RX62N グループ、RX621 グループ
RIIC マルチマスタ通信

R01AN0630JJ0100
Rev.1.00
2011.07.15

要旨

本アプリケーションノートではルネサス MCU の RIIC（I^2C バスインタフェース）を用いたマルチマスタでの通信例について説明します。

対象デバイス

RX62N グループ、RX621 グループ

RX62N グループ、RX621 グループと同様の I/O レジスタ（周辺装置制御レジスタ）を持つ他の RX ファミリでも本プログラムを使用することができます。ただし、一部の機能を機能追加等で変更している場合がありますのでマニュアルで確認してください。このアプリケーションノートのご使用に際しては十分な評価を行ってください。

目次

1. 仕様

本アプリケーションノートでは、マスタ送信（10byte）を行い、その後マスタ受信（10Byte）を行います。通信ビットレートは 100kbps とします。

表 1 に使用する周辺機能と用途を、図 1 に接続図を示します。

表 1　使用する周辺機能と用途

周辺機能	用途
RIIC	RIIC 通信用
割り込みコントローラ（ICU）	割り込み用

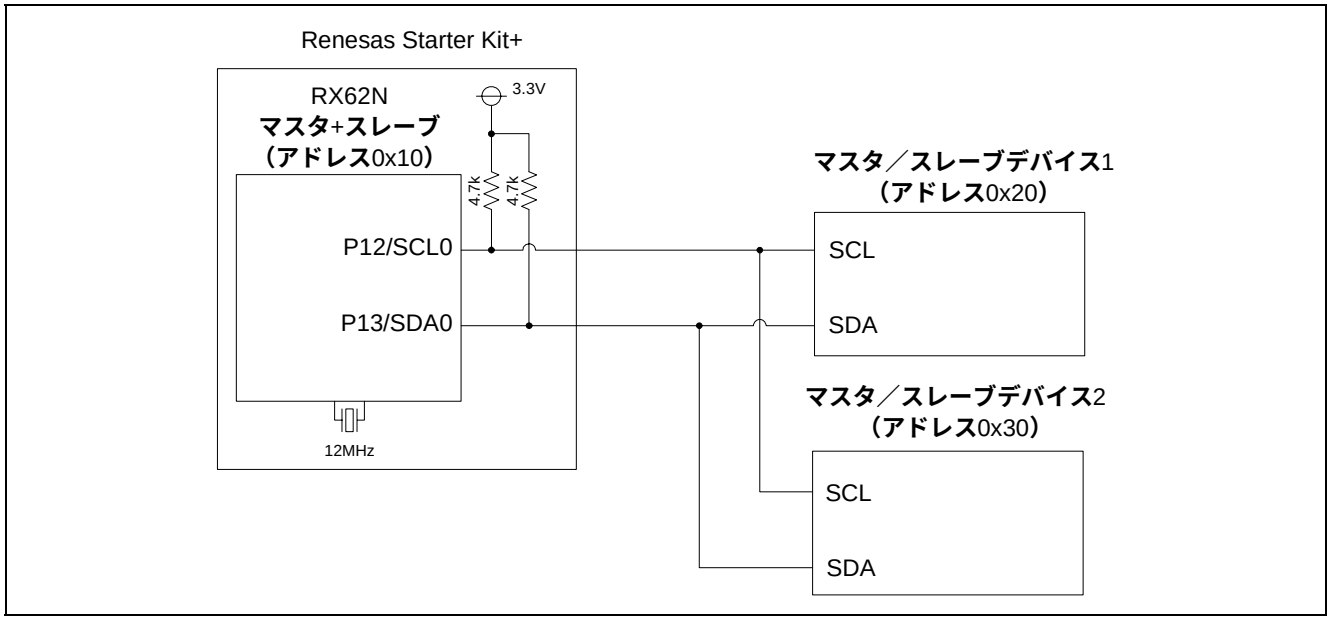

図 1　接続図

2. 動作確認環境

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　動作確認条件

項目	内容
使用マイコン	RX62N（R5F562NBDBG）
動作周波数	12MHz（ICLK=96MHz、PCLK=48MHz）
動作電圧	5.0V（CPU 動作電圧は 3.3V）
統合開発環境	Version 4.08.00.011
C コンパイラ	RX Standard Toolchain（V.1.0.1.0）
動作モード	シングルチップモード
サンプルコードのバージョン	1.00
使用ボード	Renesas Development Tools（製品型名：R0K5562N0S000BE）に同梱されている RSK+ RX62N （製品型名：R0K5562N0C00BE）を使用

3. ソフトウェア説明

3.1 動作概要

- マスタ送信を行います。マスタ送信中にアービトレーションロストを検出した場合は、他のマスタデバイスの通信を優先し、スレーブ動作を行います。
- マスタ受信を行います。マスタ受信中にアービトレーションロストを検出した場合は、他のマスタデバイスの通信を優先し、スレーブ動作を行います。
- スレーブ動作中にスレーブアドレスが一致した場合は R/W#ビットに従ってスレーブ受信、またはスレーブ送信を開始します。
- スレーブ通信完了およびマスタ通信完了時にコールバック関数を呼び出します。

本アプリケーションノートの RIIC の設定条件を表 3 に示します。

表 3　RIIC 設定

項目	内容
チャネル	**チャネル** 0
マスタ／スレーブ動作	**マスタ送信、マスタ受信、スレーブ受信、スレーブ送信**
アドレスフォーマット	7 **ビットアドレスフォーマット**
スレーブアドレス	0x10
転送速度	100kbps
アービトレーションロスト検出	• **マスタアービトレーションロスト検出**
タイムアウト検出	• SCL **ラインが** Low、High **のときカウント** • **ロングモード（**16 **ビットカウンタ）**

【注】 I^2C **バスの通信フォーマットについては、「**RX62N **グループ、**RX621 **グループ ユーザーズマニュアル ハードウェア編」、および** I^2C **バスの規格書を参照してください。**

3.2　ファイル構成

表 4 にサンプルコードで使用するファイルを示します。なお、統合開発環境で自動生成されるファイルは除きます。

表 4　ファイル構成

ファイル名	概要	備考
main.c	メイン処理	
riic.c	RIIC 制御関連の処理	
riic_int.c	RIIC 割り込み処理	
riic.h	RIIC 関連ヘッダファイル	
intprg.c	HEW が自動生成するファイル （本プログラムで使用する RIIC の割り込み関数のみを削除しています）	

3.3　定数一覧

表 5 にサンプルコードで使用する定数を示します。

表 5　定数

定数名	設定値	内容
RIIC_OK	0	関数 RiicStart のリターン値として使用します。
RIIC_NG	1	関数 RiicStart のリターン値として使用します。
RIIC_BUS_BUSY	2	関数 RiicStart のリターン値として使用します。
RIIC_ST_MST_IDLE	0	関数 RiicGetMstState のリターン値として使用します。
RIIC_ST_MST_BUSY	1	関数 RiicGetMstState のリターン値として使用します。
RIIC_ST_MST_NACK	2	関数 RiicGetMstState のリターン値として使用します。
RIIC_ST_MST_AL	3	関数 RiicGetMstState のリターン値として使用します。
RIIC_ST_MST_TMO	4	関数 RiicGetMstState のリターン値として使用します。
RIIC_ST_MST_COMPLETE	5	関数 RiicGetMstState のリターン値として使用します。
RIIC_SET	1	フラグの設定値として使用します。
RIIC_CLEAR	0	フラグの設定値として使用します。
SELF_ADDRESS	0x10	自アドレス
MST_DATA_NUM	(10+1)	マスタ送受信数として使用します。
SLV_DATA_NUM	10	スレーブ送受信数として使用します。
MST_WRITE	0x00	RiicMstStart の引数として使用します。
MST_READ	0x01	RiicMstStart の引数として使用します。

3.4　変数一覧

表 7 にグローバル変数を示します。

表 7　グローバル変数

型	変数名	内容	使用関数
uint8_t	MstTrmBuff[256]	マスタ送信用バッファ	main
uint8_t	MstRcvBuff[256]	マスタ受信用バッファ	main
uint8_t	SlvTrmBuff[256]	スレーブ送信用バッファ	main
uint8_t	SlvRcvBuff[256]	スレーブ受信用バッファ	main

3.5　関数一覧

表 8 に関数を示します。

表 8　関数

関数名	概要
main	メイン処理
CbSlaveTrm	コールバック関数（スレーブ送信完了）
CbSlaveRcv	コールバック関数（スレーブ受信完了）
CbMaster	コールバック関数（マスタ送信／受信完了）
RiicIni	ユーザ I/F 関数。RIIC 初期設定。通信許可／禁止設定
RiicMstStart	ユーザ I/F 関数。マスタ通信開始（マスタ送信／マスタ受信）
RiicGetMstState	ユーザ I/F 関数。マスタ通信結果取得
RiicSlvStart	ユーザ I/F 関数。スレーブ動作開始（スレーブ送信／スレーブ受信）
RiicTDRE	送信データエンプティ割り込み処理
RiicTEND	送信終了割り込み処理
RiicRDRF	受信データフル割り込み処理
RiicSTOP	ストップコンディション検出割り込み処理
RiicNACK	NACK 検出割り込み処理
RiicAL	アービトレーションロスト検出割り込み処理
RiicTMO	タイムアウト検出割り込み処理

3.6　関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	・RIIC の初期設定 ・スレーブ動作開始 ・マスタ送信開始 ・マスタ受信開始
引数	なし
リターン値	なし
備考	

CbSlaveTrm	
概要	コールバック関数（スレーブ送信完了）
ヘッダ	なし
宣言	void CbSlaveTrm(void)
説明	・スレーブ送信完了後、呼び出されます。
引数	なし
リターン値	なし
備考	

CbSlaveRcv	
概要	コールバック関数（スレーブ受信完了）
ヘッダ	なし
宣言	void CbSlaveRcv(void)
説明	・スレーブ受信完了後、呼び出されます。
引数	なし
リターン値	なし
備考	

CbMaster	
概要	コールバック関数（マスタ送信／受信完了）
ヘッダ	なし
宣言	void CbMaster(void)
説明	・　マスタ送信、またはマスタ受信後、呼び出されます。
引数	なし
リターン値	なし
備考	

RiicIni	
概要	RIIC 初期設定
ヘッダ	riic.h
宣言	void RiicIni(uint8_t, uint8_t)
説明	・RIIC の初期設定をします。
引数	第 1 引数　uint8_t in_SelfAddr 自アドレス（最下位ビットには 0 を設定してください） 第 2 引数　uint8_t in_Enable 0　　：RIIC 通信禁止 0 以外　：RIIC 通信許可
リターン値	なし
備考	

RiicSlvStart	
概要	スレーブ動作開始
ヘッダ	riic.h
宣言	void RiicSlvStart(uint8_t * , uint8_t * , uint32_t , CallBackFunc, CallBackFunc)
説明	スレーブ動作を開始します。スレーブ動作中にスレーブアドレスが一致した場合は R/W# ビットに従ってスレーブ送信、またはスレーブ受信を開始します。 【スレーブ送信】 ・ スレーブ送信開始後は NACK を受信するまで第 2 引数で指定した送信バッファの先頭からデータを送信します。 ・ 送信数が第 3 引数で指定した送信データ数を超える場合は 0xFF を送信します。 ・ 送信完了後、第 4 引数で指定したコールバック関数を呼び出します。ただし、第 3 引数で指定した送信データ数未満のデータ送信中に NACK を受信した場合はコールバック関数を呼び出さず、再びスレーブ送信が開始されたときに第 2 引数で指定した送信バッファの先頭から送信します。 【スレーブ受信】 ・ スレーブ受信開始後はストップコンディションを検出まで第 1 引数で指定した受信バッファの先頭からデータを受信します。 ・ 第 3 引数で指定した受信データ数を超えた受信データは第 1 引数で指定した受信バッファには格納されません。 ・ 受信完了後、第 5 引数で指定したコールバック関数を呼び出します。ただし、第 3 引数で指定した受信データ数未満のデータ送信中にストップコンディションを検出した場合はコールバック関数を呼び出さず、再びスレーブ受信が開始されたときに第 3 引数で指定した受信バッファの先頭から受信（上書き）します。
引数	第 1 引数　uint8_t * in_RcvAddr スレーブ受信データ格納ポインタ。 この引数で示されたバッファへデータを格納していきます。 第 2 引数　uint8_t * in_TrmAddr スレーブ送信データ格納ポインタ。 この引数で示されたバッファからデータを送信していきます。 第 3 引数　uint32_t in_num 送信／受信データ数。 送信／受信するデータ数を指定します。 第 4 引数　CallBackFunc cbTrm スレーブ送信完了コールバック関数。 スレーブ送信が完了するとこの関数をコールします。 第 5 引数　CallBackFunc cbRcv スレーブ受信完了コールバック関数。 スレーブ受信が完了するとこの関数をコールします。
リターン値	なし
備考	・スレーブ送信／スレーブ受信が完了するとスレーブ動作を停止します。スレーブ動作を継続する場合は本関数を再度コールしてください。

RiicStart		
概要	マスタ通信開始	
ヘッダ	riic.h	
宣言	uint8_t RiicMstStart(uint8_t, uint8_t *, uint32_t, CallBackFunc)	
説明	マスタ通信を開始します。 マスタ通信開始後、以下の条件でコールバック関数を呼び出します。 ・NACK 受信 ・アービトレーションロスト検出 ・タイムアウト検出 ・マスタ送信／マスタ受信完了 上記通信結果は関数 RiicGetMstState で取得することができます。	
引数	第 1 引数　uint8_t in_addr スレーブアドレス（最下位ビットは R/W ビット）。 最下位ビットが 0 の場合はマスタ送信を行い、1 の場合はマスタ受信を行います。 第 2 引数　uint8_t *in_buff 通信するデータ格納ポインタ。 マスタ送信する場合は、この引数で示されたバッファからデータを送信していきます。 マスタ受信する場合は、この引数で示されたバッファへデータを格納していきます。 第 3 引数　uint32_t in_num 送信／受信データ数。 送信／受信するデータ数を指定します。送信するアドレスも含まれます。 第 4 引数　CallBackFunc cb コールバック関数。	
リターン値	RIIC_OK	正常終了
	RIIC_NG	引数エラー（送信／受信データ数が 2 未満の場合）
	RIIC_BUS_BUSY	バスビジー
備考		

RiicGetMstState		
概要	マスタ通信結果取得	
ヘッダ	riic.h	
宣言	uint8_t RiicGetMstState(void)	
説明	・マスタ通信の結果を返します。	
引数	なし	
リターン値	RIIC_ST_MST_IDLE	通信開始前
	RIIC_ST_MST_BUSY	通信中
	RIIC_ST_MST_NACK	NACK 受信
	RIIC_ST_MST_AL	アービトレーションロスト検出
	RIIC_ST_MST_TMO	タイムアウト検出
	RIIC_ST_MST_COMPLETE	マスタ送信／マスタ受信完了
備考		

3.7　フローチャート

3.7.1　メイン処理

図 2 にメイン関数のフローチャートを示します。

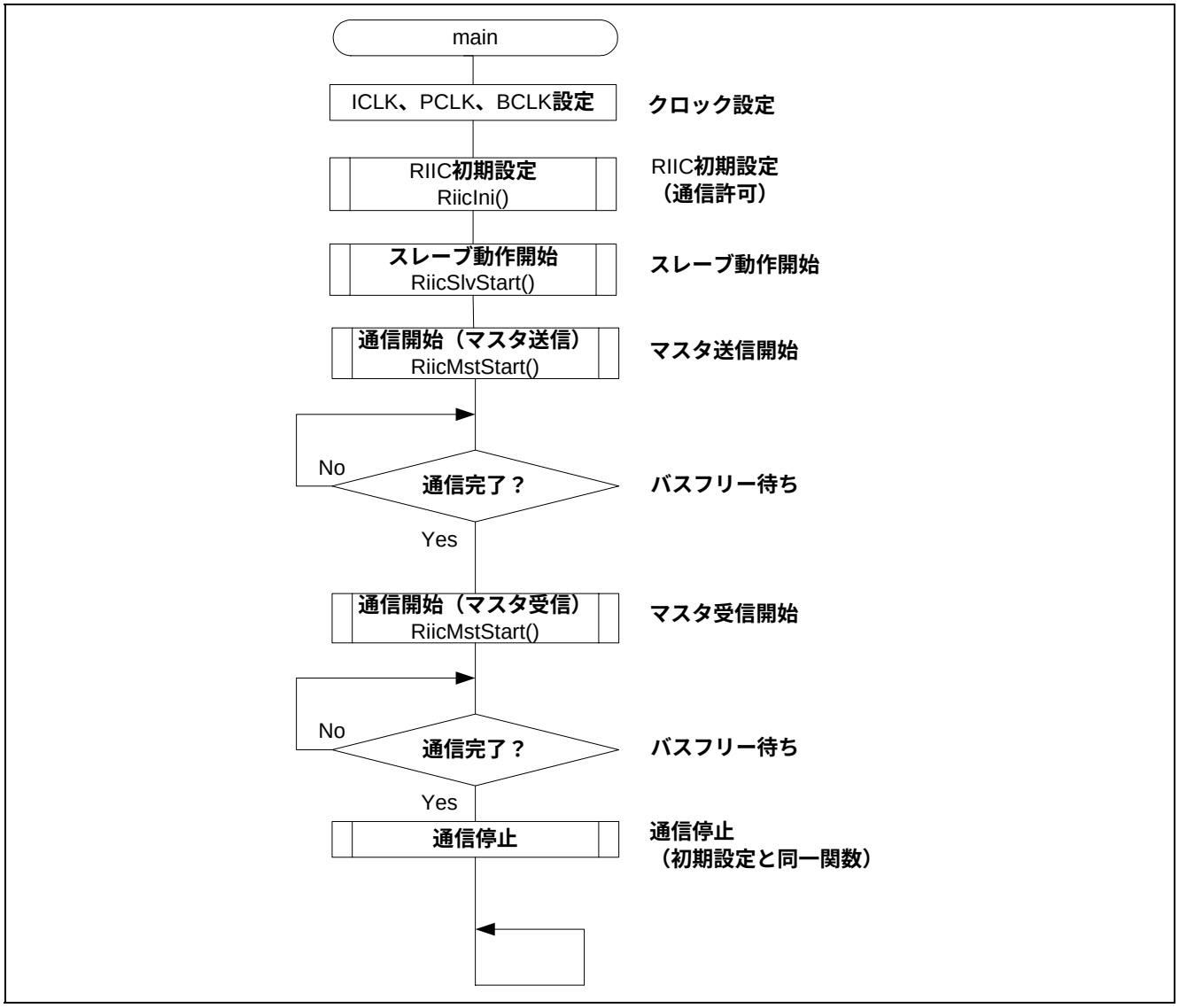

図 2　メイン処理

3.7.2　コールバック関数（スレーブ送信完了）

図 3 にコールバック関数（スレーブ送信完了）のフローチャートを示します。

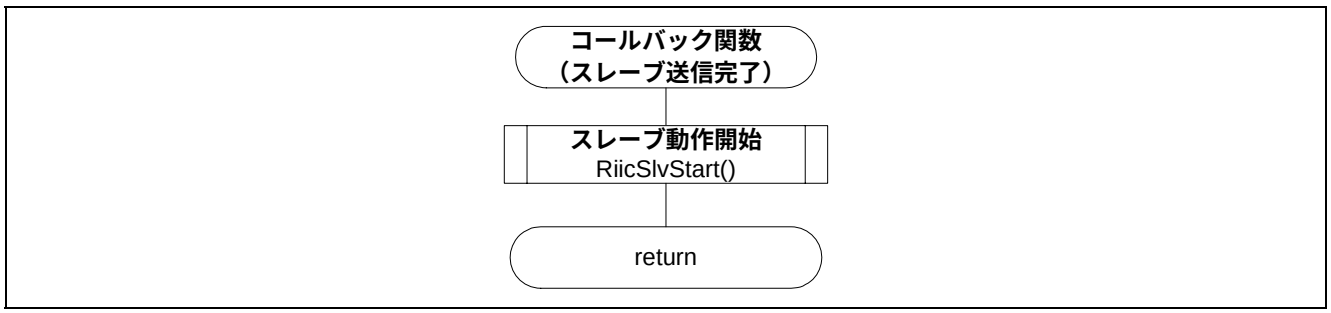

図 3　コールバック関数（スレーブ送信完了）

3.7.3　コールバック関数（スレーブ受信完了）

図 4 にコールバック関数（スレーブ受信完了）のフローチャートを示します。

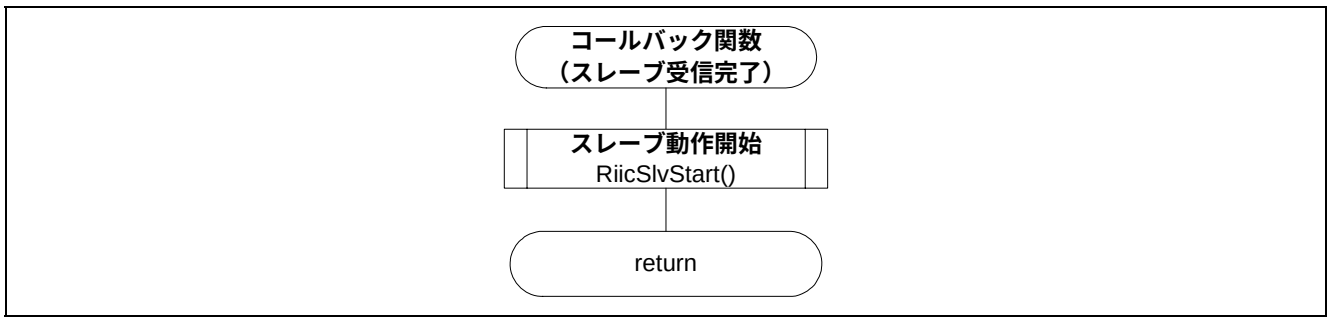

図 4　コールバック関数（スレーブ受信完了）

3.7.4　コールバック関数（マスタ送信／受信完了）

図 5 にコールバック関数（マスタ送信／受信完了）のフローチャートを示します。

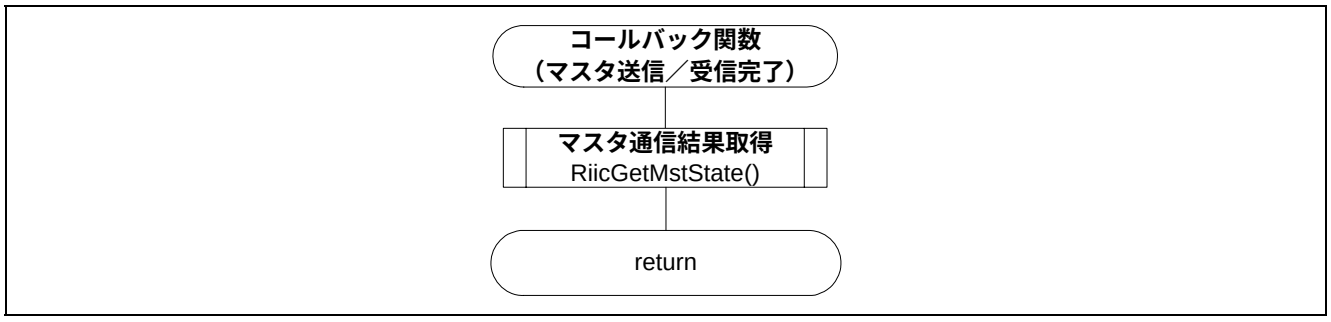

図 5　コールバック関数（マスタ送信／受信完了）

3.7.5　RIIC 初期設定

図 6 に RIIC 初期設定のフローチャートを示します。

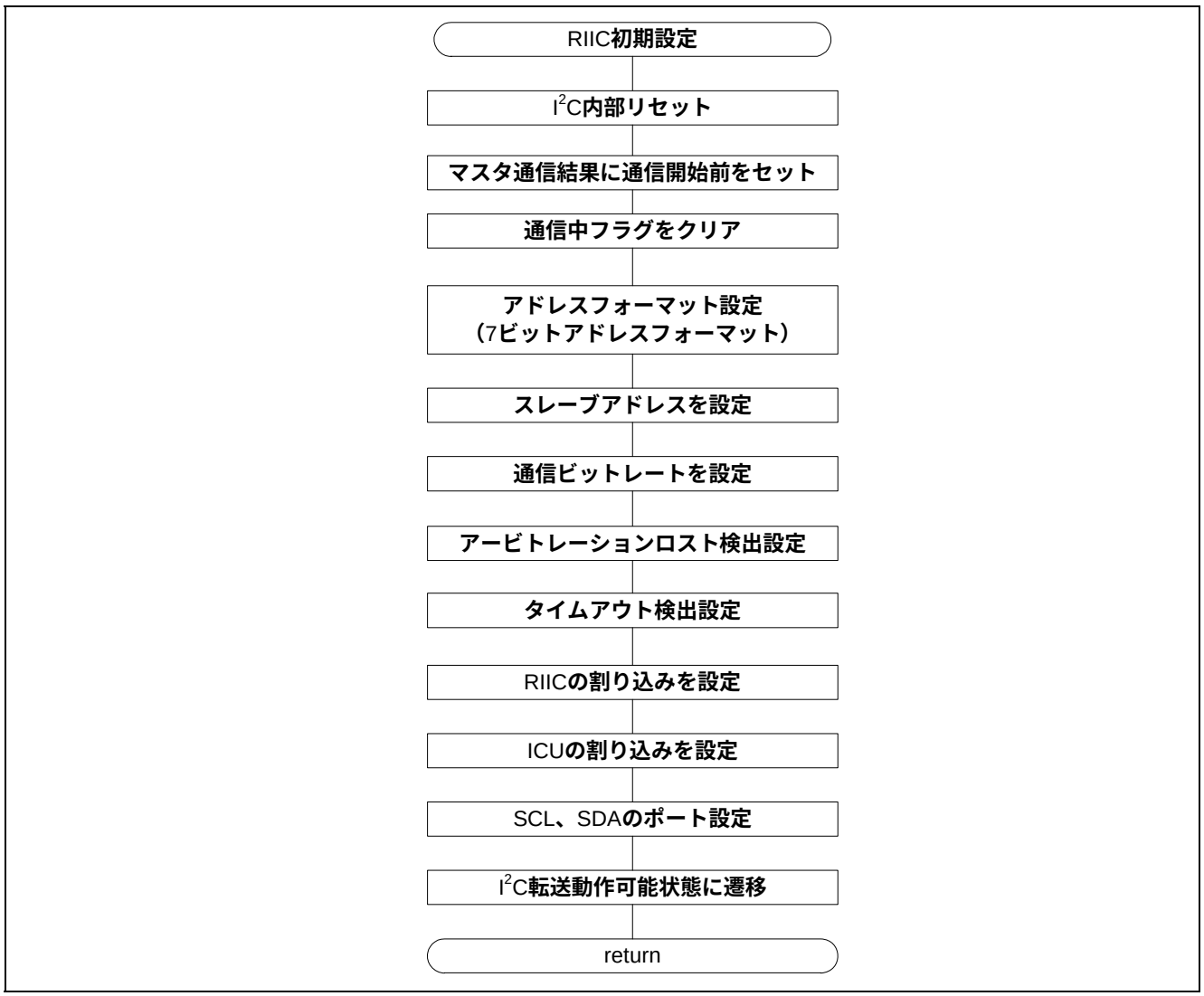

図 6　RIIC 初期設定

3.7.6　スレーブ動作開始

図 7 にスレーブ動作開始のフローチャートを示します。

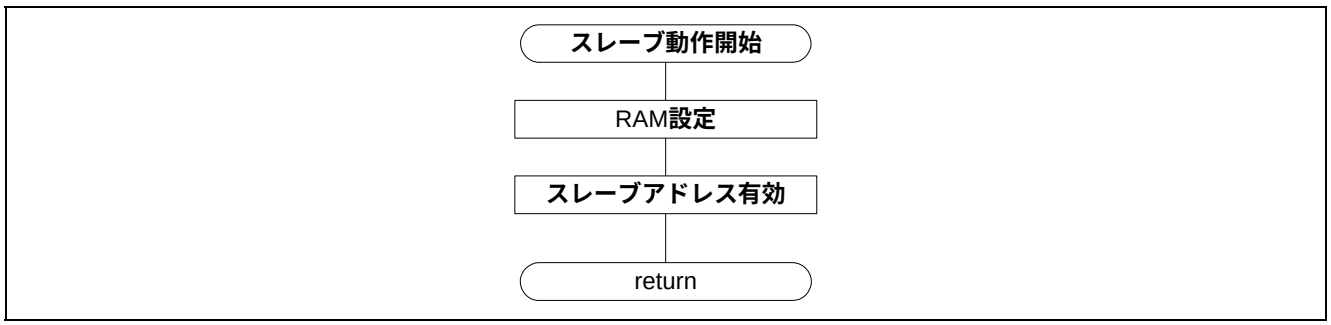

図 7　スレーブ動作開始

3.7.7　マスタ通信開始

図 8 にマスタ通信開始のフローチャートを示します。

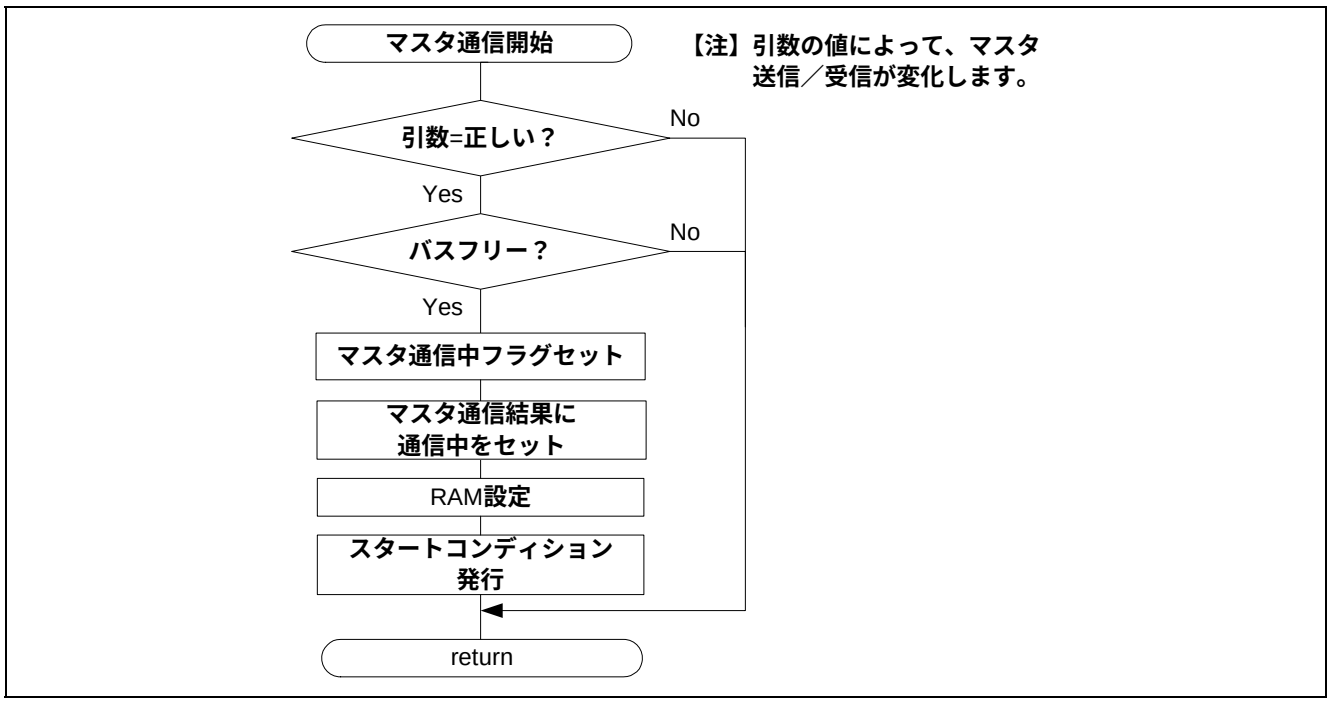

図 8　マスタ通信開始

3.7.8　マスタ通信結果取得

図 9 にマスタ通信結果取得のフローチャートを示します。

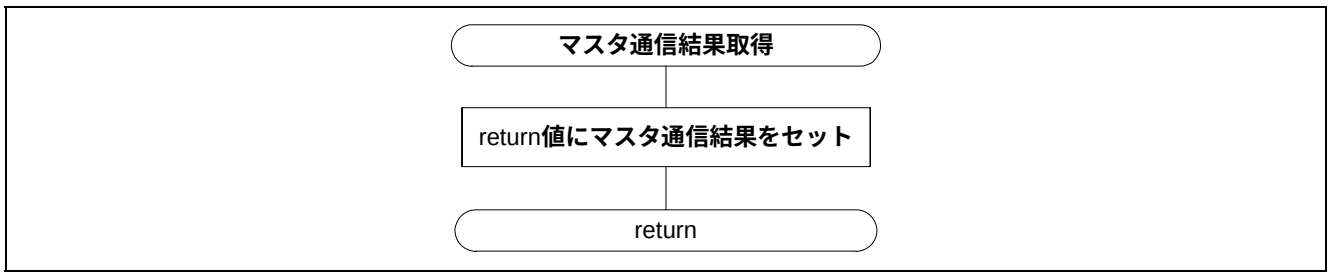

図 9　マスタ通信結果取得

3.7.9　RDRF 割り込み処理

図 10 に RDRF 割り込み処理のフローチャートを示します。

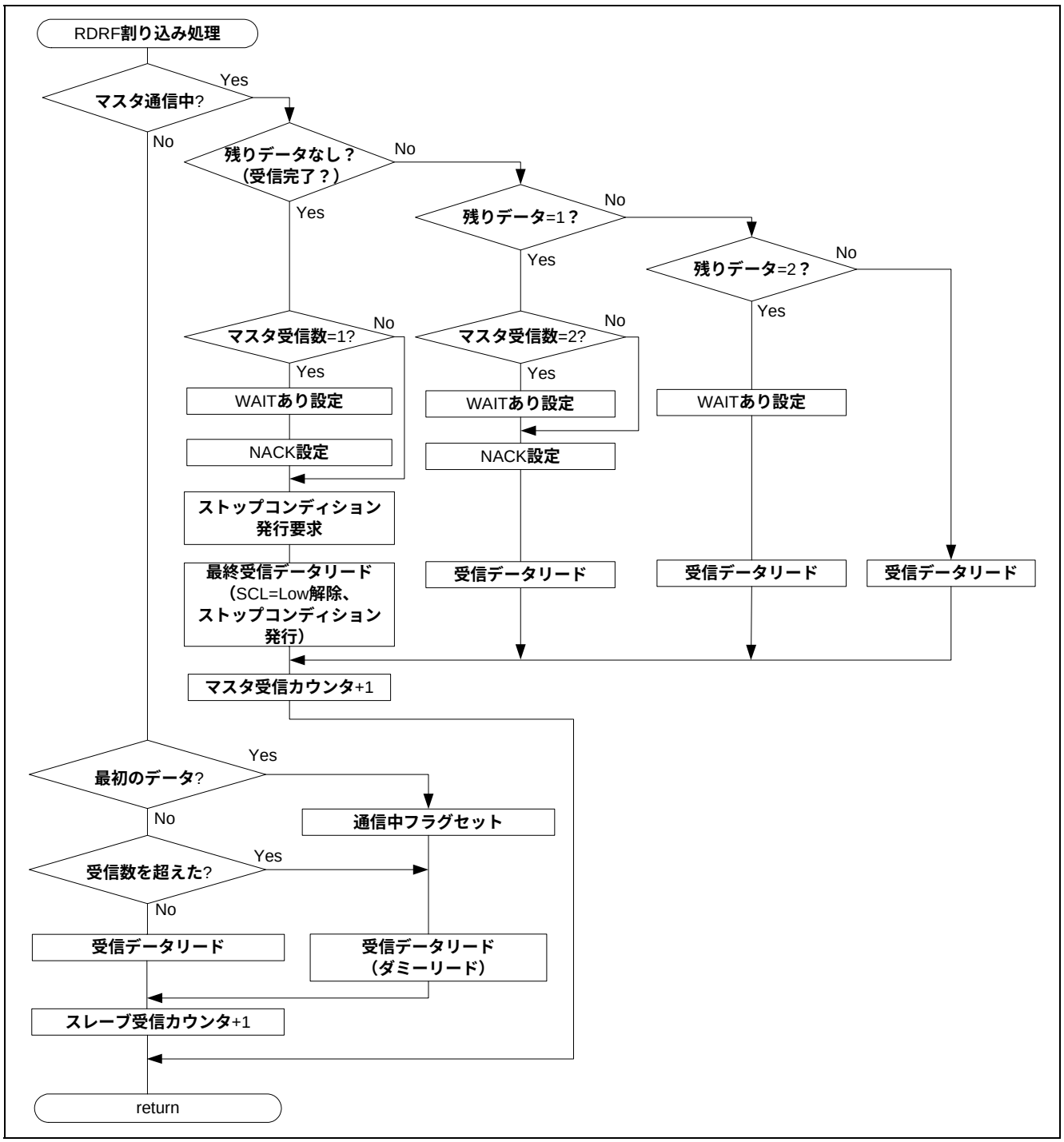

図 10　RDRF 割り込み処理

3.7.10　TDRE 割り込み処理

図 11 に TDRE 割り込み処理のフローチャートを示します。

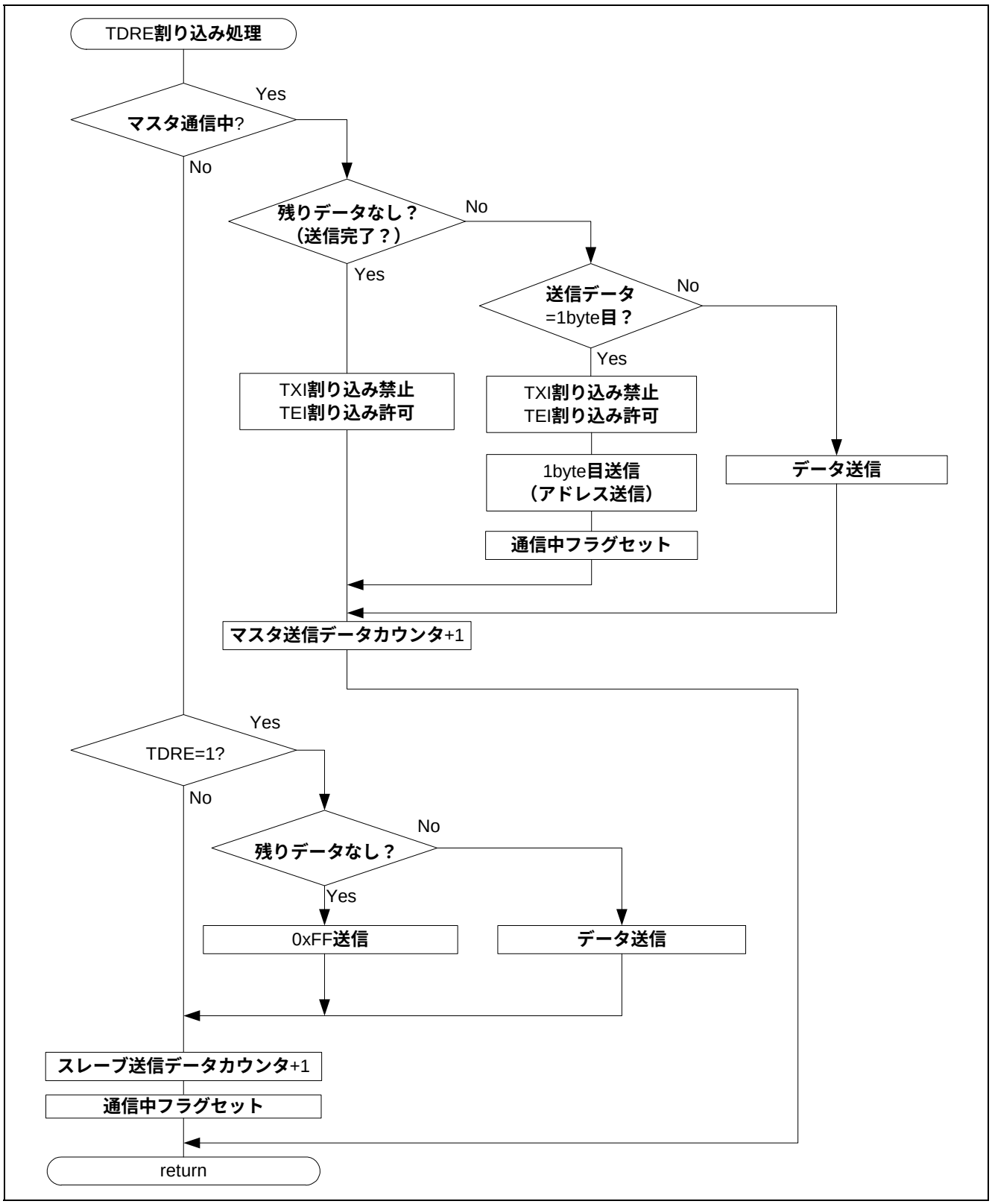

図 11　TDRE 割り込み処理

3.7.11　TEND 割り込み処理

図 12 に TEND 割り込み処理のフローチャートを示します。

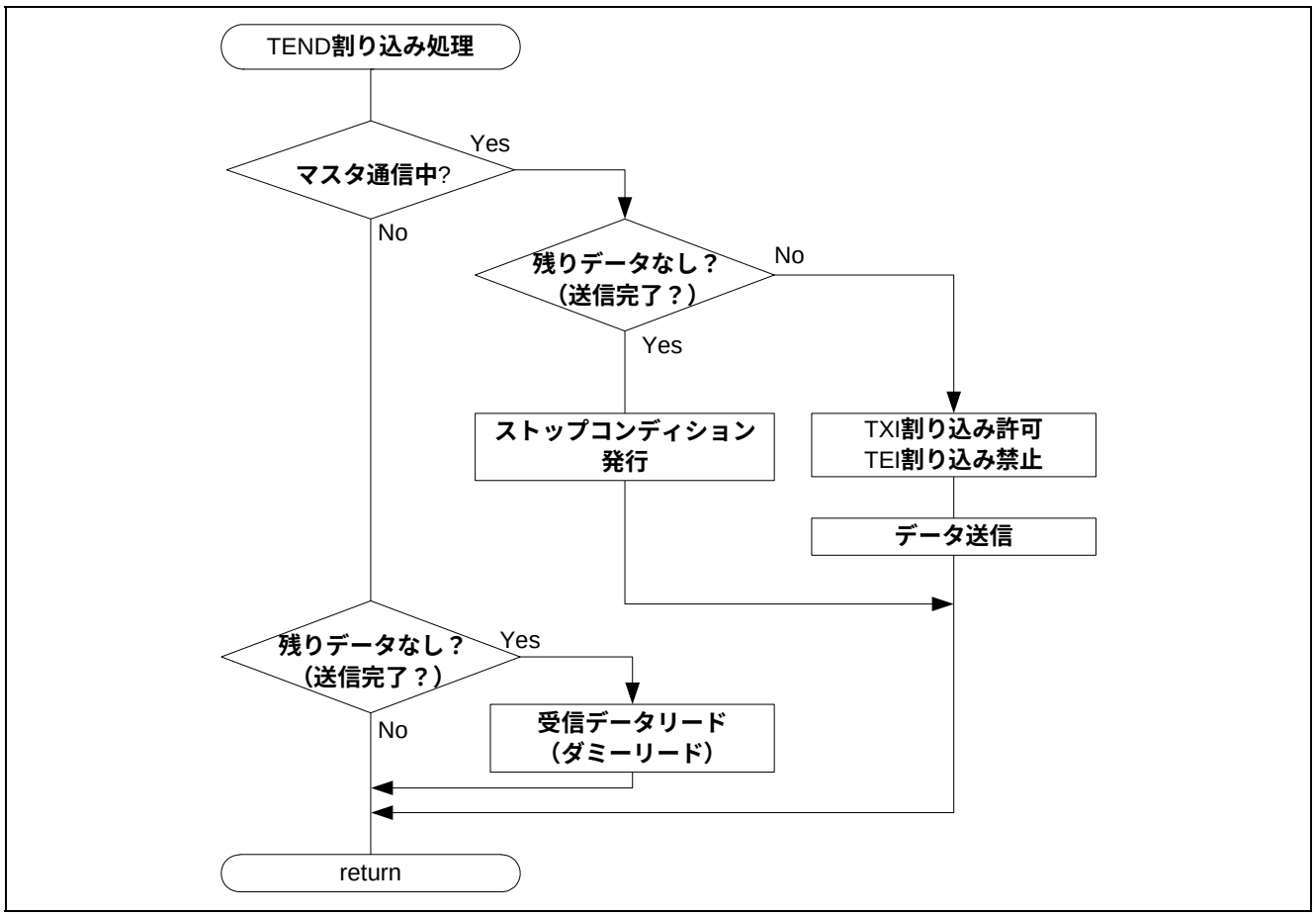

図 12　TEND 割り込み処理

3.7.12　ストップコンディション検出割り込み処理

図 13 にストップコンディション検出割り込み処理のフローチャートを示します。

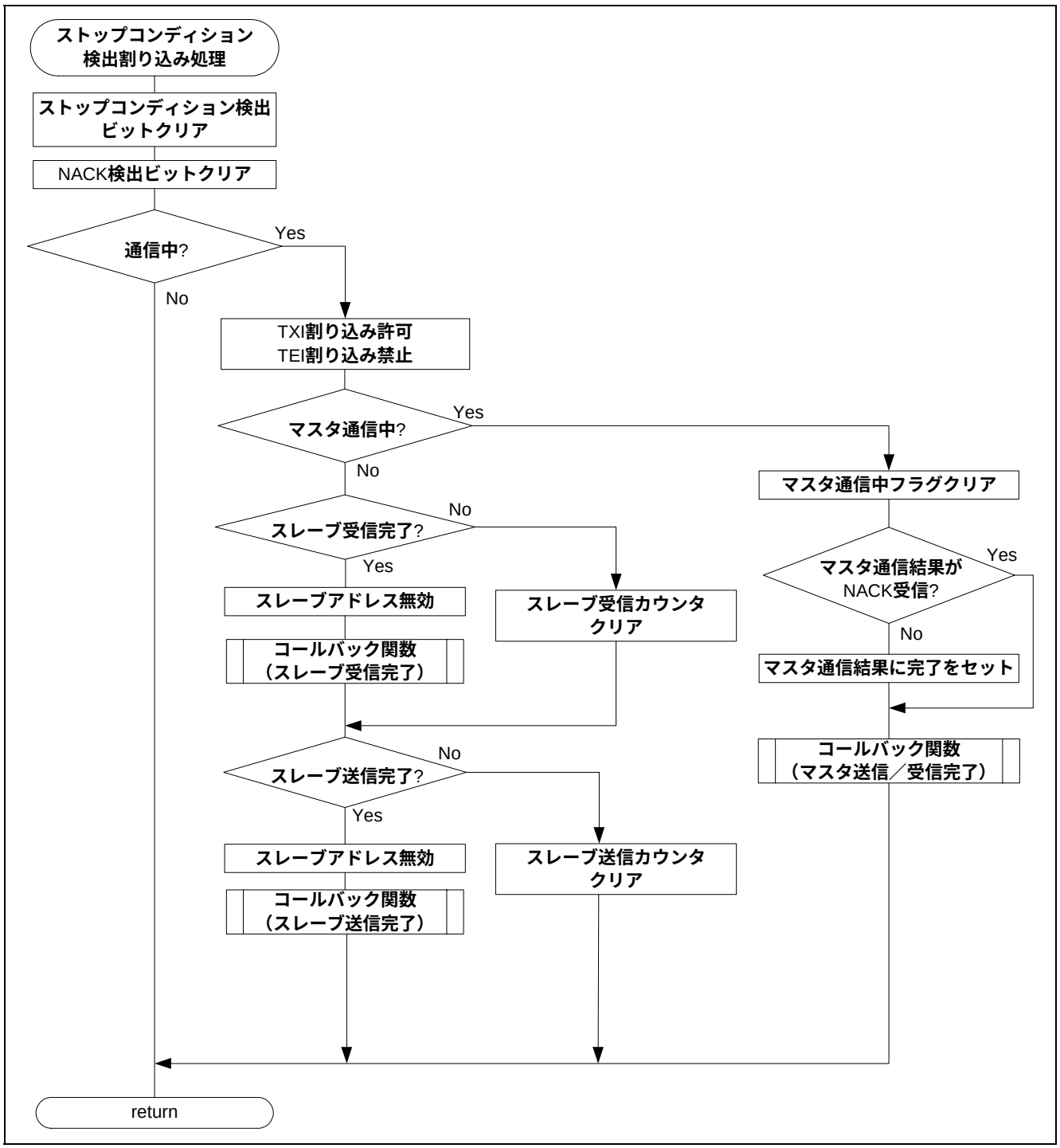

図 13　ストップコンディション検出割り込み処理

3.7.13　NACK 割り込み処理

図 14 に NACK 割り込み処理割り込み処理のフローチャートを示します。

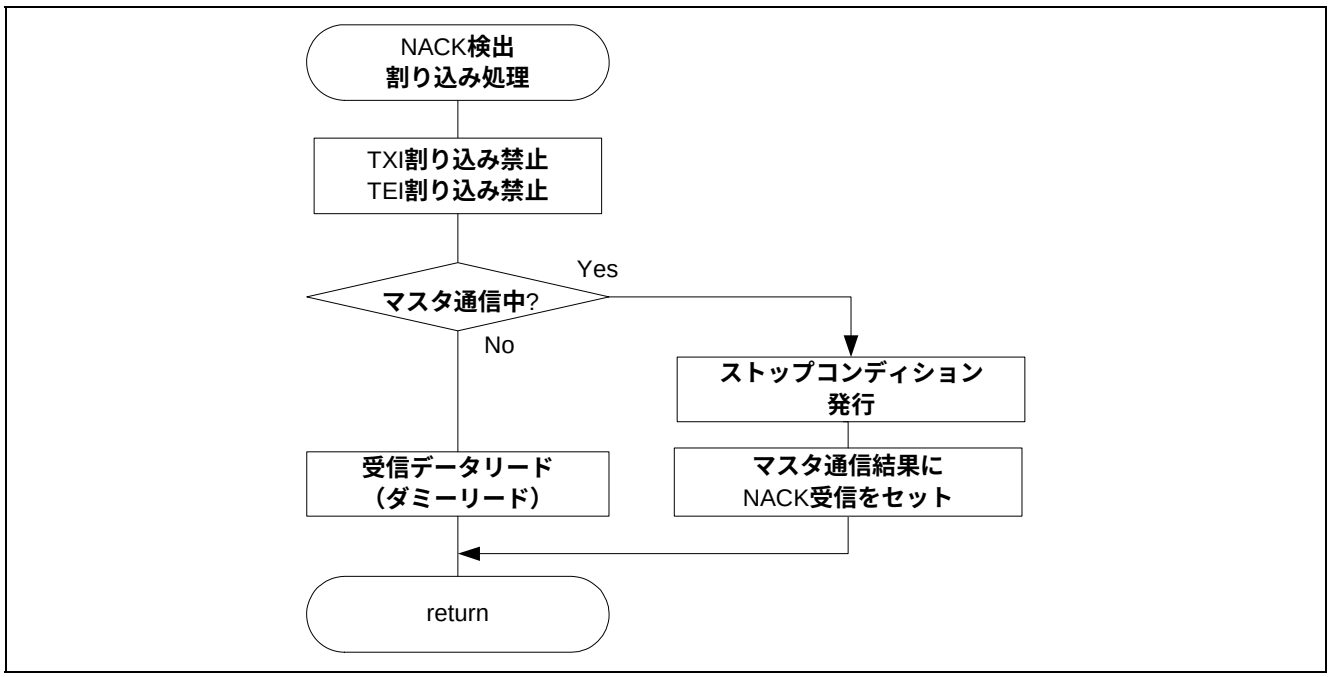

図 14　NACK 割り込み処理

3.7.14　アービトレーションロスト検出割り込み処理

図 15 にアービトレーションロスト検出割り込み処理のフローチャートを示します。

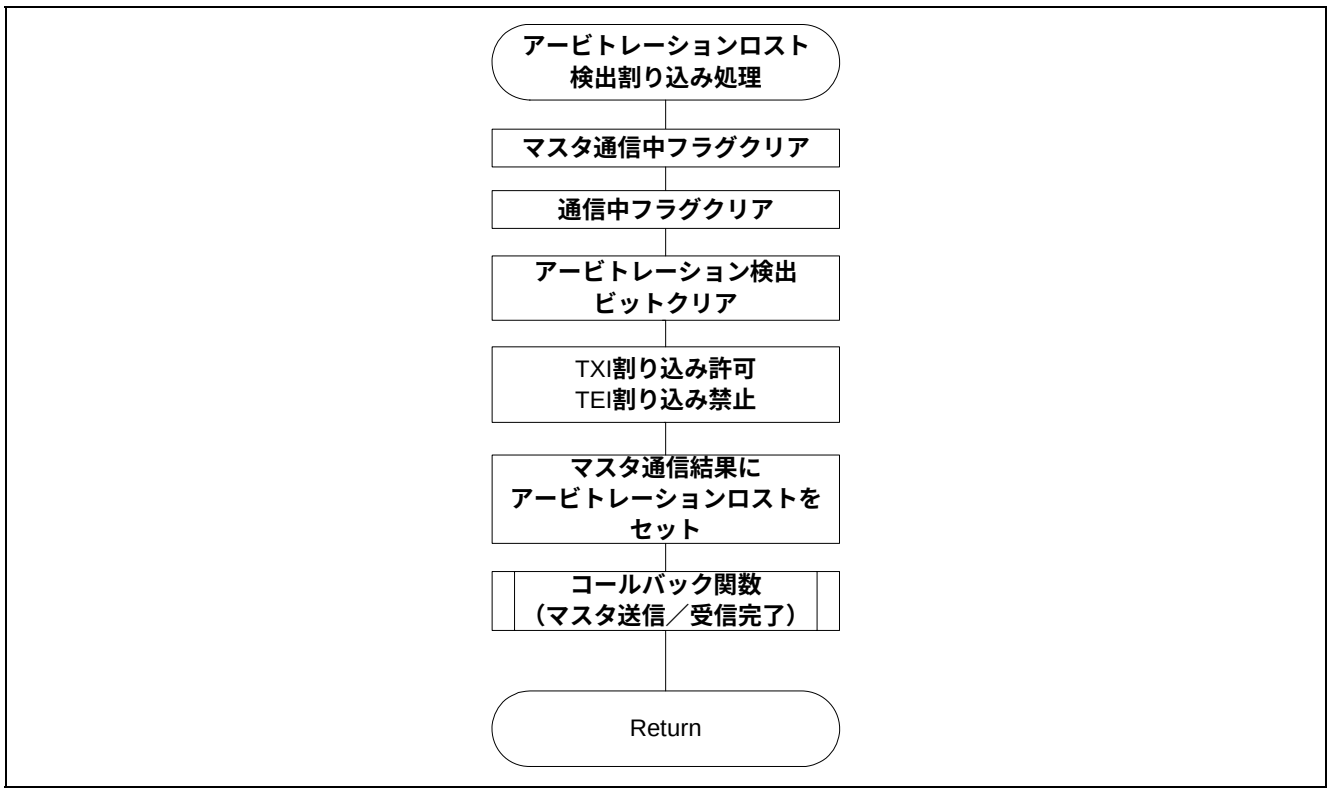

図 15　アービトレーションロスト検出割り込み処理

3.7.15　タイムアウト検出割り込み処理

図 16 にタイムアウト検出割り込み処理のフローチャートを示します。

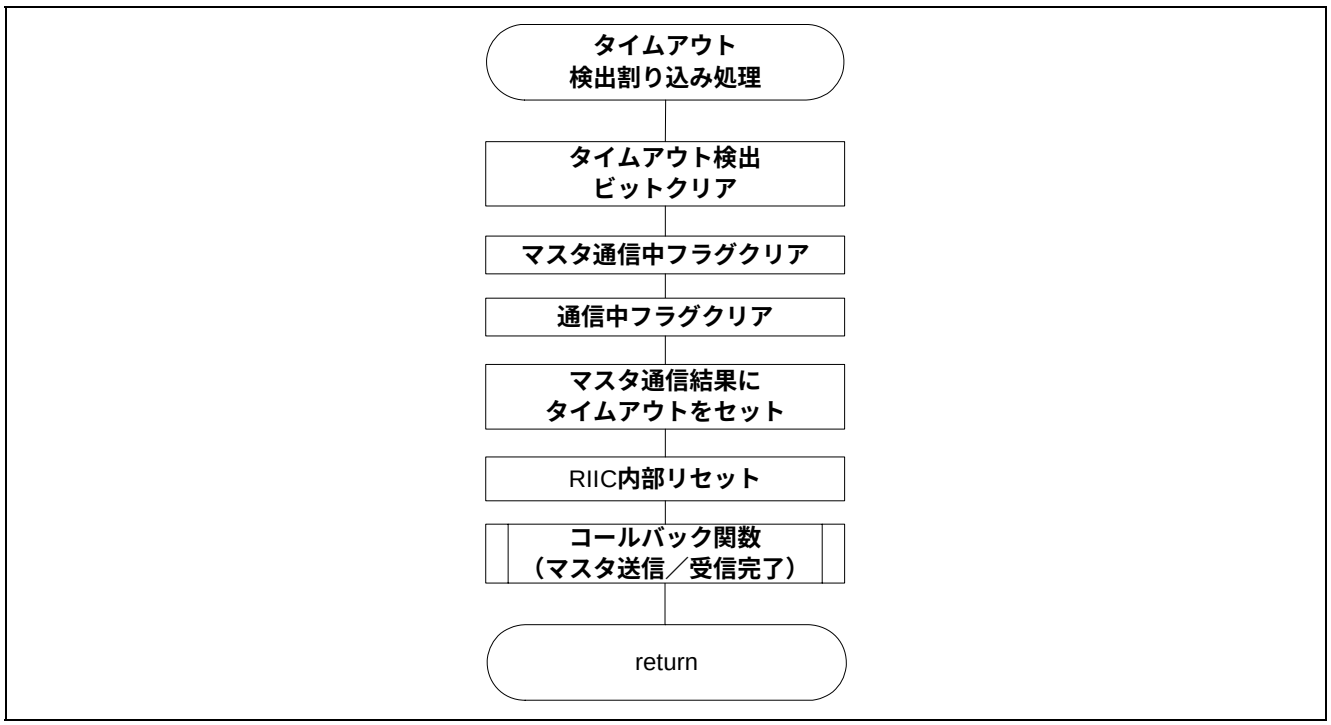

図 16　タイムアウト割り込み処理

4. 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。

5. 参考ドキュメント

- RX62N グループ、RX621 グループ ユーザーズマニュアル ハードウェア編　Rev.1.11
  （最新版をルネサス エレクトロニクスホームページから入手してください。）
- テクニカルアップデート/テクニカルニュース
  （最新の情報をルネサス エレクトロニクスホームページから入手してください。）
- C コンパイラマニュアル
  RX ファミリ C コンパイラパッケージ　V.1.0.1.0
  （最新版をルネサス エレクトロニクスホームページから入手してください。）

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ
http://japan.renesas.com/

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

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2011.07.15	—	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（予約領域）のアクセス禁止
   【注意】リザーブアドレス（予約領域）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（予約領域）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、製品型名ごとにシステム評価試験を実施してください。
   同じグループのマイコンでも型名が違うと、内部 ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品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ものですが、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、海底中継機器、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て、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が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

Colophon 1.0