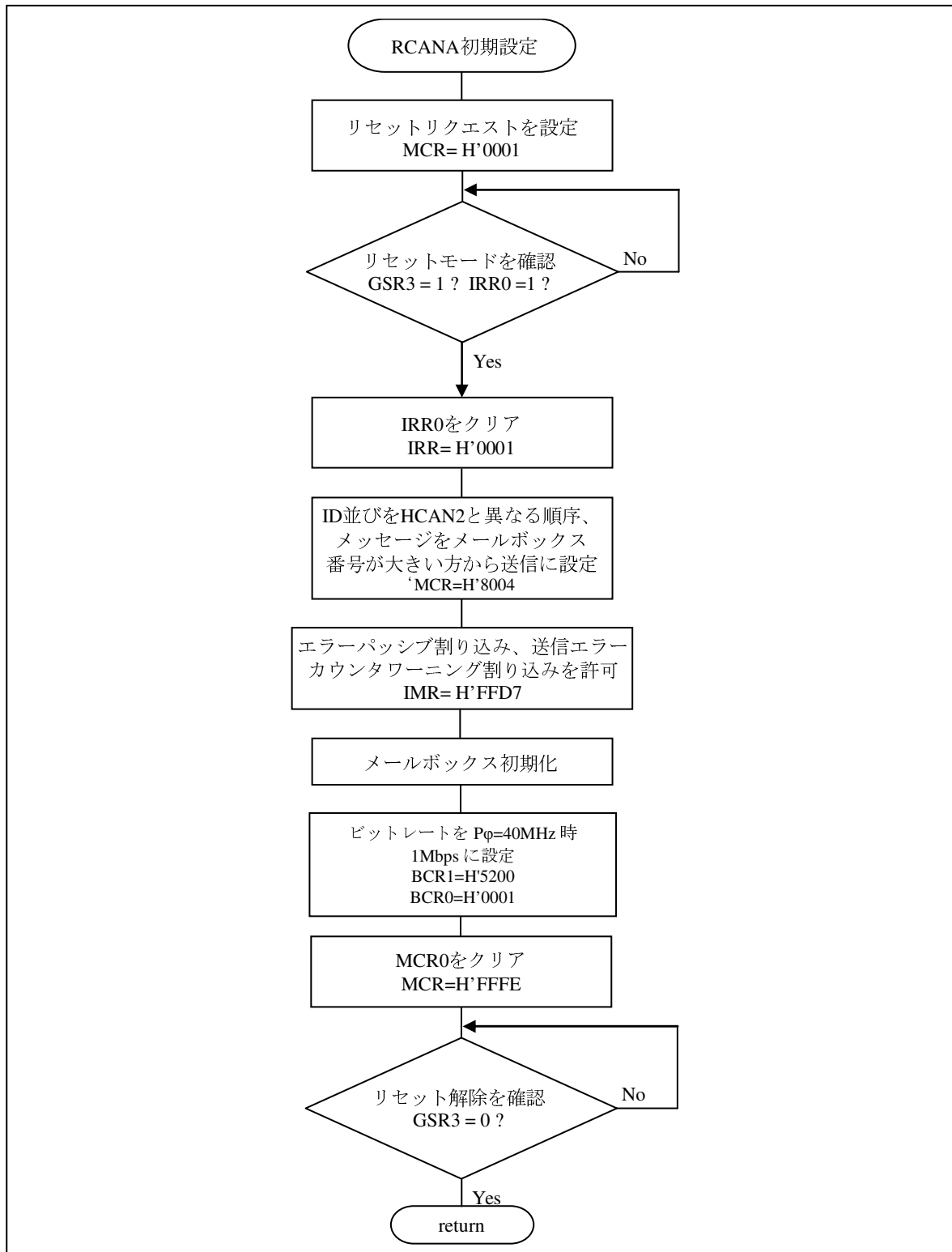
## (b) RCAN 送受信端子設定ルーチン



## (c) SH72546R 割り込み設定ルーチン

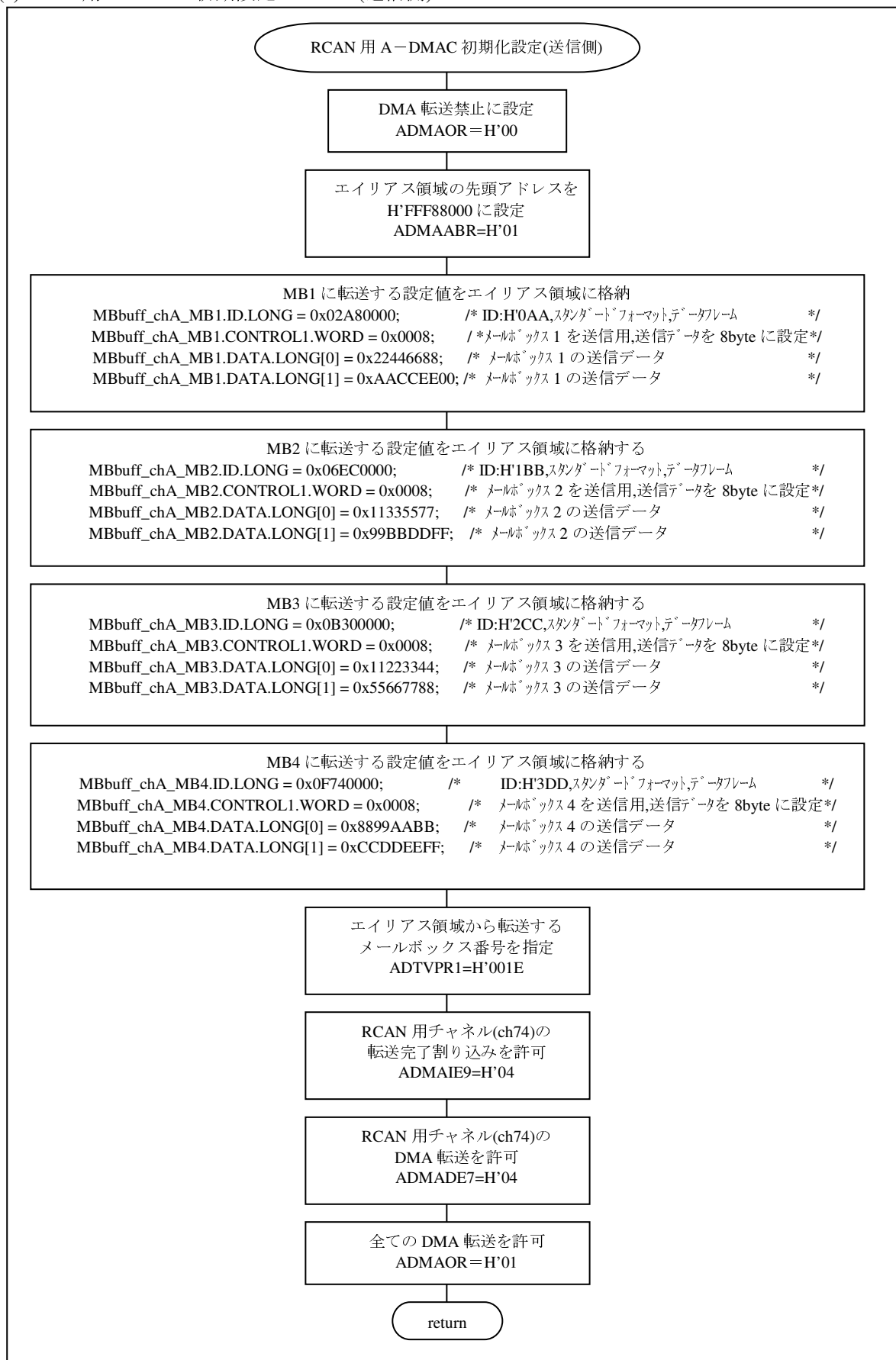


## (d)RCANA 初期設定ルーチン





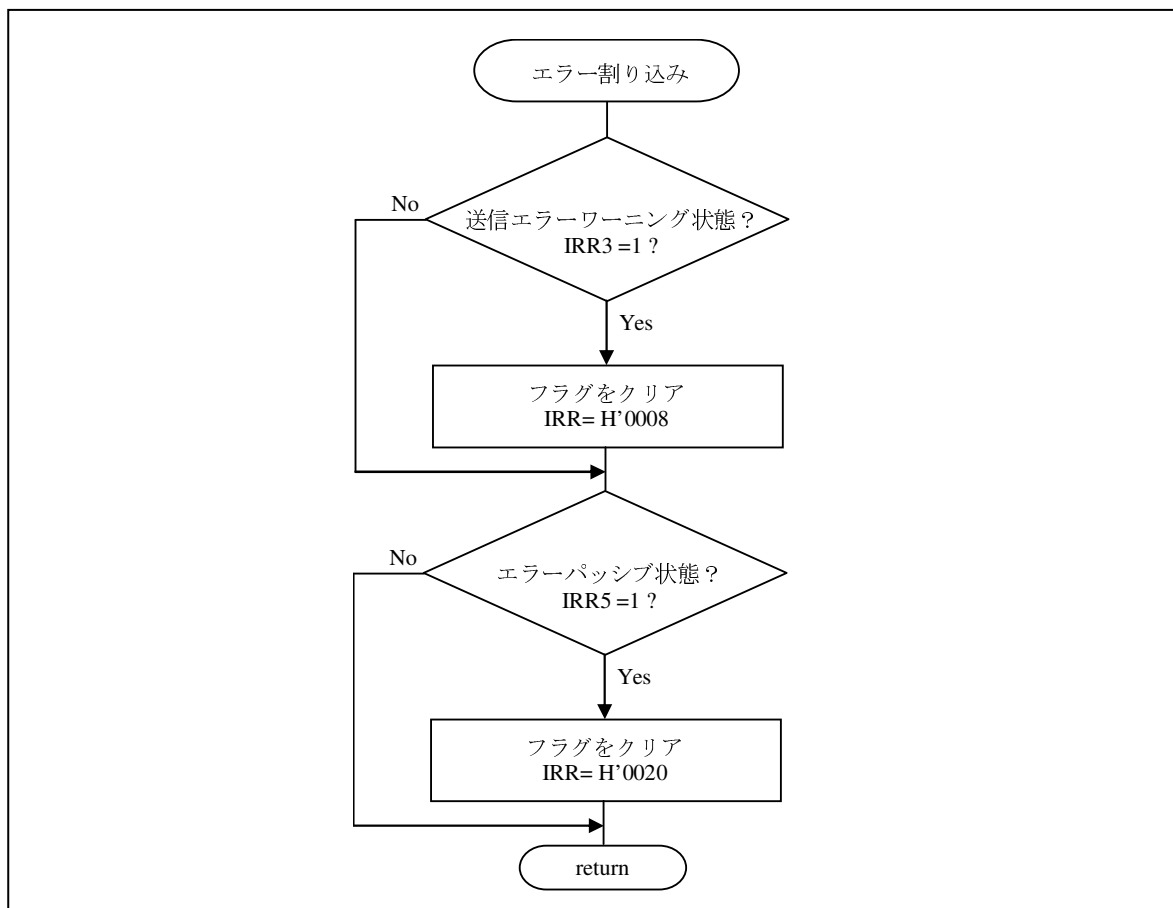
## (e)RCAN 用 A-DMAC 初期設定ルーチン(送信側)



## (f)A-DMAC 転送完了割り込みルーチン

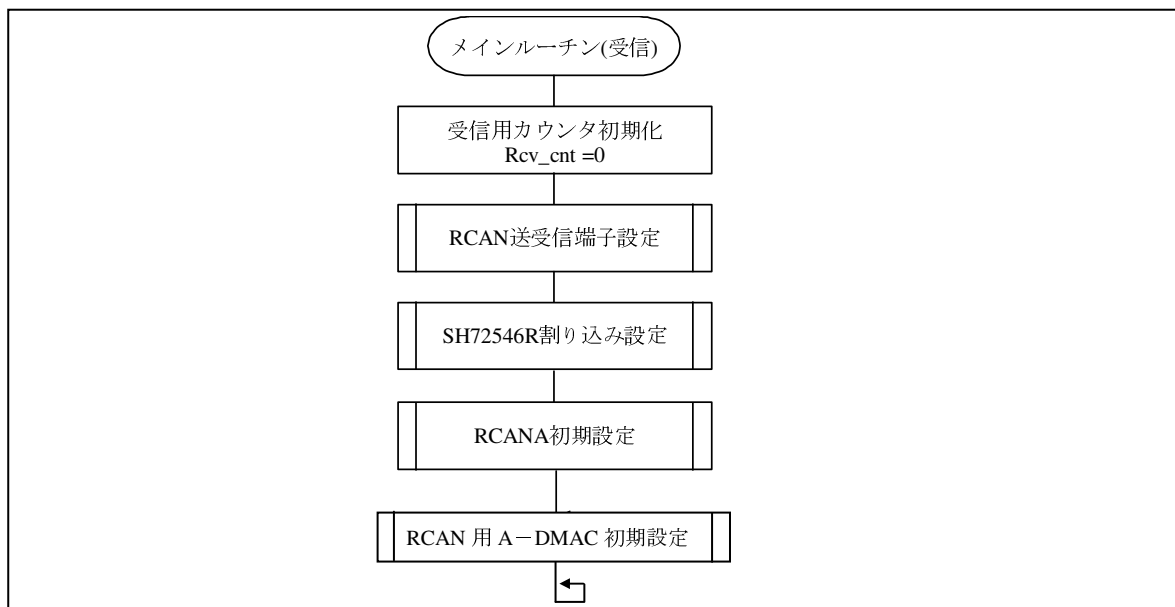


## (g)エラー割り込みルーチン

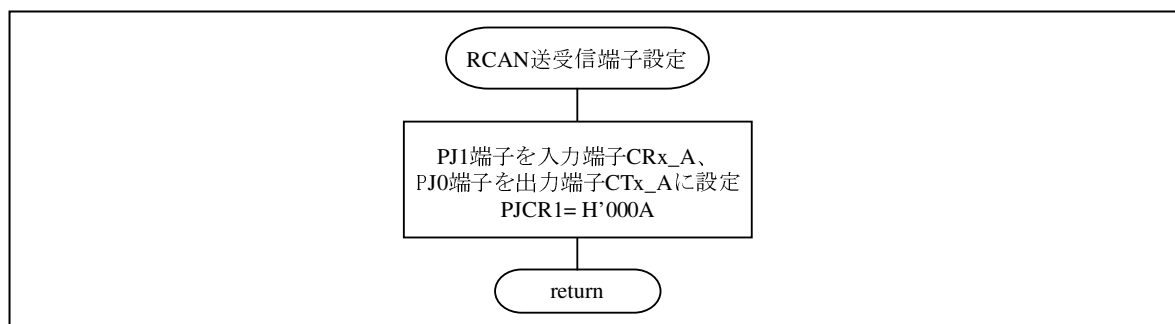


## 1.5.2 受信側フローチャート

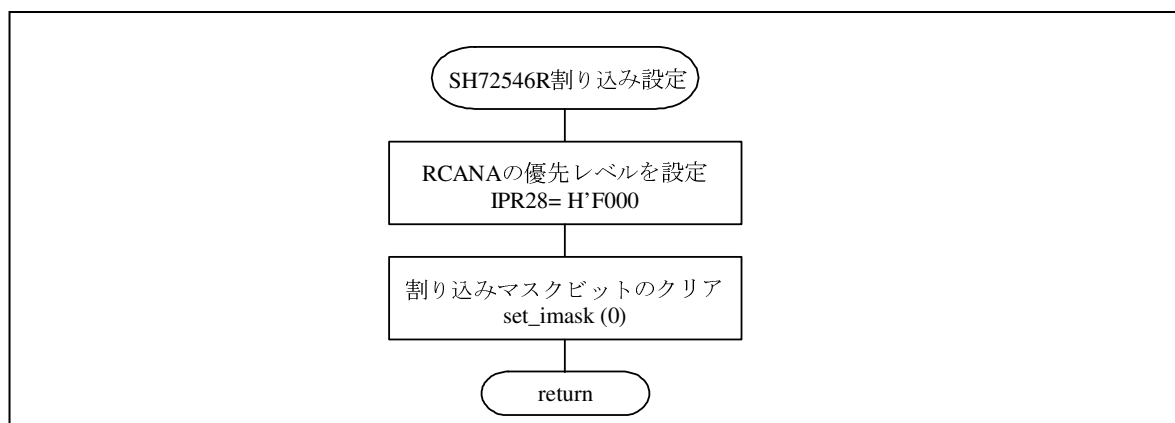
## (a) メインルーチン



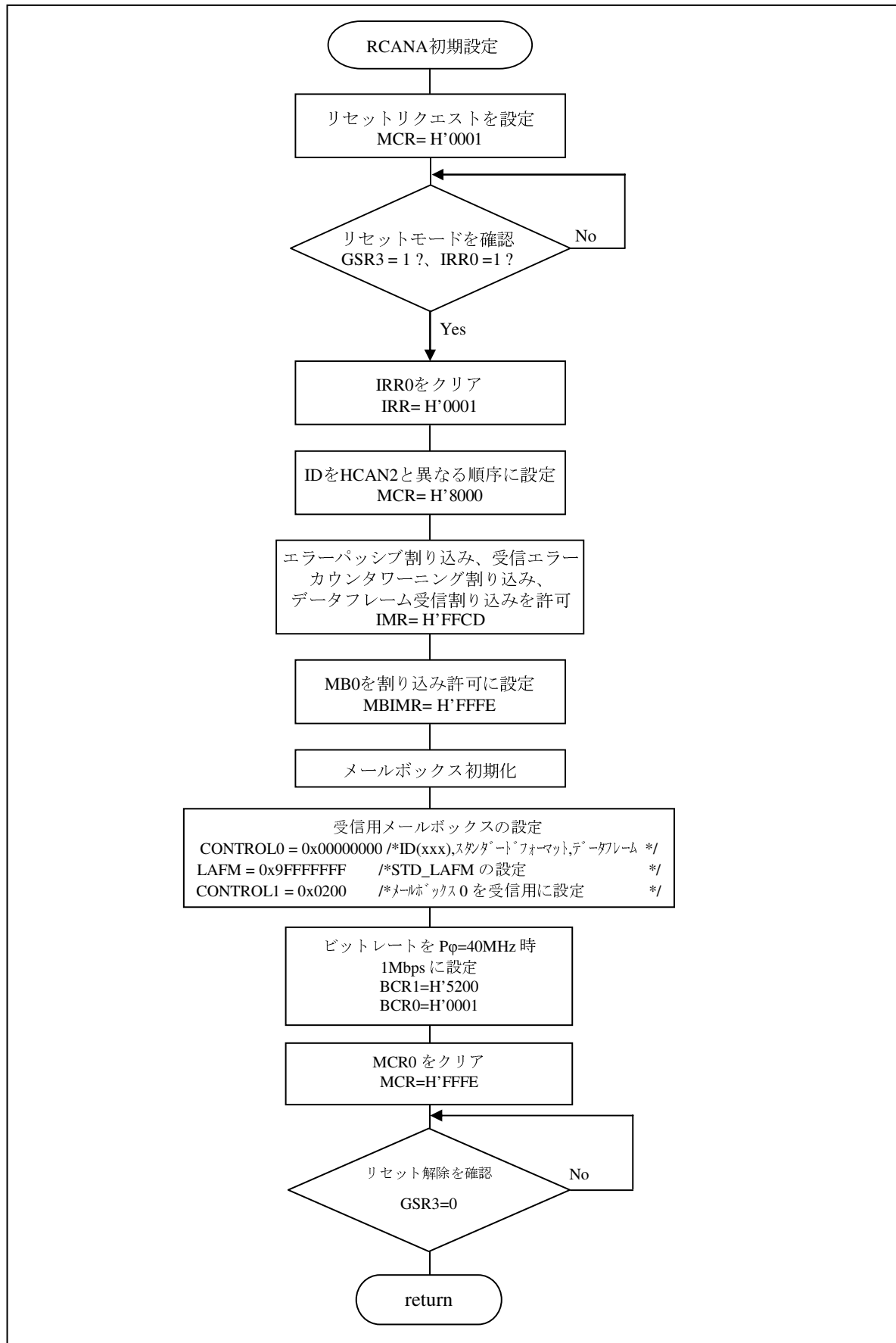
## (b) RCAN 送受信端子設定ルーチン



## (c) SH72546R 割り込み設定ルーチン



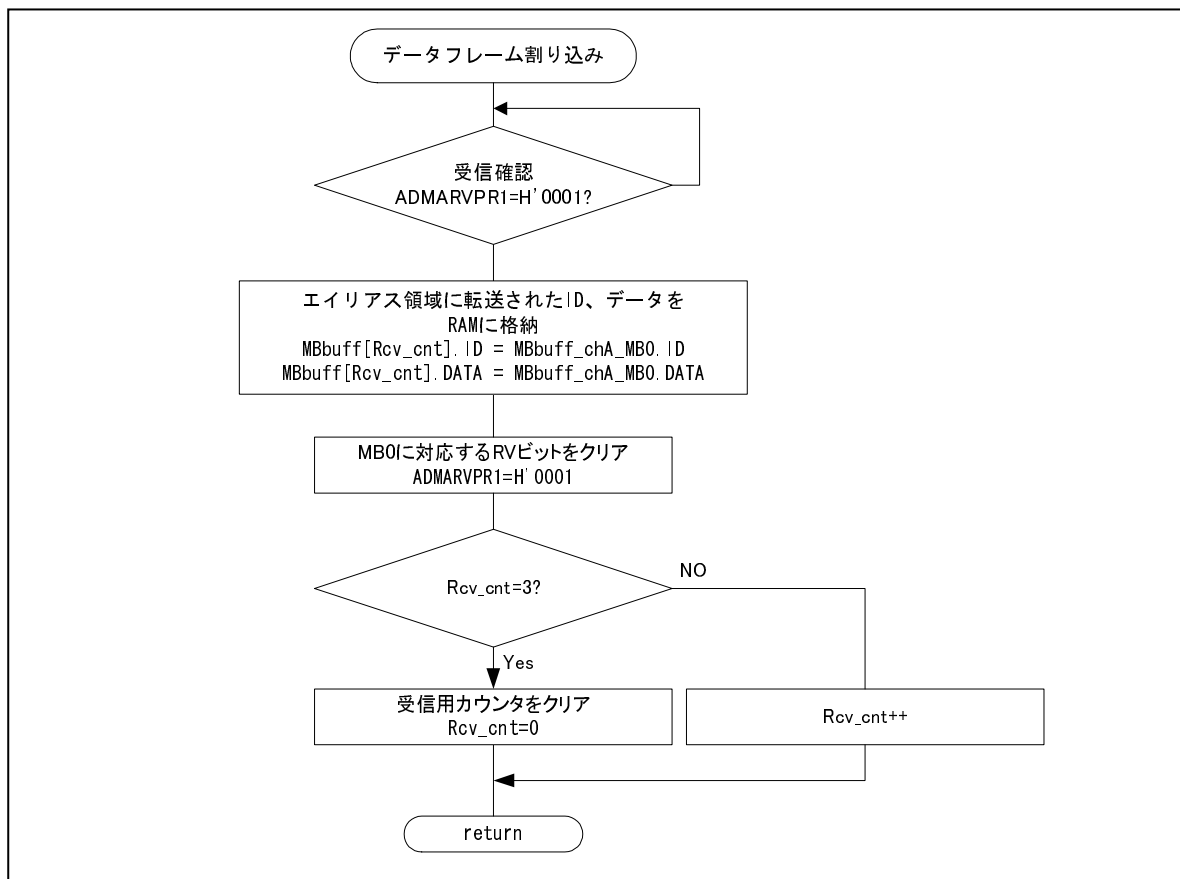
(d) RCANA 初期設定ルーチン



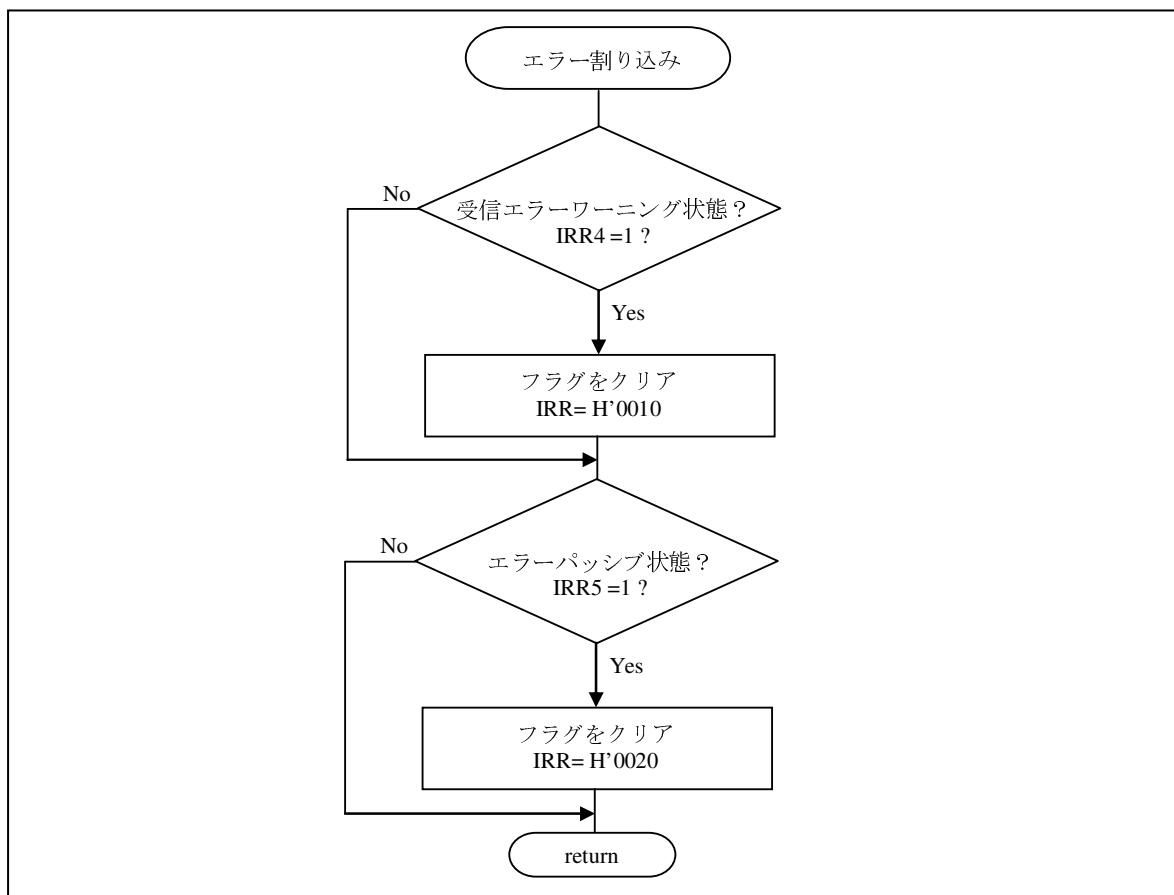
## (e) A-DMAC 初期化ルーチン(受信側)



## (f) データフレーム受信割り込みルーチン



## (g) エラー割り込みルーチン



## 1.6 サンプルプログラム

## (a) 送信側プログラムリスト

```

/*****
 * DISCLAIMER
 * This software is supplied by Renesas Electronics Corporation and is only
 * intended for use with Renesas products. No other uses are authorized. This
 * software is owned by Renesas Electronics Corporation and is protected under
 * all applicable laws, including copyright laws.
 * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
 * THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
 * LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
 * ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
 * ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
 * BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
 * Renesas reserves the right, without notice, to make changes to this software
 * and to discontinue the availability of this software. By using this software,
 * you agree to the additional terms and conditions found by accessing the
 * following link:
 * http://www.renesas.com/disclaimer *
 * Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
 *****/
/*****
 * File Name      : SH7254R.c
 * Version        : 1.00
 * Device(s)      : SH72546R
 * Tool-Chain     : High-performance Embedded Workshop (Ver.4.08.00).
 * OS              : None
 * H/W Platform   : SH7254R
 * Description     : This is the main tutorial code.
 * Operation      : RCAN_TRM
 * Limitations    : None
 *****/
/*****
 * History : DD.MM.YYYY Version Description
 *          : 23.08.2011 1.00   First Release
 *****/
/*****
Includes <System Includes> , "Project Includes"
 *****/
#include <machine.h> /* ライブラリ関数用ヘッダファイル */
#include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
/*****
Macro definitions
 *****/
#define MBbuff_chA_MB1 (*(struct st_mbbuff*)0xFFFF88220) /* RCANA_MB1 エリア領域 */
#define MBbuff_chA_MB2 (*(struct st_mbbuff*)0xFFFF88240) /* RCANA_MB2 エリア領域 */
#define MBbuff_chA_MB3 (*(struct st_mbbuff*)0xFFFF88260) /* RCANA_MB3 エリア領域 */
#define MBbuff_chA_MB4 (*(struct st_mbbuff*)0xFFFF88280) /* RCANA_MB4 エリア領域 */
/*****
Private global variables and functions
 *****/
void main(void );
void init_RCANA(void); /* RCAN_A の初期化 */
void init_INTC(void); /* 割り込み優先レベルの設定 */

```

```

void init_io_RCAN(void);          /* RCAN で使用する端子の設定          */
void init_admac_rcan(void);      /* A-DMAC ch74 (RCAN 送信用) の設定    */
unsigned char i, j;
/*****
        RCAN 用マールボックスの定義
*****/
struct st_mbbuff
{
    union
    {
        unsigned long LONG;
        struct
        {
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;

    union
    {
        unsigned long LONG;
        struct
        {
            unsigned short H;
            unsigned short L;
        }WORD;
    }LAFM;

    union
    {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;

    union
    {
        unsigned short WORD;
        struct
        {
            unsigned short H;
            unsigned short L;
        }BYTE;
    }CONTROL1;

    unsigned short TimeStamp;
};

/*****
* Function Name: main
* Description   : The main loop Main function
* Arguments    : none
* Return Value : none
*****/
void main(void)
{
    init_io_RCAN();          /* RCAN 送受信端子の設定          */
    init_INTC();            /* 割り込みレベルの設定          */
}

```



```

    init_RCANA();                               /* RCANA の初期設定          */
    init_admac_rcan();                          /* RCAN 用 A-DMAC の初期設定 */
    while(1);
} /* End of function main() */

/*****
* Function Name: init_io_RCAN
* Description  : RCAN 用端子の設定
* Arguments   : none
* Return Value: none
*****/
void init_io_RCAN(void)
{
    /* Configure PJCR1
    b15,14 PJ7MD[1:0] = 0 PJ7 入出力 (ポート)
    b13     リザーブビット
    b12     PJ6MD = 0     PJ6 入出力 (ポート)
    b11     リザーブビット
    b10     PJ5MD = 0     PJ5 入出力 (ポート)
    b9,8 PJ4MD[1:0] = 0 PJ4 入出力 (ポート)
    b7,6 PJ3MD[1:0] = 0 PJ3 入出力 (ポート)
    b5,4 PJ2MD[1:0] = 0 PJ2 入出力 (ポート)
    b3,2 PJ1MD[1:0] = 2 CRx_A 入力 (RCAN-TL1)
    b1,0 PJ0MD[1:0] = 2 CTx_A 出力 (RCAN-TL1) */
    PORTJ.CR1.WORD |= 0x000A;                  /* PJ0:CTx_A 出力,PJ1:CRx_A 入力に設定*/
} /* End of function init_io_RCAN() */

/*****
* Function Name: init_INTC
* Description  : 割り込みレベルの設定
* Arguments   : none
* Return Value: none
*****/
void init_INTC(void)
{
    /* Configure IPR28
    b15-12 RCANA = H'F     割り込みの優先順位
    b11-8  RCANB = 0     割り込みの優先順位
    b7-4  RCANC = 0     割り込みの優先順位
    b3-0  リザーブビット*/
    INTC.IPR28.WORD = 0xE000;                  /* CAN の割り込み優先レベル設定      */

    /* Configure IPR29
    b15-12 A-DMAC = H'F   割り込みの優先順位
    b11-8  リザーブビット
    b7-4  リザーブビット
    b3-0  リザーブビット*/
    INTC.IPR29.WORD = 0xF000;                  /* A-DMAC の割り込み優先レベル設定  */
    set_imask(0);
} /* End of function init_INTC() */

/*****
* Function Name: init_RCANA
* Description  : RCAN_A の初期化
* Arguments   : none
* Return Value: none
*****/

```

```

*****/
void init_RCANA(void)
{
    /* Configure MCR
    b15    MCR15 = 0    RCAN-TL1 と HCAN2 は同等の順序)
    b14    MCR14 = 0    通常の復帰シーケンス (128×11 レセッシブビット) で RCAN-TL1 バス
オフ状態を維持
    b13-11 リザーブビット
    b10-8  TST[2:0] = 0 ノーマルモード
    b7     MCR7 = 0    CAN バスのアクティビティによる自動ウェイクモードが無効
    b6     MCR6 = 0    バスオフ時にホルトモードには入らず、復帰シーケンスが終了するのを待ちま
す
    b5     MCR5 = 0    PH2 入出力 (ポート) CAN スリープモードが解除されています
    b4,3   リザーブビット
    b2     MCR2 = 0    メッセージ ID 優先順に送信
    b1     MCR1 = 0    ホルトモードリクエストをクリア
    b0     MCR0 = 1    CAN インタフェースのリセットモード遷移リクエスト*/
    RCANA.MCR.WORD |= 0x0001;          /* リセットリクエスト (HW リセット時は自動的にセット) */
    while((RCANA.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANA.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */

    /**** インタラプトリクエストレジスタ (IRR) の設定 *****/
    /* Configure IRR
    b15    IRR15 = 0    TCMR1 のタイマコンペアマッチが発生していない
    b14    IRR14 = 0    TCMR0 のタイマコンペアマッチが発生していない
    b13    IRR13 = 0    イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバラン
が発生していない
    b12    IRR12 = 0    バスアイドル状態
    b11    IRR11 = 0    TCMR2 のタイマコンペアマッチが発生していない
    b10    IRR10 = 0    新しいシステムマトリックスの先頭でない
    b9     IRR9 = 0    メッセージオーバーラン/オーバーライト通知がない
    b8     IRR8 = 0    送信または送信キャンセルするメッセージが処理中でない
    b7     IRR7 = 0    [クリア条件] 1 を書き込む
    b6     IRR6 = 0    [クリア条件] 1 を書き込む
    b5     IRR5 = 0    [クリア条件] 1 を書き込む
    b4     IRR4 = 0    [クリア条件] 1 を書き込む
    b3     IRR3 = 0    [クリア条件] 1 を書き込む
    b2     IRR2 = 0    [クリア条件] RFPR のすべてのビットがクリア
    b1     IRR1 = 0    [クリア条件] RXPR のすべてのビットがクリア
    b0     IRR0 = 1    ソフトウェアリセットモードまたはホルトモードまたは CAN スリープモード
へ遷移*/
    RCANA.IRR.WORD = 0x0001;          /* IRR0 クリア (クリア条件:1 ライト) */

    /**** マスタコントロールレジスタ (MCR) の設定 *****/
    /* Configure MCR
    b15    MCR15 = 1    RCAN-TL1 と HCAN2 は異なる順序
    b14    MCR14 = 0    通常の復帰シーケンス (128×11 レセッシブビット) で RCAN-TL1 バス
オフ状態を維持
    b13-11 リザーブビット
    b10-8  TST[2:0] = 0 ノーマルモード
    b7     MCR7 = 0    CAN バスのアクティビティによる自動ウェイクモードが無効
    b6     MCR6 = 0    バスオフ時にホルトモードには入らず、復帰シーケンスが終了するのを待ちま
す
    b5     MCR5 = 0    PH2 入出力 (ポート) CAN スリープモードが解除されています

```

```

b4,3 リザーブビット
b2      MCR2 = 1      メールボックス番号順 (メールボックス 31→メールボックス 1) に送信
b1      MCR1 = 0      ホルトモードリクエストをクリア
b0      MCR0 = 0      リセットモードリクエストをクリア*/
RCANA.MCR.WORD |= 0x8004;          /* ID 並び替え:MCR15=1,MCR2=1 に設定*/

/**** 割り込みの設定 *****/
/* Configure IMR
b15     IMR15 = 1     IRR の対応する割り込みをマスクする
b14     IMR14 = 1     IRR の対応する割り込みをマスクする
b13     IMR13 = 1     IRR の対応する割り込みをマスクする
b12     IMR12 = 1     IRR の対応する割り込みをマスクする
b11     IMR11 = 1     IRR の対応する割り込みをマスクする
b10     IMR10 = 1     IRR の対応する割り込みをマスクする
b9      IMR9  = 1     IRR の対応する割り込みをマスクする
b8      IMR8  = 1     IRR の対応する割り込みをマスクする
b7      IMR7  = 1     IRR の対応する割り込みをマスクする
b6      IMR6  = 1     IRR の対応する割り込みをマスクする
b5      IMR5  = 0     対応する IRRをマスクしない (割り込み要因が発生すると IRQが生成される)
b4      IMR4  = 1     IRR の対応する割り込みをマスクする
b3      IMR3  = 0     対応する IRRをマスクしない (割り込み要因が発生すると IRQが生成される)
b2      IMR2  = 1     IRR の対応する割り込みをマスクする
b1      IMR1  = 1     IRR の対応する割り込みをマスクする
b0      IMR0  = 1     IRR の対応する割り込みをマスクする*/
RCANA.IMR.WORD &= 0xFFD7;          /* IRR6,5,4,3 の割り込み許可          */

/**** メールボックス (RAM エリア) 初期化 *****/
for(i = 0;i < 32;i++)              /* メールボックスを 0 にクリア          */
{
/* Configure CONTROL0
b15     IDE = H'F     スタンダードフォーマット
b14     RTR = 0       データフレーム
b13     リザーブビット
b12-2   STDID[10:0] = 0 スタンダード ID
b1,0 EXTID[17:16] = 0 エクステンデッド ID
b15-0   EXTID[15:0] = 0 エクステンデッド ID*/
    RCANA.MSG[i].CONTROL0.LONG = 0;

/* Configure LAFM
b15     IDE_LAFM = 0   対応する IDE ビットが有効
b14,13  リザーブビット
b12-2   STDID_LAFM[10:0] = 0 対応する STDID ビットが有効
b1,0 EXTID_LAFM[17:16] = 0 対応する EXTID ビットが有効
b15-0   EXTID_LAFM[15:0] = 0 対応する EXTID ビットが有効*/
    RCANA.MSG[i].LAFM.LONG = 0;
    for(j = 0;j < 2;j++)
    {
        RCANA.MSG[i].DATA.LONG[j] = 0;
    }
}

MBbuff_chA_MB1.ID.LONG=0;
MBbuff_chA_MB1.LAFM.LONG=0;
MBbuff_chA_MB1.DATA.LONG[0]=0;

```

```

MBbuff_chA_MB1.DATA.LONG[1]=0;
MBbuff_chA_MB1.CONTROL1.WORD=0;
MBbuff_chA_MB1.TimeStamp=0;

MBbuff_chA_MB2.ID.LONG=0;
MBbuff_chA_MB2.LAFM.LONG=0;
MBbuff_chA_MB2.DATA.LONG[0]=0;
MBbuff_chA_MB2.DATA.LONG[1]=0;
MBbuff_chA_MB2.CONTROL1.WORD=0;
MBbuff_chA_MB2.TimeStamp=0;

MBbuff_chA_MB3.ID.LONG=0;
MBbuff_chA_MB3.LAFM.LONG=0;
MBbuff_chA_MB3.DATA.LONG[0]=0;
MBbuff_chA_MB3.DATA.LONG[1]=0;
MBbuff_chA_MB3.CONTROL1.WORD=0;
MBbuff_chA_MB3.TimeStamp=0;

MBbuff_chA_MB4.ID.LONG=0;
MBbuff_chA_MB4.LAFM.LONG=0;
MBbuff_chA_MB4.DATA.LONG[0]=0;
MBbuff_chA_MB4.DATA.LONG[1]=0;
MBbuff_chA_MB4.CONTROL1.WORD=0;
MBbuff_chA_MB4.TimeStamp=0;

/**** 転送ビットレートの設定 *****/
/*ビットレート: 1Mbps (Pφ=40MHz)*/
/* Configure BCR1
b15-12 TSG1[3:0] = H'5PRSEG + PHSEG1=6 タイムクオンタ
b11 リザーブビット
b10-8 TSG2[2:0] = H'2PHSEG2=3 タイムクオンタ
b7,6 リザーブビット
b5,4 SJW[1:0] = 0同期ジャンプ幅=1 タイムクオンタ
b3-1 リザーブビット
b0 BSP = 0 1 か所でビットサンプリングが行われます (タイムセグメント 1 の最後)
*/
RCANA.BCR1.WORD = 0x5200; /*
TSEG1=5(6tq), TSEG2=2(3tq), SJW=0, BSP=0, (pφ=40MHz) */

/* Configure BCR0
b15-8 リザーブビット
b7-0 BRP[7:0] = 14×周辺バスクロック*/
RCANA.BCR0.WORD = 0x0001; /* BRP=1 */

/* Configure MCR
b15 MCR15 = 1 RCAN-TL1 と HCAN2 は異なる順序
b14 MCR14 = 1 MCR6 がセットされると RCAN-TL1 はバスオフ状態のあと、ただちにホ
ルトモードに入ります
b13-11 リザーブビット
b10-8 TST[2:0] = 1リスンオンリモード (受信専用モード)
b7 MCR7 = 1 CAN バスのアクティビティによる自動ウェイクモードが有効
b6 MCR6 = 1 バスオフ時に MCR1 設定によるホルトモード遷移を有効にします
b5 MCR5 = 1 CAN スリープモードへの遷移が有効です
b4,3 リザーブビット
b2 MCR2 = 1 メールボックス番号順 (メールボックス 31→メールボックス 1) に送信
b1 MCR1 = 1 ホルトモード遷移リクエスト

```

```

b0      MCR0 = 0      リセットモードリクエストをクリア*/
RCANA.MCR.WORD &= 0xFFFE;          /* MCR0 クリア */

while((RCANA.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?
*/
} /* End of function init_RCANA() */

/*****
* Function Name: init_admac_rcan
* Description   : RCAN 用 A-DMAC の設定
* Arguments    : none
* Return Value : none
*****/
void init_admac_rcan(void)
{
    /* Configure ADMAOR
    b7-1 リザーブビット
    b0      DME = 0      DMA マスタイネーブルフラグ*/
    ADMAC.ADMAOR.BIT.DME = 0;          /* DMA 転送禁止 */

    /* A-DMAC に関する設定 */
    /* Configure ADMAABR
    b7-3 リザーブビット
    b2-0 AA[2:0] = 1      エイリアス領域アドレス*/
    ADMAC.ADMAABR.BIT.AA = 1;          /* エイリアス領域の先頭アドレス(H'FFF88000) */
/*****
RCAN メールボックスの設定
*****/

    /*---- メールボックス 1 -----*/
    MBbuff_chA_MB1.ID.LONG = 0x02A80000; /* ID:H'0AA, スタートフォーマット, データフレーム*/
    MBbuff_chA_MB1.CONTROL1.WORD = 0x0008; /* メールボックス 1 を送信用, 送信データを 8byte
に設定 */
    MBbuff_chA_MB1.DATA.LONG[0] = 0x22446688; /* メールボックス 1 の送信データ */
    MBbuff_chA_MB1.DATA.LONG[1] = 0xAACCEE00; /* メールボックス 1 の送信データ */
    /*---- メールボックス 2 -----*/
    MBbuff_chA_MB2.ID.LONG = 0x06EC0000; /* ID:H'1BB, スタートフォーマット, データフレーム*/
    MBbuff_chA_MB2.CONTROL1.WORD = 0x0008; /* メールボックス 2 を送信用, 送信データを 8byte
に設定 */
    MBbuff_chA_MB2.DATA.LONG[0] = 0x11335577; /* メールボックス 2 の送信データ */
    MBbuff_chA_MB2.DATA.LONG[1] = 0x99BBDDFF; /* メールボックス 2 の送信データ */
    /*---- メールボックス 3 -----*/
    MBbuff_chA_MB3.ID.LONG = 0x0B300000; /* ID:H'2CC, スタートフォーマット, データフレーム*/
    MBbuff_chA_MB3.CONTROL1.WORD = 0x0008; /* メールボックス 3 を送信用, 送信データを 8byte
に設定 */
    MBbuff_chA_MB3.DATA.LONG[0] = 0x11223344; /* メールボックス 3 の送信データ */
    MBbuff_chA_MB3.DATA.LONG[1] = 0x55667788; /* メールボックス 3 の送信データ */
    /*---- メールボックス 4 -----*/
    MBbuff_chA_MB4.ID.LONG = 0x0F740000; /* ID:H'3DD, スタートフォーマット, データフレーム*/
    MBbuff_chA_MB4.CONTROL1.WORD = 0x0008; /* メールボックス 4 を送信用, 送信データを 8byte
に設定 */
    MBbuff_chA_MB4.DATA.LONG[0] = 0x8899AABB; /* メールボックス 4 の送信データ */
    MBbuff_chA_MB4.DATA.LONG[1] = 0xCCDDEEFF; /* メールボックス 4 の送信データ */

    /*---- A-DMAC 転送要求の設定 -----*/

```

```

/* Configure ADMATVPR1
b15-0 MB15-1 MBx に対応するフラグが 1 にセットされます*/
ADMAC.ADMATVPR1.WORD = 0x001E;          /* chA_MB1-4 転送要求          */

/*---- A-DMAC 転送転送終了割り込みの設定 -----*/
/* Configure ADMAIE9
b7      リザーブビット
b6      リザーブビット
b5      リザーブビット
b4      リザーブビット
b3      リザーブビット
b2      Ch74 = 1      割り込みの許可
b1      リザーブビット
b0      リザーブビット*/
ADMAC.ADMAIE9.BIT.Channel74 = 1;       /* 割り込み許可          */

/* Configure ADMADE
b7      リザーブビット
b6      リザーブビット
b5      リザーブビット
b4      リザーブビット
b3      リザーブビット
b2      Ch74 = 1      DMA 転送の許可
b1      リザーブビット
b0      Ch72 = 0      DMA 転送の禁止*/
ADMAC.ADMADE7.BIT.Channel74 = 1;       /* RCANチャネル(ch74)用のDMA転送許可 */

/* Configure ADMAOR
b7-1 リザーブビット
b0      DME = 1      DMA マスタイネーブルフラグ*/
ADMAC.ADMAOR.BIT.DME = 1;              /* DMA 転送許可          */
} /* End of function init_admac_rcan() */

/*****
* Function Name: int_te74
* Description   : A-DMAC 転送終了割り込み
* Arguments    : none
* Return Value : none
*****/
void int_te74(void)
{
  /* Configure ADMAIE9
b7      リザーブビット
b6      リザーブビット
b5      リザーブビット
b4      リザーブビット
b3      リザーブビット
b2      Ch74 = 0      割り込みの禁止
b1      リザーブビット
b0      リザーブビット*/
ADMAC.ADMAIE9.BIT.Channel74 = 0;       /* 転送終了割り込み禁止 */

  /* Configure ADMADE
b7      リザーブビット
b6      リザーブビット

```

```

b5      リザーブビット
b4      リザーブビット
b3      リザーブビット
b2      Ch74 = 0      DMA 転送の禁止
b1      リザーブビット
b0      Ch72 = 0      DMA 転送の禁止*/
ADMAC.ADMADE7.BIT.Channel174 = 0;          /* RCANチャネル(ch74)用のDMA転送禁止 */

/* Configure TXPR
b15-0   TXPR1[15:0] = 0対応するメールボックスが送信メッセージアイドル状態
b15-1   TXPR0[15:1] = H'1E対応するメールボックスに送信リクエストが発生
b0      リザーブビット*/
RCANA.TXPR.LONG = 0x0000001E;              /* MB1-4 を送信待ち状態に設定 */

} /* End of function int_te74() */

/*****
* Function Name: int_ersa
* Description   : RCAN_A エラー割り込み
* Arguments    : none
* Return Value : none
*****/
void int_ersa(void)
{
  if(RCANA.IRR.WORD & 0x0008)              /* IRR3 = 1? */
  {
    /* Configure IRR
    b15      IRR15 = 0   TCMR1 のタイマコンペアマッチが発生していない
    b14      IRR14 = 0   TCMR0 のタイマコンペアマッチが発生していない
    b13      IRR13 = 0   イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバ
ランが発生していない
    b12      IRR12 = 0   バスアイドル状態
    b11      IRR11 = 0   TCMR2 のタイマコンペアマッチが発生していない
    b10      IRR10 = 0   新しいシステムマトリックスの先頭でない
    b9       IRR9  = 0   メッセージオーバーラン/オーバーライト通知がない
    b8       IRR8  = 0   送信または送信キャンセルするメッセージが処理中でない
    b7       IRR7  = 0   [クリア条件] 1 を書き込む
    b6       IRR6  = 0   [クリア条件] 1 を書き込む
    b5       IRR5  = 0   [クリア条件] 1 を書き込む
    b4       IRR4  = 0   [クリア条件] 1 を書き込む
    b3       IRR3  = 1   送信エラーによるエラーワーニング状態
    b2       IRR2  = 0   [クリア条件] RFPR のすべてのビットがクリア
    b1       IRR1  = 0   [クリア条件] RXPR のすべてのビットがクリア
    b0       IRR0  = 0   [クリア条件] 1 を書き込む*/
    RCANA.IRR.WORD = 0x0008;              /* 送信エラーカウンタリング 割り込みフラグをクリア */
  }
  else if(RCANA.IRR.WORD & 0x0020)        /* IRR5 = 1? */
  {
    /* Configure IRR
    b15      IRR15 = 0   TCMR1 のタイマコンペアマッチが発生していない
    b14      IRR14 = 0   TCMR0 のタイマコンペアマッチが発生していない
    b13      IRR13 = 0   イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバ
ランが発生していない
    b12      IRR12 = 0   バスアイドル状態

```

```
b11      IRR11 = 0   TCMR2 のタイマコンペアマッチが発生していない
b10      IRR10 = 0   新しいシステムマトリックスの先頭でない
b9       IRR9  = 0   メッセージオーバーラン/オーバーライト通知がない
b8       IRR8  = 0   送信または送信キャンセルするメッセージが処理中でない
b7       IRR7  = 0   [クリア条件] 1 を書き込む
b6       IRR6  = 0   [クリア条件] 1 を書き込む
b5       IRR5  = 1   送信/受信エラーによるエラーパッシブ状態
b4       IRR4  = 0   [クリア条件] 1 を書き込む
b3       IRR3  = 0   [クリア条件] 1 を書き込む
b2       IRR2  = 0   [クリア条件] RFPR のすべてのビットがクリア
b1       IRR1  = 0   [クリア条件] RXPR のすべてのビットがクリア
b0       IRR0  = 0   [クリア条件] 1 を書き込む*/
RCANA.IRR.WORD = 0x0020;          /* エラーパッシブ 割り込みフラグをクリア */
*/

}

} /* End of function int_ersa() */
```



(b) 受信側プログラムリスト

```

/*****
* DISCLAIMER
* This software is supplied by Renesas Electronics Corporation and is only
* intended for use with Renesas products. No other uses are authorized. This
* software is owned by Renesas Electronics Corporation and is protected under
* all applicable laws, including copyright laws.
* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
* THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
* LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
* ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
* ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
* BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
* Renesas reserves the right, without notice, to make changes to this software
* and to discontinue the availability of this software. By using this software,
* you agree to the additional terms and conditions found by accessing the
* following link:
* http://www.renesas.com/disclaimer *
* Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
*****/
/*****
* File Name      : SH72546R.c
* Version        : 1.00
* Device(s)      : SH7254R
* Tool-Chain     : High-performance Embedded Workshop (Ver.4.08.00).
* OS             : None
* H/W Platform   : SH72546R
* Description     : This is the main tutorial code.
* Operation      : RCAN_RCV
* Limitations    : None
*****/
/*****
* History : DD.MM.YYYY Version Description
*         : 23.08.2011 1.00   First Release
*****/
/*****
Includes <System Includes> , "Project Includes"
*****/
#include <machine.h> /* ライブラリ関数用ヘッダファイル */
#include "iodefine.h" /* 周辺レジスタ定義ヘッダファイル */
/*****
Macro definitions
*****/
#define MBbuff_chA_MB0 (*(struct st_mbbuff*)0xFFFF88200) /* RCANA_MB0 エイリアス領域 */
/*****
Private global variables and functions
*****/
void main(void );
void set_RCANA(void); /* RCAN_A の初期化 */
void init_io_RCAN(void); /* RCAN で使用する端子の設定 */
void init_INTC(void); /* 割り込み優先レベルの設定 */
void init_admac_rcan(void); /* A-DMAC ch72 (RCAN 受信用) の設定 */
unsigned char Rcv_cnt; /* 受信用カウンタ */
unsigned char i,j;
*****/

```

```

                                退避用 RAM エリアの定義                                *
*****/
struct
{
    union
    {
        unsigned long LONG;
        struct
        {
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;

    union
    {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;

}MBbuff[4];

/*****
                                RCANA_MB 用エリア領域の定義
*****/
struct st_mbbuff
{
    union
    {
        unsigned long LONG;
        struct
        {
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;

    union
    {
        unsigned long LONG;
        struct
        {
            unsigned short H;
            unsigned short L;
        }WORD;
    }LAFM;

    union
    {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;

    union
    {
        unsigned short WORD;
    }
}

```

```

    struct
    {
        unsigned short H;
        unsigned short L;
    }BYTE;
}CONTROL1;

unsigned short TimeStamp;
};

/*****
* Function Name: main
* Description   : Main function
* Arguments    : none
* Return Value : none
*****/
void main(void)
{
    Rcv_cnt = 0; /* 受信用カウンタ初期化 */
    /*--- RCAN 送受信端子の設定 -----*/
    init_io_RCAN();
    /*--- 割り込みレベルの設定 -----*/
    init_INTC();
    /*--- RCAN の初期設定 -----*/
    set_RCANA();
    /*--- RCAN 用 A-DMAC の初期設定 -----*/
    init_admac_rcan();
    while(1);
} /* End of function main() */

/*****
RCAN 用端子の設定
*****/
/*****
* Function Name: init_io_RCAN
* Description   : RCAN 用端子の設定
* Arguments    : none
* Return Value : none
*****/
void init_io_RCAN(void)
{
    /* Configure PJCR1
    b15,14 PJ7MD[1:0] = 0 PJ7 入出力 (ポート)
    b13     リザーブビット
    b12     PJ6MD = 0     PJ6 入出力 (ポート)
    b11     リザーブビット
    b10     PJ5MD = 0     PJ5 入出力 (ポート)
    b9,8 PJ4MD[1:0] = 0 PJ4 入出力 (ポート)
    b7,6 PJ3MD[1:0] = 0 PJ3 入出力 (ポート)
    b5,4 PJ2MD[1:0] = 0 PJ2 入出力 (ポート)
    b3,2 PJ1MD[1:0] = 2 CRx_A 入力 (RCAN-TL1)
    b1,0 PJ0MD[1:0] = 2 CTx_A 出力 (RCAN-TL1) */
    PORTJ.CR1.WORD |= 0x000A; /* PJ0:CTx_A 出力, PJ1:CRx_A 入力に設定 */
} /* End of function init_io_RCAN() */

/*****
* Function Name: init_INTC

```

```

* Description   : 割り込みレベルの設定
* Arguments    : none
* Return Value  : none
*****/
void init_INTC(void)
{
    /* Configure IPR28
    b15-12 RCANA = H'F    割り込みの優先順位
    b11-8  RCANB = 0    割り込みの優先順位
    b7-4  RCANC = 0    割り込みの優先順位
    b3-0 リザーブビット*/
    INTC.IPR28.WORD = 0xF000;          /* CAN の割り込み優先レベル設定      */
    set_imask(0);
} /* End of function init_INTC() */

/*****
* Function Name: set_RCANA
* Description   : RCAN_A の初期化
* Arguments    : none
* Return Value  : none
*****/
void set_RCANA(void)
{
    /* Configure MCR
    b15    MCR15 = 0    RCAN-TL1 と HCAN2 は同等の順序)
    b14    MCR14 = 0    通常の復帰シーケンス (128×11 レセッシブビット) で RCAN-TL1 バス
オフ状態を維持
    b13-11 リザーブビット
    b10-8  TST[2:0] = 0 ノーマルモード
    b7     MCR7 = 0    CAN バスのアクティビティによる自動ウェイクモードが無効
    b6     MCR6 = 0    バスオフ時にホルトモードには入らず、復帰シーケンスが終了するのを待ちま
す
    b5     MCR5 = 0    PH2 入出力 (ポート) CAN スリープモードが解除されています
    b4,3  リザーブビット
    b2     MCR2 = 0    メッセージ ID 優先順に送信
    b1     MCR1 = 0    ホルトモードリクエストをクリア
    b0     MCR0 = 1    CAN インタフェースのリセットモード遷移リクエスト*/
    RCANA.MCR.WORD |= 0x0001;          /* リセットリクエスト(HWリセット時は自動的にセット) */
    while((RCANA.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ETリセット状態) */
    while((RCANA.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */

    /**** インタラプトリクエストレジスタ (IRR) の設定 *****/
    /* Configure IRR
    b15    IRR15 = 0    TCMR1 のタイマコンペアマッチが発生していない
    b14    IRR14 = 0    TCMR0 のタイマコンペアマッチが発生していない
    b13    IRR13 = 0    イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバラン
が発生していない
    b12    IRR12 = 0    バスアイドル状態
    b11    IRR11 = 0    TCMR2 のタイマコンペアマッチが発生していない
    b10    IRR10 = 0    新しいシステムマトリックスの先頭でない
    b9     IRR9 = 0    メッセージオーバーラン/オーバーライト通知がない
    b8     IRR8 = 0    送信または送信キャンセルするメッセージが処理中でない
    b7     IRR7 = 0    [クリア条件] 1 を書き込む
    b6     IRR6 = 0    [クリア条件] 1 を書き込む

```

```

b5      IRR5 = 0      [クリア条件] 1 を書き込む
b4      IRR4 = 0      [クリア条件] 1 を書き込む
b3      IRR3 = 0      [クリア条件] 1 を書き込む
b2      IRR2 = 0      [クリア条件] RFPR のすべてのビットがクリア
b1      IRR1 = 0      [クリア条件] RXPR のすべてのビットがクリア
b0      IRR0 = 1      ソフトウェアリセットモードまたはホルトモードまたは CAN スリープ
モードへ遷移*/
RCANA.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト)          */

/**** マスタコントロールレジスタ (MCR) の設定 *****/
/* Configure MCR
b15     MCR15 = 1     RCAN-TL1 と HCAN2 は異なる順序
b14     MCR14 = 0     通常の復帰シーケンス (128×11 レセッシブビット) で RCAN-TL1 バス
オフ状態を維持
b13-11  リザーブビット
b10-8   TST[2:0] = 0 ノーマルモード
b7      MCR7 = 0     CAN バスのアクティビティによる自動ウェイクモードが無効
b6      MCR6 = 0     バスオフ時にホルトモードには入らず、復帰シーケンスが終了するのを待ちま
す
b5      MCR5 = 0     PH2 入出力 (ポート) CAN スリープモードが解除されています
b4,3   リザーブビット
b2      MCR2 = 0     メッセージ ID 優先順に送信
b1      MCR1 = 0     ホルトモードリクエストをクリア
b0      MCR0 = 0     リセットモードリクエストをクリア*/
RCANA.MCR.WORD |= 0x8000;        /* ID 並び替え:MCR15=1 (初期値) に設定 */

/**** 割り込みの設定 *****/
/* Configure IMR
b15     IMR15 = 1     IRR の対応する割り込みをマスクする
b14     IMR14 = 1     IRR の対応する割り込みをマスクする
b13     IMR13 = 1     IRR の対応する割り込みをマスクする
b12     IMR12 = 1     IRR の対応する割り込みをマスクする
b11     IMR11 = 1     IRR の対応する割り込みをマスクする
b10     IMR10 = 1     IRR の対応する割り込みをマスクする
b9      IMR9 = 1     IRR の対応する割り込みをマスクする
b8      IMR8 = 1     IRR の対応する割り込みをマスクする
b7      IMR7 = 1     IRR の対応する割り込みをマスクする
b6      IMR6 = 1     IRR の対応する割り込みをマスクする
b5      IMR5 = 0     対応する IRRをマスクしない (割り込み要因が発生すると IRQが生成される)
b4      IMR4 = 0     対応する IRRをマスクしない (割り込み要因が発生すると IRQが生成される)
b3      IMR3 = 1     IRR の対応する割り込みをマスクする
b2      IMR2 = 1     IRR の対応する割り込みをマスクする
b1      IMR1 = 0     対応する IRRをマスクしない (割り込み要因が発生すると IRQが生成される)
b0      IMR0 = 1     IRR の対応する割り込みをマスクする*/
RCANA.IMR.WORD &= 0xFFCD;        /* IRR5,4,1 の割り込み許可          */

/* Configure MBIMR0
b15     MBIMR1_15 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b14     MBIMR1_14 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b13     MBIMR1_13 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b12     MBIMR1_12 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b11     MBIMR1_11 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b10     MBIMR1_10 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b9      MBIMR1_9 = 1 IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止

```

```

b8      MBIMR1_8 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b7      MBIMR1_7 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b6      MBIMR1_6 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b5      MBIMR1_5 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b4      MBIMR1_4 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b3      MBIMR1_3 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b2      MBIMR1_2 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b1      MBIMR1_1 = 1IRR1、IRR2、IRR8、IRR9 の割り込み要求を禁止
b0      MBIMR1_0 = 0IRR1、IRR2、IRR8、IRR9 の割り込み要求を許可*/
RCANA.MBIMR.WORD.MBIMR0 &= 0xFFFE;          /* MB0 割り込み許可          */

/**** メールボックス初期化 *****/
for(i = 0;i < 4;i++)                          /* 受信用 RAM アreaを 0 にクリア */
{
    MBbuff[i].ID.LONG = 0;
    MBbuff[i].DATA.LONG[0] = 0;
    MBbuff[i].DATA.LONG[1] = 0;
}

for(i = 0;i < 32;i++)                          /* メールボックスを 0 にクリア */
{
    /* Configure CONTROL0
b15     IDE = H'F   スタANDARDフォーマット
b14     RTR = 0     データフレーム
b13     リザーブビット
b12-2   STDID[10:0] = 0 スタANDARD ID
b1,0 EXTID[17:16] = 0 エクステンデッド ID
b15-0   EXTID[15:0] = 0 エクステンデッド ID*/
    RCANA.MSG[i].CONTROL0.LONG = 0;

    /* Configure LAFM
b15     IDE_LAFM = 0   対応する IDE ビットが有効
b14,13  リザーブビット
b12-2   STDID_LAFM[10:0] = 0 対応する STDID ビットが有効
b1,0 EXTID_LAFM[17:16] = 0 対応する EXTID ビットが有効
b15-0   EXTID_LAFM[15:0] = 0 対応する EXTID ビットが有効*/
    RCANA.MSG[i].LAFM.LONG = 0;
    for(j = 0;j < 2;j++)
    {
        RCANA.MSG[i].DATA.LONG[j] = 0;
    }
}

/**** 受信用メールボックスの設定 *****/
/* Configure CONTROL0
b15     IDE = H'F   スタANDARDフォーマット
b14     RTR = 0     データフレーム
b13     リザーブビット
b12-2   STDID[10:0] = 0 スタANDARD ID
b1,0 EXTID[17:16] = 0 エクステンデッド ID
b15-0   EXTID[15:0] = 0 エクステンデッド ID*/
RCANA.MSG[0].CONTROL0.LONG = 0x00000000; /* ID(XXX),スタANDARDフォーマット,データフレーム */

/* Configure LAFM
b15     IDE_LAFM = 1   対応する IDE ビットが有効

```

```

b14,13 リザーブビット
b12-2  STDID_LAFM[10:0] = H'3FFF 対応する STDID ビットが有効
b1,0 EXTID_LAFM[17:16] = H'3 対応する EXTID ビットが有効
b15-0  EXTID_LAFM[15:0] = H'FFFF 対応する EXTID ビットが有効*/
RCANA.MSG[0].LAFM.LONG = 0x9fffffff;      /* STD_LAFM の設定          */

/* Configure CONTROL1
b15,14 リザーブビット
b13    NMC = 0          オーバランモード
b12    ATX           データフレームの自動送信無効
b11    DART = 0       スタンダード ID
b10-8  MBC[2:0] = H'2 メールボックスコンフィギュレーション
b7-4 リザーブビット
b3-0 DLC[3:0] = 0 データ長コード*/
RCANA.MSG[0].CONTROL1.WORD = 0x0200;      /* メールボックス 0 を受信用に設定  */

/**** 転送ビットレートの設定 *****/
/* Configure BCR1
b15-12 TSG1[3:0] = H'5PRSEG + PHSEG1=6 タイムクオンタ
b11    リザーブビット
b10-8  TSG2[2:0] = H'2PHSEG2=3 タイムクオンタ
b7,6 リザーブビット
b5,4 SJW[1:0] = 0同期ジャンプ幅=1 タイムクオンタ
b3-1 リザーブビット
b0     BSP = 0          1 か所でビットサンプリングが行われます (タイムセグメント 1 の最後)
*/
RCANA.BCR1.WORD = 0x5200;                  /*
TSEG1=5(6tq), TSEG2=2(3tq), SJW=0, BSP=0, (pφ=40MHz) */

/* Configure BCR0
b15-8  リザーブビット
b7-0 BRP[7:0] = 14×周辺バスクロック*/
RCANA.BCR0.WORD = 0x0001;                  /* BSP=0, BRP=1 (Pφ=40MHz 時 1Mbps) */

/* Configure MCR
b15    MCR15 = 1      RCAN-TL1 と HCAN2 は異なる順序
b14    MCR14 = 1      MCR6 がセットされると RCAN-TL1 はバスオフ状態のあと、ただちにホ
ルトモードに入ります
b13-11 リザーブビット
b10-8  TST[2:0] = 1 リスンオンリモード (受信専用モード)
b7     MCR7 = 1      CAN バスのアクティビティによる自動ウェイクモードが有効
b6     MCR6 = 1      バスオフ時に MCR1 設定によるホルトモード遷移を有効にします
b5     MCR5 = 1      CAN スリープモードへの遷移が有効です
b4,3 リザーブビット
b2     MCR2 = 1      メールボックス番号順 (メールボックス 31→メールボックス 1) に送信
b1     MCR1 = 1      ホルトモード遷移リクエスト
b0     MCR0 = 0      リセットモードリクエストをクリア*/
RCANA.MCR.WORD &= 0xFFFE;                  /* MCR0 ｸﾘｱ          */

while((RCANA.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?          */
} /* End of function set_RCANA() */

/*****
*
* Function Name: init_admac_rcan

```

```

* Description   : RCAN 用 A-DMAC の設定
* Arguments    : none
* Return Value : none
*****
/
void init_admac_rcan(void)
{
  /* Configure ADMAOR
  b7-1 リザーブビット
  b0      DME = 0      DMA マスタイネーブルフラグ*/
  ADMAC.ADMAOR.BIT.DME = 0;          /* DMA 転送禁止          */

  /* A-DMAC に関する設定 */
  /* Configure ADMAABR
  b7-3 リザーブビット
  b2-0 AA[2:0] = 1     エイリアス領域アドレス*/
  ADMAC.ADMAABR.BIT.AA = 1;         /* エイリアス領域の先頭アドレス(H'FFF88000) */

  /* Configure ADMAD7
  b7      リザーブビット
  b6      リザーブビット
  b5      リザーブビット
  b4      リザーブビット
  b3      リザーブビット
  b2      Ch74 = 0     DMA 転送の禁止
  b1      リザーブビット
  b0      Ch72 = 1     DMA 転送の許可*/
  ADMAC.ADMAD7.BIT.Channel72 = 1;   /* RCAN チャネル(ch72)用の DMA 転送許可 */

  /* Configure ADMAOR
  b7-1 リザーブビット
  b0      DME = 1      DMA マスタイネーブルフラグ*/
  ADMAC.ADMAOR.BIT.DME = 1;         /* DMA 転送許可          */
} /* End of function init_admac_rcan() */

*****
*
* Function Name: int_rma0
* Description   : RCAN_A データフレーム受信割り込み
* Arguments    : none
* Return Value : none
*****
/
void int_rma0(void)
{
  while(ADMAC.ADMARVPR1.WORD != 0x0001); /* IRR1 セット, MB0 受信          */
  MBbuff[Rcv_cnt].ID.LONG=MBbuff_chA_MB0.ID.LONG; /* 受信 ID を RAM に格納          */

  for(i = 0; i < 2; i++)
  {
    /* 受信データを RAM に格納          */
    MBbuff[Rcv_cnt].DATA.LONG[i] = MBbuff_chA_MB0.DATA.LONG[i];
  }
  /* Configure ADMARVPR1
  b15-0 MB15-0 MBx に対応するフラグが 1 にセットされます*/
  ADMAC.ADMARVPR1.WORD = 0x0001; /* MB0 に対応する RV ビットのクリア */
}

```



```

    if(Rcv_cnt == 3)
    {
        Rcv_cnt = 0;
    }
else
    {
        Rcv_cnt++;
    }
} /* End of function int_rma0() */

/*****
* Function Name: int_ersa
* Description   : RCAN_A エラー割り込み
* Arguments    : none
* Return Value : none
*****/
void int_ersa(void)
{
    if(RCANA.IRR.WORD & 0x0010)                /* IRR4 = 1?                */
    {
        /* Configure IRR
b15    IRR15 = 0    TCMR1 のタイマコンペアマッチが発生していない
b14    IRR14 = 0    TCMR0 のタイマコンペアマッチが発生していない
b13    IRR13 = 0    イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバラン
が発生していない
b12    IRR12 = 0    バスアイドル状態
b11    IRR11 = 0    TCMR2 のタイマコンペアマッチが発生していない
b10    IRR10 = 0    新しいシステムマトリックスの先頭でない
b9     IRR9  = 0    メッセージオーバーラン/オーバーライト通知がない
b8     IRR8  = 0    送信または送信キャンセルするメッセージが処理中でない
b7     IRR7  = 0    [クリア条件] 1 を書き込む
b6     IRR6  = 0    [クリア条件] 1 を書き込む
b5     IRR5  = 0    [クリア条件] 1 を書き込む
b4     IRR4  = 1    受信エラーによるエラーワーニング状態
b3     IRR3  = 0    [クリア条件] 1 を書き込む
b2     IRR2  = 0    [クリア条件] RFPR のすべてのビットがクリア
b1     IRR1  = 0    [クリア条件] RXPR のすべてのビットがクリア
b0     IRR0  = 0    [クリア条件] 1 を書き込む*/
        RCANA.IRR.WORD = 0x0010;                /* 受信エラーカウンタリング割り込みフラグをクリア */
    }
else if(RCANA.IRR.WORD & 0x0020)                /* IRR5 = 1?                */
    {
        /* Configure IRR
b15    IRR15 = 0    TCMR1 のタイマコンペアマッチが発生していない
b14    IRR14 = 0    TCMR0 のタイマコンペアマッチが発生していない
b13    IRR13 = 0    イベントトリガモード (テストモードを含む) でタイマ (TCNTR) オーバラン
が発生していない
b12    IRR12 = 0    バスアイドル状態
b11    IRR11 = 0    TCMR2 のタイマコンペアマッチが発生していない
b10    IRR10 = 0    新しいシステムマトリックスの先頭でない
b9     IRR9  = 0    メッセージオーバーラン/オーバーライト通知がない
b8     IRR8  = 0    送信または送信キャンセルするメッセージが処理中でない
b7     IRR7  = 0    [クリア条件] 1 を書き込む
b6     IRR6  = 0    [クリア条件] 1 を書き込む
b5     IRR5  = 1    送信/受信エラーによるエラーパッシブ状態
b4     IRR4  = 0    [クリア条件] 1 を書き込む

```

```
b3      IRR3 = 0      [クリア条件] 1 を書き込む
b2      IRR2 = 0      [クリア条件] RFPR のすべてのビットがクリア
b1      IRR1 = 0      [クリア条件] RXPR のすべてのビットがクリア
b0      IRR0 = 0      [クリア条件] 1 を書き込む*/
        RCANA.IRR.WORD = 0x0020;          /* エラーハップ割り込みフラグをクリア */
    }
} /* End of function int_ersa() */
```

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/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

## 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いません。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情報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・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/>