
RX63N グループ、RX631グループ

R01AN1925JJ0100

Rev.1.00

2014.04.01

RIIC を使用した EEPROM 制御

要旨

本アプリケーションノートでは、RX63N グループ、RX631グループの I²C バスインタフェース(以下、RIIC)を使用して、EEPROM 制御を行う方法について説明します。

対象デバイス

- | | | |
|-------------|-------------|--------------------|
| ・RX63N グループ | 177、176 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX63N グループ | 145、144 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX63N グループ | 100 ピン版 | ROM 容量 : 768KB~2MB |
| ・RX631 グループ | 177、176 ピン版 | ROM 容量 : 256KB~2MB |
| ・RX631 グループ | 145、144 ピン版 | ROM 容量 : 256KB~2MB |
| ・RX631 グループ | 100 ピン版 | ROM 容量 : 256KB~2MB |

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

目次

1. 仕様.....	3
2. 動作確認条件.....	4
3. 関連アプリケーションノート	4
4. ハードウェア説明.....	5
4.1 ハードウェア構成例	5
4.2 使用端子一覧.....	5
5. ソフトウェア説明.....	6
5.1 動作概要.....	7
5.1.1 EEPROM への書き込み.....	7
5.1.2 EEPROM からの読み出し	9
5.2 ファイル構成.....	11
5.3 オプション設定メモリ	12
5.4 定数一覧.....	13
5.5 変数一覧.....	14
5.6 関数一覧.....	16
5.7 関数仕様.....	17
5.8 フローチャート	24
5.8.1 メイン処理	24
5.8.2 ポート初期設定	26
5.8.3 周辺機能初期処理.....	26
5.8.4 IRQ 初期設定	27
5.8.5 EEPROM 書き込み後のコールバック関数.....	28
5.8.6 EEPROM 読み出し後のコールバック関数.....	29
5.8.7 Debug LCD 全表示処理.....	30
5.8.8 RIIC 初期設定.....	31
5.8.9 EEPROM への書き込み開始処理.....	33
5.8.10 EEPROM からの読み出し開始処理	34
5.8.11 RIIC0 ICTXIO 割り込み処理	35
5.8.12 RIIC0 ICRXIO 割り込み処理.....	36
5.8.13 RIIC0 ICTEIO 割り込み処理	37
5.8.14 RIIC0 ICEEIO 割り込み処理.....	38
5.8.15 RIIC ステータス取得	40
5.8.16 EEPROM への書き込み/読み出し終了時の処理.....	40
6. サンプルコード	41
7. 参考ドキュメント.....	41

1. 仕様

RIIC のシングルマスタ通信を使用して、EEPROM への読み書きを行います。

リセット解除後に、LCD の表示データ(3 パターン)を EEPROM に書き込んだのち、待機状態になります。

SW3 を押下するごとに、EEPROM から LCD の表示データを 1 パターンずつ読み出し、LCD に表示します。

- 転送速度 : 100kbps
- データ転送サイズ : 1 ページ(16 バイト)単位
- アドレスフォーマット : 7 ビットアドレスフォーマット
- 動作モード : マスタ送信、マスタ受信

I²C バスの通信フォーマットについては、「RX63N グループ、RX631グループ ユーザーズマニュアル ハードウェア編」、および I²C バスの規格書を参照してください。

表 1.1に使用する周辺機能と用途を、図 1.1に動作概要を示します。

表1.1 使用する周辺機能と用途

周辺機能	用途
RIIC	EEPROM のデータ書き込み、読み出し
外部端子割り込み(以下、IRQ)	LCD 表示データ取り込み SW
I/O ポート	LCD 表示

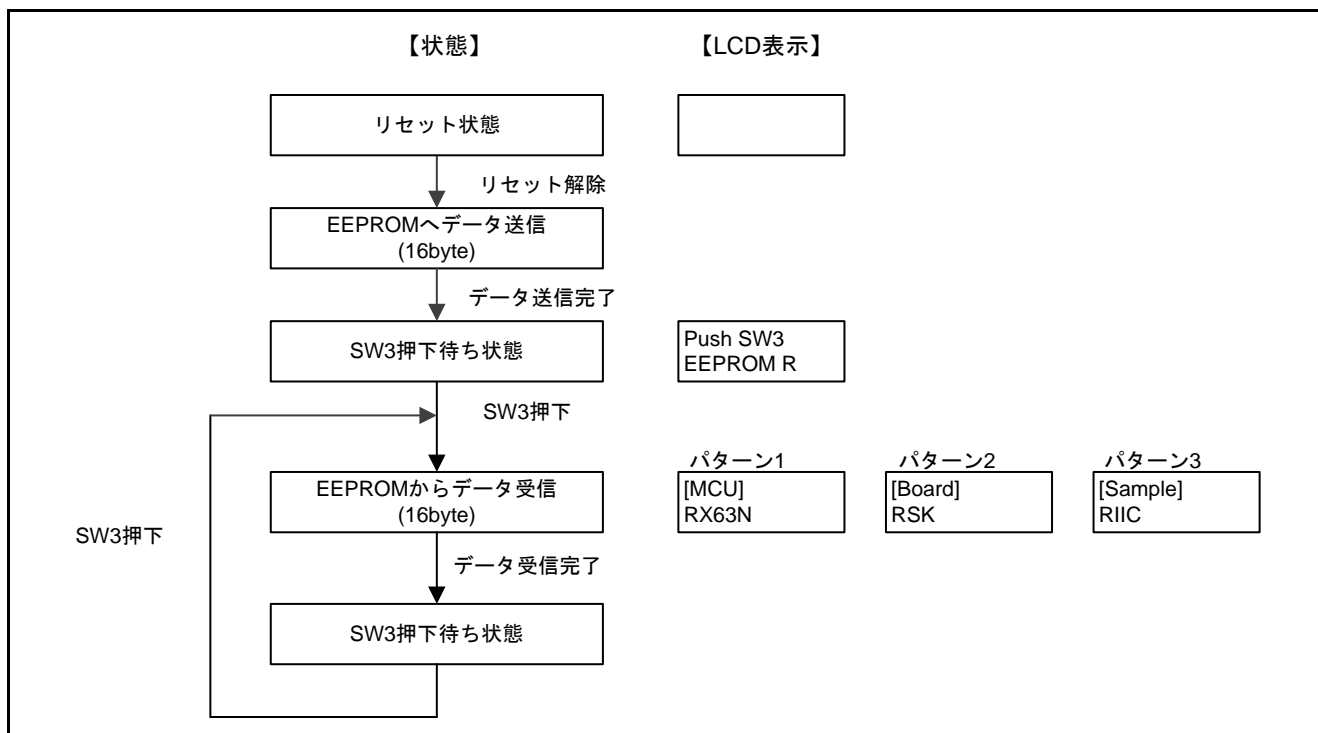


図1.1 動作概要

2. 動作確認条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表2.1 動作確認条件

項目	内容
使用マイコン	R5F563NBDDFC (RX63N グループ)
動作周波数	<ul style="list-style-type: none"> メインクロック: 12MHz PLL: 192MHz (メインクロック 1 分周 16 通倍) システムクロック (ICLK): 96MHz (PLL 2 分周) 周辺モジュールクロック B (PCLKB): 48MHz (PLL 4 分周)
動作電圧	3.3V
統合開発環境	ルネサスエレクトロニクス製 High-performance Embedded Workshop Version 4.09.01
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family V.1.02 Release 01 コンパイルオプション -cpu=rx600 -output=obj="\$(CONFIGDIR)¥\$(FILELEAF).obj" -debug -nologo (統合開発環境のデフォルト設定を使用しています)
iodefine.h のバージョン	Version 1.6A
エンディアン	リトルエンディアン
動作モード	シングルチップモード
プロセッサモード	スーパバイザモード
サンプルコードのバージョン	Version 1.00
使用ボード	Renesas Starter Kit+ for RX63N (製品型名: R0K50563NC000BE)
EEPROM	Renesas R1EX24016ASAS0A

3. 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。併せて参照してください。

- RX63N グループ、RX631 グループ 初期設定例 Rev.1.10 (R01AN1245JJ0110_RX63N)
- RX63N High-performance Embedded Workshop 用ルネサススタータキットのサンプルコード (R01AN1395JG0100_RX63N)

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

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

4. ハードウェア説明

4.1 ハードウェア構成例

図 4.1に接続例を示します。

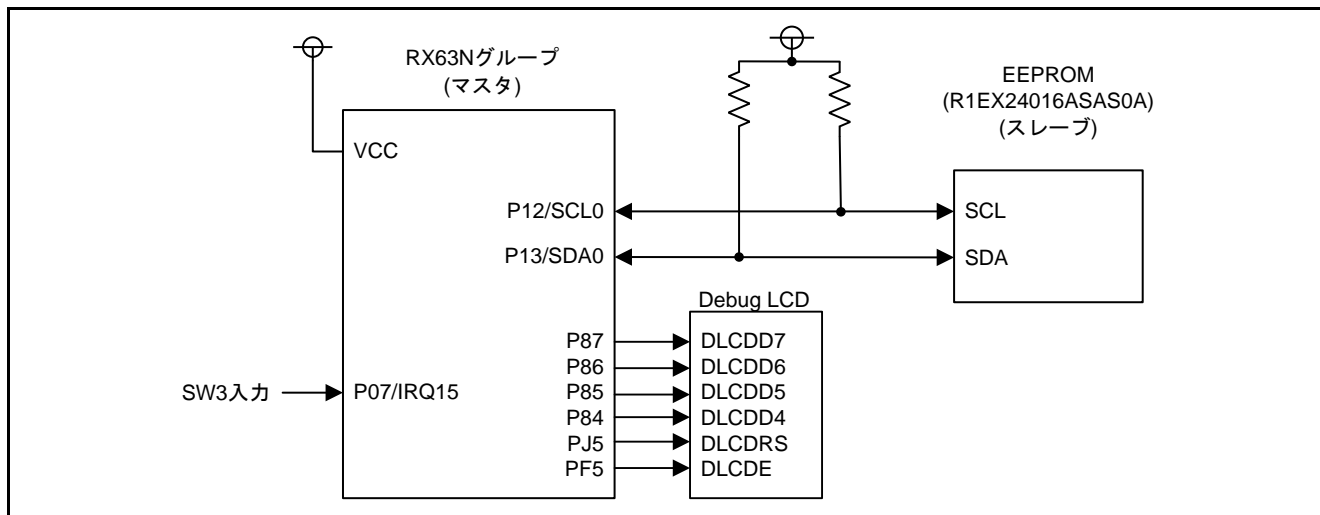


図4.1 接続例

4.2 使用端子一覧

表 4.1に使用端子と機能を示します。

使用端子は 176 ピン版の製品を想定しています。176 ピン版未満の製品を使用する場合は、使用する製品に合わせて端子を選択してください。

表4.1 使用端子と機能

端子名	入出力	内容
P12/SCL0	入出力	シリアルクロック入出力端子
P13/SDA0	入出力	シリアルデータ入出力端子
P87	出力	Debug LCD Data bit 7 出力
P86	出力	Debug LCD Data bit 6 出力
P85	出力	Debug LCD Data bit 5 出力
P84	出力	Debug LCD Data bit 4 出力
PJ5	出力	Debug LCD Enable 出力
PF5	出力	Debug LCD Register select 出力
P07/IRQ15	入力	マスター受信実行 SW 入力

5. ソフトウェア説明

リセット解除後、IIC の初期化を行います。

その後、マスタ送信を使用して EEPROM に 1 ページ(16 バイト)単位でデータの書き込みを行います。送信処理が完了すると、コールバック関数を呼び出します。コールバック関数で IIC ステータスを取得し、マスタ送信が正常に終了していたかを確認します。マスタ送信が正常終了していた場合、次のデータの送信準備を行います。NACK 受信、またはアービトレーションロスト検出で終了していた場合、同じデータの書き込み準備を行います。タイムアウトが発生した場合、LCD に “[ERROR] TIMEOUT” と表示してループ処理を実行します。

3 ページのデータが書き込み完了したとき、LCD に “Push SW3” を表示します。

SW3 が押下されると、マスタ受信を使用して EEPROM から 1 ページ分のデータを読み出します。受信処理が完了すると、コールバック関数を呼び出します。コールバック関数で IIC ステータスを取得し、マスタ受信が正常に終了していたかを確認します。マスタ受信が正常終了していた場合、LCD に EEPROM から読み出したデータを表示します。エラーが発生した場合は、LCD に発生したエラー情報を表示します。

使用する周辺機能の設定を以下に示します。

<IIC>

- ・動作モード : マスタ送信、マスタ受信
- ・アドレスフォーマット : 7 ビットアドレスフォーマット
- ・内部基準クロック : PCLK/8 クロック
- ・転送速度 : 約 100kbps(注 1)
- ・使用する割り込み : 送信データエンプティ(ICTXI)
: 送信終了割り込み(ICTEI)
: 受信データフル割り込み(ICRXI)
: ストップコンディション検出割り込み(SPI)
: アービトレーションロスト割り込み(ALI)

注 1. ICBRL.BRL[4:0]=23、ICBRH.BRH[4:0]=28 を設定しています。

実際の転送速度は、SCL ラインの立ち上がり時間(tr)を 1000ns、SCL ラインの立ち下がり時間(tf)を 300ns とした場合、

$$\begin{aligned} \text{転送速度} &= 1 / \{ [(ICBRH.BRH[4:0]+1) + (ICBRL.BRL[4:0]+1)] / \text{内部基準クロック} + tr + tf \} \\ &= 1 / \{ [(28+1) + (23+1)] / (48\text{MHz} / 8) + 1000(\text{ns}) + 300(\text{ns}) \} \\ &= 98684.2\text{bps} \end{aligned}$$

となります。

5.1 動作概要

5.1.1 EEPROM への書き込み

- (1) マスタ送信開始
ICCR2.BBSY フラグが“0” (バスフリー)であることを確認して RIIC ステータス変数(riic_status)を RIIC_BUSY(ビジー)、ICIER.TIE ビットを“1” (ICTXIO 割り込みを許可)、ICCR2.ST ビットを“1” (スタートコンディションの発行を要求する)にします。
- (2) スタートコンディション発行
スタートコンディションが発行されると ICCR2.BBSY フラグが“1” (バスビジー)になります。また ICTXIO 割り込み要求が発生します。TXIO 割り込み処理で ICDRT レジスタに EEPROM のスレーブアドレスと W ビット(“0”)を書きます。
- (3) データ送信
ICDRT レジスタから ICDRS レジスタにデータが転送されると、再び ICTXIO 割り込み要求が発生します。ICTXIO 割り込み処理で ICDRT レジスタに EEPROM への書き込みアドレスを書きます。
- (4) ACK/NACK 受信
SCL の 9 ビット目の立ち上がりで、EEPROM から ACK/NACK 信号が出力されます。
ACK を受信した場合は、RIIC 通信を続行し、ICTXIO 割り込み要求が発生します。ICTXIO 割り込み処理で ICDRT レジスタに EEPROM への書き込みデータを 1 バイト目から書き込みします。
NACK を受信した場合は、RIIC 通信が中断され ICEEIO 割り込み要求が発生します。ICEEIO 割り込み処理で、RIIC ステータス変数を RIIC_NACK(NACK 受信)に設定し、ICCR2.SP ビットを“1” (ストップコンディションの発行を要求する)にします。
- (5) 最終データの書き込み
1 ページ(16 バイト)分の EEPROM への書き込みデータを ICDRT レジスタに書き込んだ後、ICIER.TIE ビットを“0” (ICTXIO 割り込みを禁止)、ICIER.TEIE ビットを“1” (ICTEIO 割り込みを許可)にします。
- (6) 送信終了
最終データの送信が終了すると、TEIO 割り込み要求が発生します。TEIO 割り込み処理で ICIER.TEIE ビットを“0” (ICTEIO 割り込みを禁止)、ICCR2.SP ビットを“1” (ストップコンディションの発行を要求する)にします。
- (7) ストップコンディション発行
ストップコンディションが発行されると EEIO 割り込み要求が発生します。EEIO 割り込み処理で、送信が正常終了していた場合は、RIIC ステータス変数を RIIC_RDY(レディ)に設定し、コールバック関数を呼び出します。
NACK 受信後のストップコンディション発行時(RIIC ステータス変数が RIIC_NACK のとき)、RIIC ステータス変数を設定せずにコールバック関数を呼び出します。

図 5.1にEEPROM への書き込みタイミングを示します。

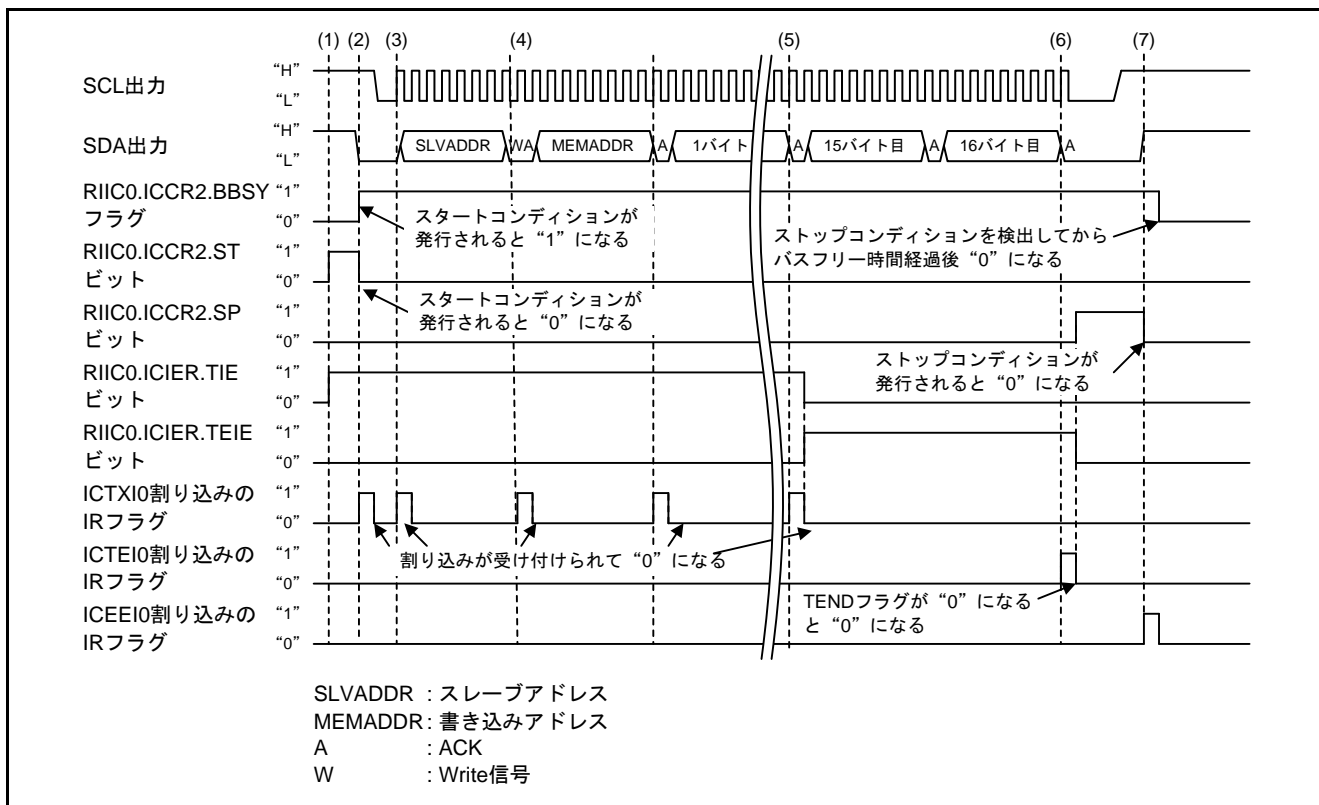


図5.1 EEPROM への書き込みタイミング

5.1.2 EEPROM からの読み出し

- (1) マスタ送信開始
IRQ15 端子の立ち下がりを検出したとき、ICCR2.BBSY フラグが“0” (バスマスター)であることを確認して RIIC ステータス変数(riic_status)を RIIC_BUSY(ビジー)、ICIER.TIE ビットを“1” (ICTXIO 割り込みを許可)、ICIER.RIE ビットを“1” (ICRXIO 割り込みを許可)、ICCR2.ST ビットを“1” (スタートコンディションの発行を要求する)にします。
- (2) スタートコンディション発行
スタートコンディションが発行されると ICCR2.BBSY フラグが“1” (バスマスター)になります。また ICTXIO 割り込み要求が発生します。ICTXIO 割り込み処理で ICDRT レジスタに EEPROM のスレーブアドレスと W ビット(“0”)を書きます。
- (3) アドレス送信
ICDRT レジスタから ICDRS レジスタにデータが転送されると、再び ICTXIO 割り込み要求が発生します。ICTXIO 割り込み処理で ICDRT レジスタに EEPROM への読み出しアドレスを書き込んだ後、ICIER.TIE ビットを“0” (ICTXIO 割り込みを禁止)、ICIER.TEIE ビットを“1” (ICTEIO 割り込みを許可)にします。
- (4) ACK/NACK 受信
SCL の 9 ビット目の立ち上がりで、EEPROM から ACK/NACK 信号が出力されます。ACK を受信した場合は、RIIC 通信を続行します。
NACK を受信した場合、RIIC 通信が中断され ICEEIO 割り込み要求が発生します。ICEEIO 割り込み処理で、RIIC ステータスを RIIC_NACK(NACK 受信)に設定し、ICCR2.SP ビットを“1” (ストップコンディションの発行を要求する)にします。
- (5) マスタ受信開始
読み出しアドレスの送信が終了すると、TEIO 割り込み要求が発生します。TEIO 割り込み処理で ICIER.TEIE ビットを“0” (ICTEIO 割り込みを禁止)、ICIER.TIE ビットを“1” (ICTXIO 割り込みを許可)、ICCR2.RS ビットを“1” (リスタートコンディションの発行を要求する)にします。
- (6) リスタートコンディション発行
リスタートコンディションが発行されると、ICTXIO 割り込み要求が発生します。ICTXIO 割り込み処理で、ICDRT レジスタに EEPROM のスレーブアドレスと R ビット(“1”)を書きます。
- (7) ACK/NACK 受信
SCL の 9 ビット目の立ち上がりで、EEPROM から ACK/NACK 信号が出力されます。ACK を受信した場合は、ICCR2.TRS ビットが“0” (受信モード)になり、ICRXIO 割り込み要求が発生します。ICRXIO 割り込み処理で、ICDRR レジスタをダミーリードします。
NACK を受信した場合、RIIC 通信が中断され、ICEEIO 割り込み要求が発生します。ICEEIO 割り込み処理で、RIIC ステータス変数を RIIC_NACK(NACK 受信)に設定し、ICCR.SP ビットを“1” (ストップコンディションの発行を要求する)にします。
- (8) データの受信完了
データの受信が完了すると、ICRXIO 割り込み要求が発生します。また、SCL の 9 ビット目に ICMR3.ACKBT ビットの値が出力されます。ICRXIO 割り込み処理で、ICDRR レジスタから受信データを読み込みます。
- (9) 最終データの受信完了
最終データの受信完了時、ICCR2.SP ビットを“1” (ストップコンディションの発行を要求する)にした後に、ICDRR レジスタから受信データを読み込みます。
- (10) ストップコンディションの発行
ストップコンディションが発行されると EEIO 割り込み要求が発生します。EEIO 割り込み処理で、受信が正常終了していた場合は、RIIC ステータス変数を RIIC_RDY(レディ)に設定し、コールバック関数を呼び出します。
NACK 受信後のストップコンディション発行時(RIIC ステータス変数が RIIC_NACK のとき)、RIIC ステータス変数を設定せずにコールバック関数を呼び出します。

図 5.2にマスタ受信のタイミングを示します。

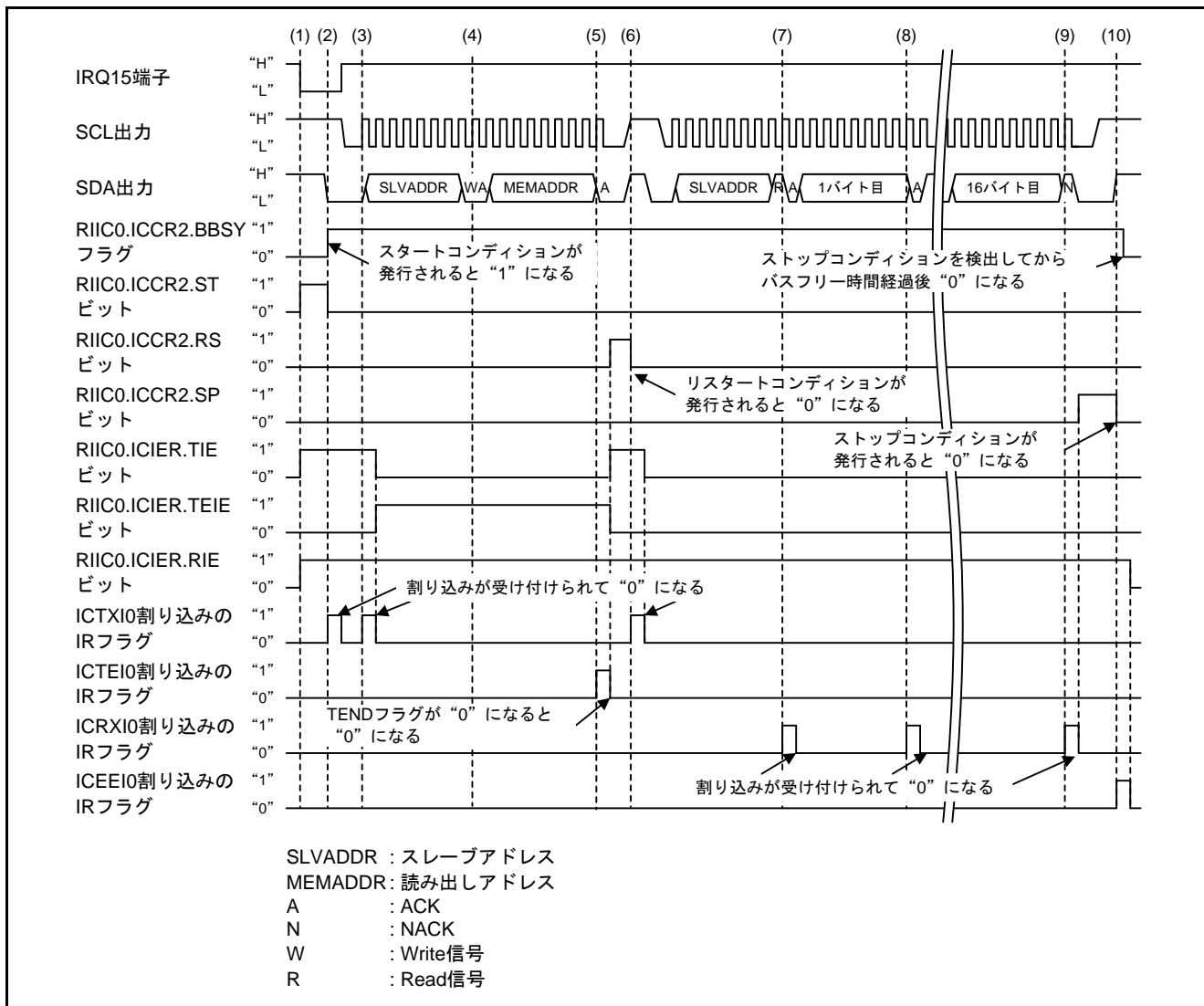


図5.2 マスタ受信のタイミング

5.2 ファイル構成

表 5.1にサンプルコードで使用するファイル、表 5.2に標準インクルードファイル、表 5.3に参照する関連アプリケーションノートの関数と設定値を示します。なお、統合開発環境で自動生成されるファイルは除きます。

表5.1 サンプルコードで使用するファイル

ファイル名	概要
main.c	メイン処理
riic.c	EEPROM への書き込み、読み出し処理
riic.h	riic.c のヘッダファイル

表5.2 標準インクルードファイル

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

表5.3 参照する関連アプリケーションノートの関数と設定値

ファイル名	関数	設定値
r_init_stop_module.c	R_INIT_StopModule()	-
r_init_stop_module.h	-	DMAC/DTC、EXDMAC、RAM0、RAM1 はモジュールストップ解除
r_init_non_existent_port.c	R_INIT_NonExistentPort()	-
r_init_non_existent_port.h	-	176 ピン版を設定
r_init_clock.c	R_INIT_Clock()	-
r_init_clock.h	-	サブクロック未使用
lcd.c	Init_LCD() Display_LCD()	-
lcd.h	-	-
rskrx63ndef.h	-	-

5.3 オプション設定メモリ

表 5.4にサンプルコードで使用するオプション設定メモリの状態を示します。必要に応じて、お客様のシステムに最適な値を設定してください。

表5.4 サンプルコードで使用するオプション設定メモリ

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

5.4 定数一覧

表 5.5～表 5.7にサンプルコードで使用する定数を示します。

表5.5 サンプルコードで使用する定数(main.c)

定数名	設定値	内容
EEPROM_ADDR	0x50	EEPROM のデバイスアドレス
EEPROM_PAGE_SIZE	16	EEPROM のページサイズ
EEPROM_RW_SIZE	16	EEPROM へのリード/ライトサイズ
DISP_PTN_NUM	3	表示文字列のパターン数
LCD_LINE_SIZE	8	LCD1 行の文字数
MSG_TIMEOUT_ERROR	"[ERROR] TIMEOUT "	タイムアウトエラー時のメッセージ
MSG_AL_ERROR	"[ERROR] AL "	アービトレーションロスト発生時のメッセージ
MSG_NACK_ERROR	"[ERROR] NACK "	NACK 受信時のメッセージ
SW3_REQ	IR(ICU, IRQ15)	SW3(IRQ15)の IR フラグ
SW_ON	1	SW3 押下検出
SW_OFF	0	SW3 押下未検出

表5.6 サンプルコードで使用する定数(iic.h)

定数名	設定値	内容
RIIC_RDY	0x00	RIIC ステータス(レディ)
RIIC_BUSY	0x01	RIIC ステータス(ビジー)
RIIC_NACK	0x02	RIIC ステータス(NACK 受信)
RIIC_AL	0x03	RIIC ステータス(アービトレーションロスト発生)
RIIC_TIMEOUT	0x04	RIIC ステータス(タイムアウト)
RIIC_OK	0x00	RIIC 処理正常終了
RIIC_FAIL	0x01	RIIC 処理失敗
RIIC_PARAM_ERR	0x02	パラメータエラー
NULL	0x00000000	NULL 指定

表5.7 サンプルコードで使用する定数(iic.c)

定数名	設定値	内容
RIIC_W_BIT	0xFE	ライトビット設定用マスクデータ
RIIC_R_BIT	0x01	リードビット設定用データ
RIIC_WRITE_SLVADDR	1	スレーブアドレスのサイズ(バイト)
RIIC_WRITE_MEMADDR	1	メモリアドレスのサイズ(バイト)
RIIC_WRITE_HEADER	RIIC_WRITE_SLVADDR + RIIC_WRITE_MEMADDR	アドレス指定部のサイズ
RIIC_MD_INIT	0	RIIC 動作モード(初期化)
RIIC_MD_WRITE	1	RIIC 動作モード(データ書き込みモード)
RIIC_MD_READ_ADDR	2	RIIC 動作モード(リードアドレス設定モード)
RIIC_MD_READ_DATA	3	RIIC 動作モード(データ読み出しモード)
RIIC_CNT_SLVADDR	0	スレーブアドレス送信判定用
RIIC_CNT_MEMADDR	1	メモリアドレス送信判定用

5.5 変数一覧

表 5.8、表 5.9にstatic 型変数を、表 5.10にconst 型変数を示します。

表5.8 static 型変数(main.c)

型	変数名	内容	使用関数
uint8_t	read_buf[16]	EEPROM からの読み出しデータ格納 バッファ	main cb_read_end
uint8_t	wpt	書き込みポインタ	main cb_write_end
uint8_t	rpt	読み出しポインタ	main cb_read_end

表5.9 static 型変数(riic.c)

型	変数名	内容	使用関数
uint8_t	riic_status	IIC ステータス	IIC_Init IIC_Write IIC_Read Excep_RIIC0_EEIO IIC_GetStatus
uint8_t	riic_slv_addr	EEPROM スレーブアドレス格納領域	IIC_Write IIC_Read Excep_RIIC0_TXIO Excep_RIIC0_TEIO
uint8_t	riic_mem_addr	EEPROM メモリアドレス格納領域	IIC_Write IIC_Read Excep_RIIC0_TXIO
uint8_t*	p_riic_rx_buf	EEPROM からの読み出しバッファ格納ポインタ	IIC_Read Excep_RIIC0_RXIO
uint8_t*	p_riic_tx_buf	EEPROM への書き込みバッファ格納ポインタ	IIC_Write Excep_RIIC0_TXIO
uint8_t	riic_tx_cnt	IIC 送信カウンタ	IIC_Write IIC_Read Excep_RIIC0_TXIO Excep_RIIC0_TEIO
uint8_t	riic_tx_size	IIC 送信サイズ	IIC_Write IIC_Read Excep_RIIC0_TXIO Excep_RIIC0_TEIO
uint8_t	riic_rx_cnt	IIC 受信カウンタ	IIC_Write IIC_Read Excep_RIIC0_RXIO
uint8_t	riic_rx_size	IIC 受信サイズ	IIC_Write IIC_Read Excep_RIIC0_RXIO
uint8_t	riic_rw_mode	IIC R/W モード	IIC_Init IIC_Write IIC_Read Excep_RIIC0_TXIO Excep_RIIC0_TEIO Excep_RIIC0_EEIO
CallBackFunc	pcb_riic_end	コールバック関数ポインタ格納領域	IIC_Write IIC_Read Excep_RIIC0_EEIO

表5.10 const 型変数(main.c)

型	変数名	内容	使用関数
static const uint8_t	write_buf[DISP_P TN_NUM][16]	EEPROM への書き込み文字列 パターン 1: "[MCU] RX63N " パターン 2: "[Board] RSK " パターン 3: "[Sample]IIC "	main

5.6 関数一覧

表 5.11に関数を示します。

表5.11 関数(main.c)

関数名	概要
main	メイン処理
port_init	ポート初期設定
R_INIT_StopModule	リセット後に動作している周辺機能の停止
R_INIT_NonExistentPort	存在しないポートの初期設定
R_INIT_Clock	クロック初期設定
peripheral_init	周辺機能初期設定
IRQ_Init	IRQ 初期設定
cb_write_end	EEPROM 書き込み後のコールバック関数
cb_read_end	EEPROM 読み出し後のコールバック関数
Display_LCD_All	Debug LCD 全表示処理
Init_LCD	Debug LCD の初期設定
Display_LCD	Debug LCD の表示処理

表5.12 関数(iic.c)

関数名	概要
IIC_Init	IIC 初期設定
IIC_Write	EEPROM への書き込み開始処理
IIC_Read	EEPROM からの読み出し開始処理
Excep_IIC0_TXI0	IIC0 ICTXI0 割り込み処理
Excep_IIC0_RXI0	IIC0 ICRXI0 割り込み処理
Excep_IIC0_TEI0	IIC0 ICTEI0 割り込み処理
Excep_IIC0_EEI0	IIC0 ICEEI0 割り込み処理
IIC_GetStatus	IIC ステータス取得
IIC_proc_end	EEPROM への書き込み/読み出し終了時の処理

5.7 関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	初期設定後、LCD への表示データを EEPROM への書き込みを行います。書き込み後、IRQ15 端子の立ち下がリエッジを検出すると、EEPROM から LCD 表示データを読み出します。
引数	なし
リターン値	なし
port_init	
概要	ポート初期設定
ヘッダ	なし
宣言	static void port_init(void)
説明	ポートの初期設定を行います。
引数	なし
リターン値	なし
R_INIT_StopModule	
概要	リセット後に動作している周辺機能の停止
ヘッダ	r_init_stop_module.h
宣言	void R_INIT_StopModule(void)
説明	モジュールストップ状態へ遷移する設定を行います。
引数	なし
リターン値	なし
備考	サンプルコードでは、モジュールストップ状態への遷移は行っていません。 本関数の詳細は、アプリケーションノート「RX63N グループ、RX631 グループ 初期設定例 Rev.1.10」を参照してください。

R_INIT_NonExistentPort

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

R_INIT_Clock

概要	クロック初期設定
ヘッダ	r_init_clock.h
宣言	void R_INIT_Clock(void)
説明	クロックの初期設定を行います。
引数	なし
リターン値	なし
備考	サンプルコードでは、システムクロックを PLL とし、サブクロックを使用しない処理を選択しています。 本関数の詳細は、アプリケーションノート「RX63N グループ、RX631 グループ 初期設定例 Rev.1.10」を参照してください。

peripheral_init

概要	周辺機能初期設定
ヘッダ	なし
宣言	static void peripheral_init(void)
説明	使用する周辺機能の初期設定を行います。
引数	なし
リターン値	なし

IRQ_Init

概要	IRQ 初期設定
ヘッダ	なし
宣言	static void IRQ_Init(void)
説明	IRQ15 の初期設定を行います。
引数	なし
リターン値	なし

cb_write_end	
概要	EEPROM 書き込み後のコールバック関数
ヘッダ	なし
宣言	static void cb_write_end (void)
説明	RIIC ステータスを取得し、EEPROM の書き込みが完了していた場合は、書き込みポインタを更新します。タイムアウトが発生していた場合、LCD にタイムアウトエラーのメッセージを表示します。
引数	なし
リターン値	なし

cb_read_end	
概要	EEPROM 読み出し後のコールバック関数
ヘッダ	なし
宣言	static void cb_read_end (void)
説明	RIIC ステータスを取得し、EEPROM の読み出しが完了していた場合は、読み出したデータを LCD に表示し、読み出しポインタを更新します。エラーが発生していた場合は、エラーメッセージを表示します。
引数	なし
リターン値	なし

Display_LCD_All	
概要	Debug LCD 全表示処理
ヘッダ	なし
宣言	static void Display_LCD_All(uint8_t *p_disp)
説明	LCD に指定した文字列を表示します。
引数	uint8_t* p_disp 表示文字列
リターン値	なし

Init_LCD	
概要	Debug LCD 初期設定
ヘッダ	lcd.h
宣言	void Init_LCD(void)
説明	LCD の初期化を行います
引数	なし
リターン値	なし
備考	本関数は RX63N High-performance Embedded Workshop 用ルネサススタータキットのサンプルコードを使用しています。

Display_LCD	
概要	Debug LCD 表示処理
ヘッダ	lcd.h
宣言	void Display_LCD(uint8_t position, uint8_t* string)
説明	LCD へ指定した文字を表示します。
引数	uint8_t position 文字列の表示位置 uint8_t* string 表示する文字列
リターン値	なし
備考	本関数は RX63N High-performance Embedded Workshop 用ルネサススタータキットのサンプルコードを使用しています。
RIIC_Init	
概要	RIIC 初期設定
ヘッダ	riic.h
宣言	void RIIC_Init (void)
説明	RIIC の初期設定を行います。
引数	なし
リターン値	なし
RIIC_Write	
概要	EEPROM への書き込み開始処理
ヘッダ	riic.h
宣言	uint8_t RIIC_Write (uint8_t slv_addr, uint8_t mem_addr, uint8_t * pbuf, uint8_t num, CallbackFunc pcb)
説明	EEPROM への書き込み情報を設定した後、スタートコンディションを出力して EEPROM への書き込みを開始します。
引数	uint8_t slv_addr : EEPROM のスレーブアドレス uint8_t mem_addr : EEPROM の書き込みアドレス uint8_t* pbuf : EEPROM への書き込みデータ格納ポインタ uint8_t num : EEPROM への書き込みサイズ CallbackFunc pcb : 書き込み完了後のコールバック関数
リターン値	RIIC_OK : EEPROM への書き込み開始処理成功 RIIC_FAIL : EEPROM への書き込み開始処理失敗

IIC_Read	
概要	EEPROM からの読み出し開始処理
ヘッダ	riic.h
宣言	uint8_t IIC_Read (uint8_t slv_addr, uint8_t mem_addr, uint8_t * pbuf, uint8_t num, CallbackFunc pcb)
説明	EEPROM からの読み出し情報を設定した後、スタートコンディションを出力して EEPROM からの読み出しを開始します。
引数	uint8_t slv_addr : EEPROM のスレーブアドレス uint8_t mem_addr : EEPROM の読み出しアドレス uint8_t* pbuf : EEPROM から読み出しデータ格納ポインタ uint8_t num : EEPROM への読み出しサイズ CallbackFunc pcb : 読み出し完了後のコールバック関数
リターン値	RIIC_OK : EEPROM への読み出し開始処理成功 RIIC_FAIL : EEPROM への読み出し開始処理失敗 RIIC_PARAM_ERR : 引数の値が不正

Excep_RIIC0_TXI0	
概要	RIIC0 ICTXI0 割り込み処理
ヘッダ	なし
宣言	static void Excep_RIIC0_TXI0(void)
説明	ICTXI0 割り込み処理を行います。送信データを ICDRT レジスタに書き込みます。最終データの書き込み後、ICTXI0 割り込み要求を禁止にします。また、マスタ送信時は ICTEI0 割り込み要求を許可にします。
引数	なし
リターン値	なし

Excep_RIIC0_RXI0	
概要	RIIC0 ICRXI0 割り込み処理
ヘッダ	なし
宣言	static void Excep_RIIC0_RXI0(void)
説明	ICRXI0 割り込み処理を行います。ICDRR レジスタから受信データを読み出します。最終データの受信時は、ストップコンディションを出力してから受信データを読み出します。
引数	なし
リターン値	なし

Excep_RIIC0_TEI0	
概要	RIIC0 ICTEI0 割り込み処理
ヘッダ	なし
宣言	static void Excep_RIIC0_TEI0(void)
説明	ICTEI0 割り込み処理を行います。 EEPROM 書き込み処理での ICTEI0 割り込み処理の場合は、ICTEI0 割り込み要求を禁止にした後、ストップコンディションを出力します。 EEPROM 読み出し処理での ICTEI0 割り込み処理の場合は、ICTEI0 割り込み要求を禁止にした後、リスタートコンディションを出力します。
引数	なし
リターン値	なし

Excep_RIIC0_EEI0	
概要	RIIC0 ICEEI0 割り込み処理
ヘッダ	なし
宣言	static void Excep_RIIC0_EEI0(void)
説明	ICEEI0 割り込み処理を行います。 割り込み発生要因がストップコンディション検出の場合、かつ NACK 受信でない場合は、RIIC ステータスを RIIC_RDY に設定してコールバック関数を呼び出します。 割り込み発生要因がストップコンディション検出の場合、かつ NACK 受信の場合は、RIIC ステータスを変更せずコールバック関数を呼び出します。 割り込み発生要因がアービトレーションロスト発生の場合、RIIC ステータスを RIIC_AL に設定してコールバック関数を呼び出します。 割り込み発生要因がタイムアウトの場合、RIIC ステータスを RIIC_TIMEOUT に設定してコールバック関数を呼び出します。 割り込み発生要因が NACK 受信の場合、RIIC ステータスを RIIC_NACK に設定して、ストップコンディションを出力します。
引数	なし
リターン値	なし

IIC_GetStatus	
概要	IIC ステータス取得
ヘッダ	riic.h
宣言	uint8_t IIC_GetStatus (void)
説明	IIC ステータスを返します。
引数	なし
リターン値	RIIC_RDY : レディ状態 RIIC_BUSY : ビジー状態 RIIC_NACK : NACK 受信 RIIC_AL : アービトレーションロスト発生 RIIC_TIMEOUT: タイムアウト発生

IIC_proc_end	
概要	EEPROM への書き込み/読み出し終了時の処理
ヘッダ	なし
宣言	static void IIC_proc_end(void)
説明	すべての IIC0 割り込み要求を禁止にし、ICMR3.RDRFS ビット、ICMR3.ACKBT ビット、ICMR3.WAIT ビットを初期化する。
引数	なし
リターン値	なし

5.8 フローチャート

5.8.1 メイン処理

図 5.3にメイン処理のフローチャートを示します。

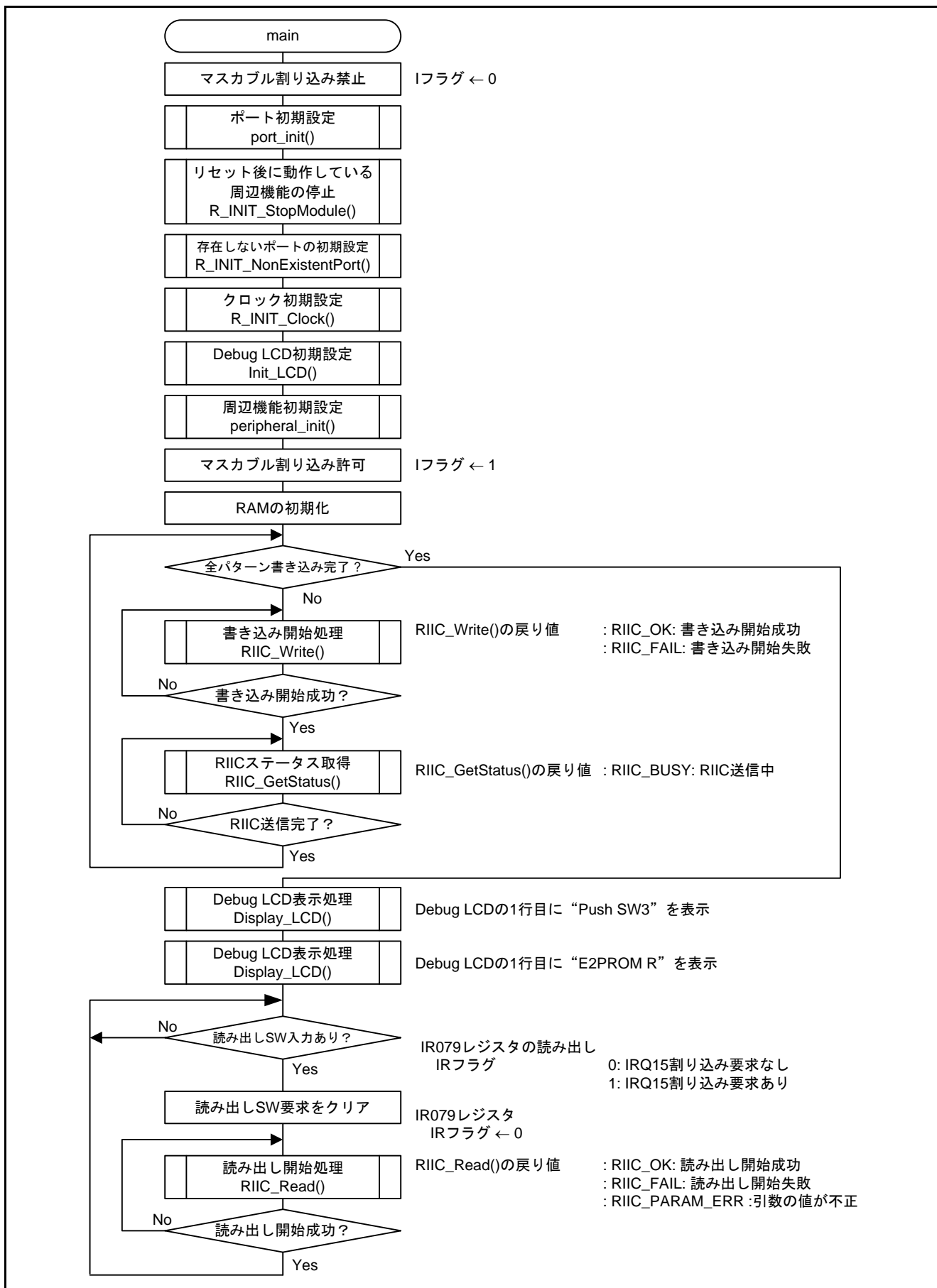


図5.3 メイン処理

5.8.2 ポート初期設定

図 5.4にポート初期設定のフローチャートを示します。

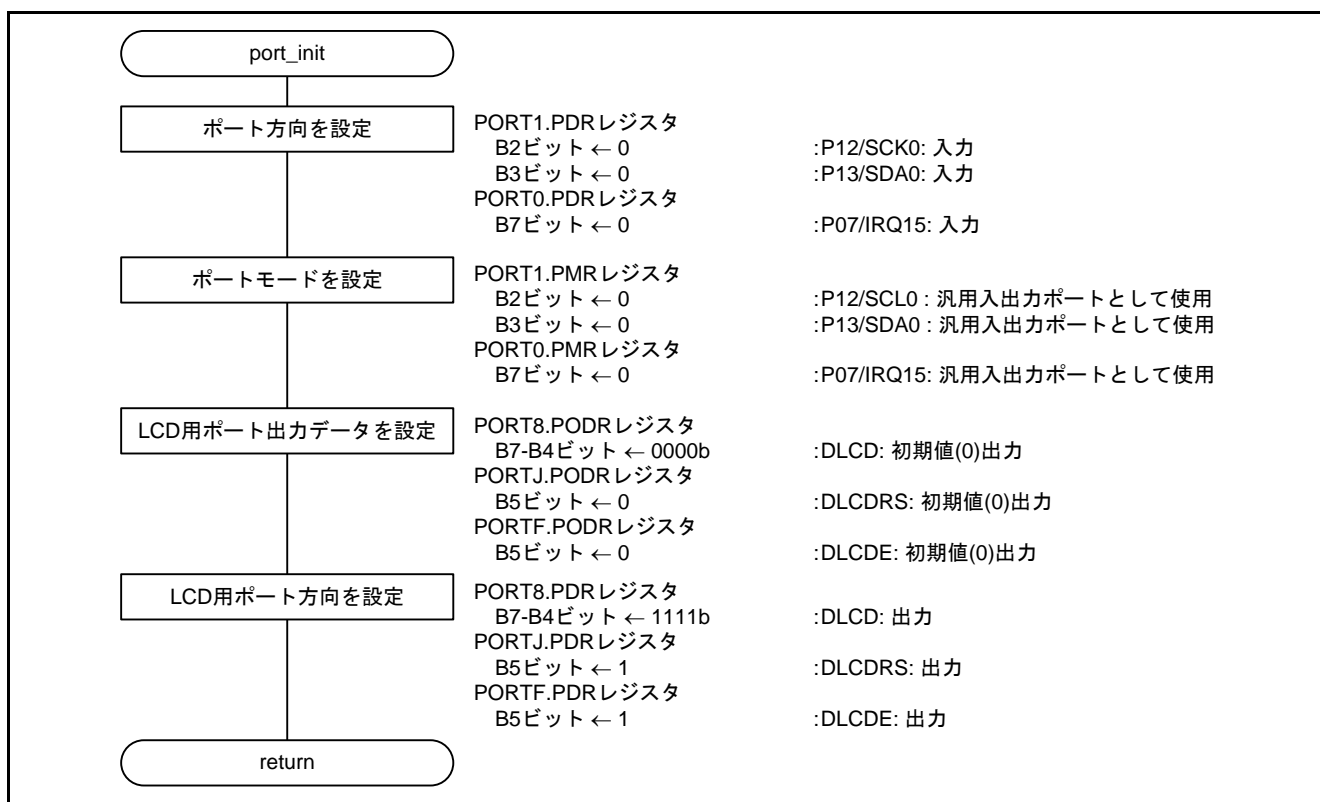


図5.4 ポート初期設定

5.8.3 周辺機能初期処理

図 5.5に周辺機能初期設定のフローチャートを示します。

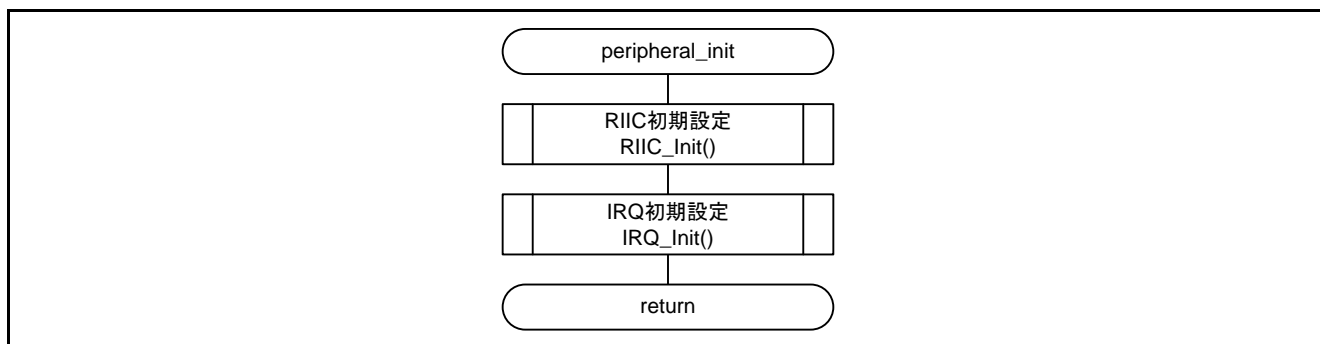


図5.5 周辺機能初期設定

5.8.4 IRQ 初期設定

図 5.6にIRQ 初期設定のフローチャートを示します。

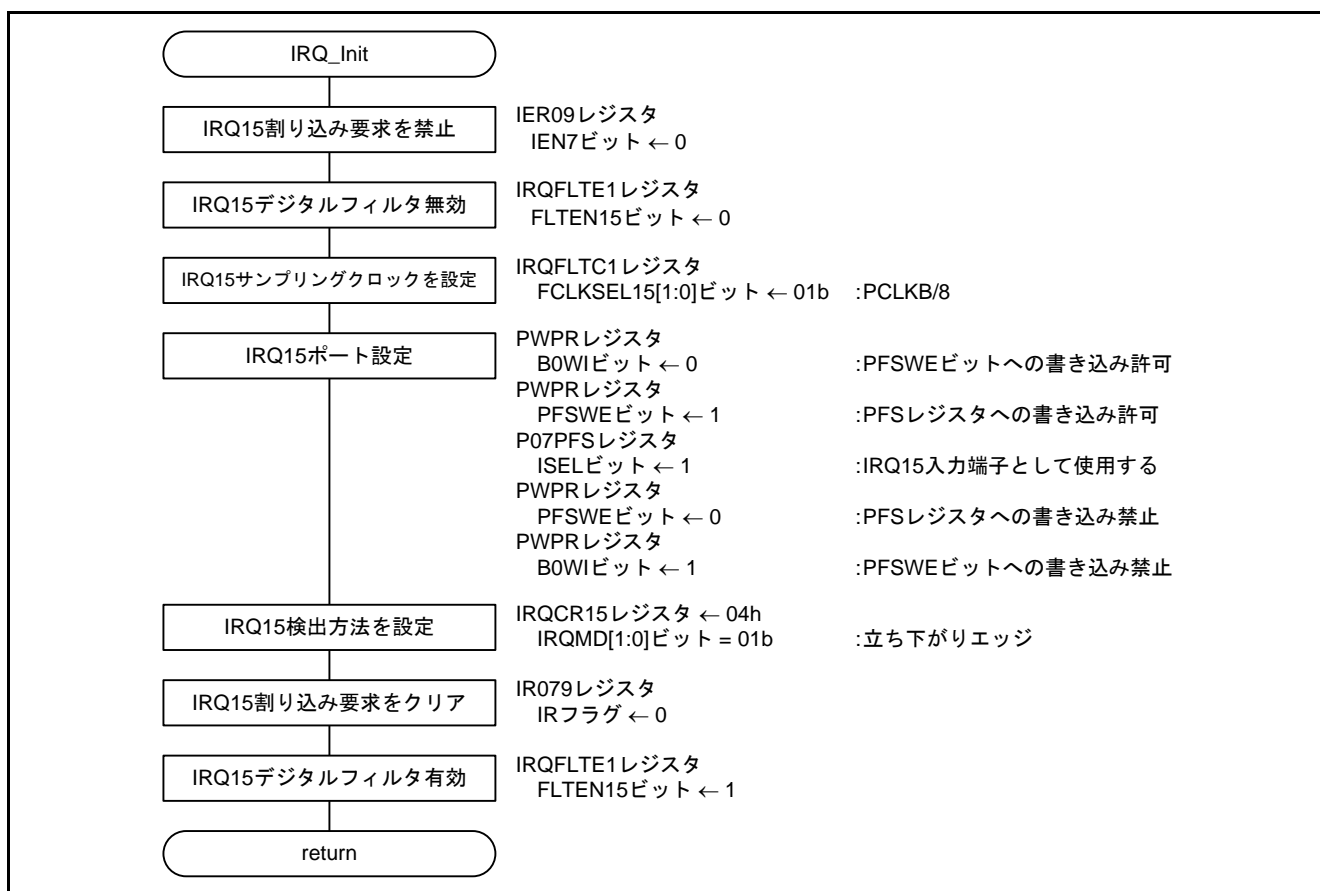


図5.6 IRQ 初期設定

5.8.5 EEPROM 書き込み後のコールバック関数

図 5.7にEEPROM 書き込み後のコールバック関数のフローチャートを示します。

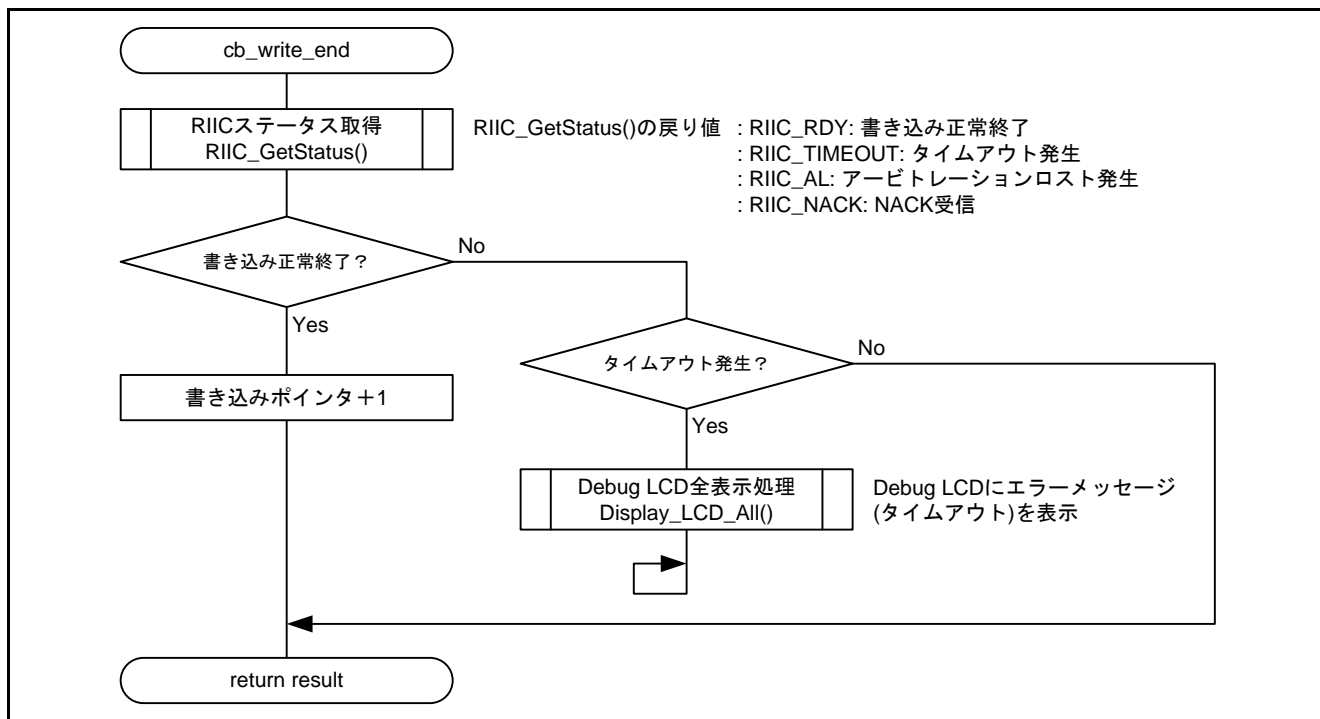


図5.7 EEPROM 書き込み後のコールバック関数

5.8.6 EEPROM 読み出し後のコールバック関数

図 5.8にEEPROM 読み出し後のコールバック関数のフローチャートを示します。

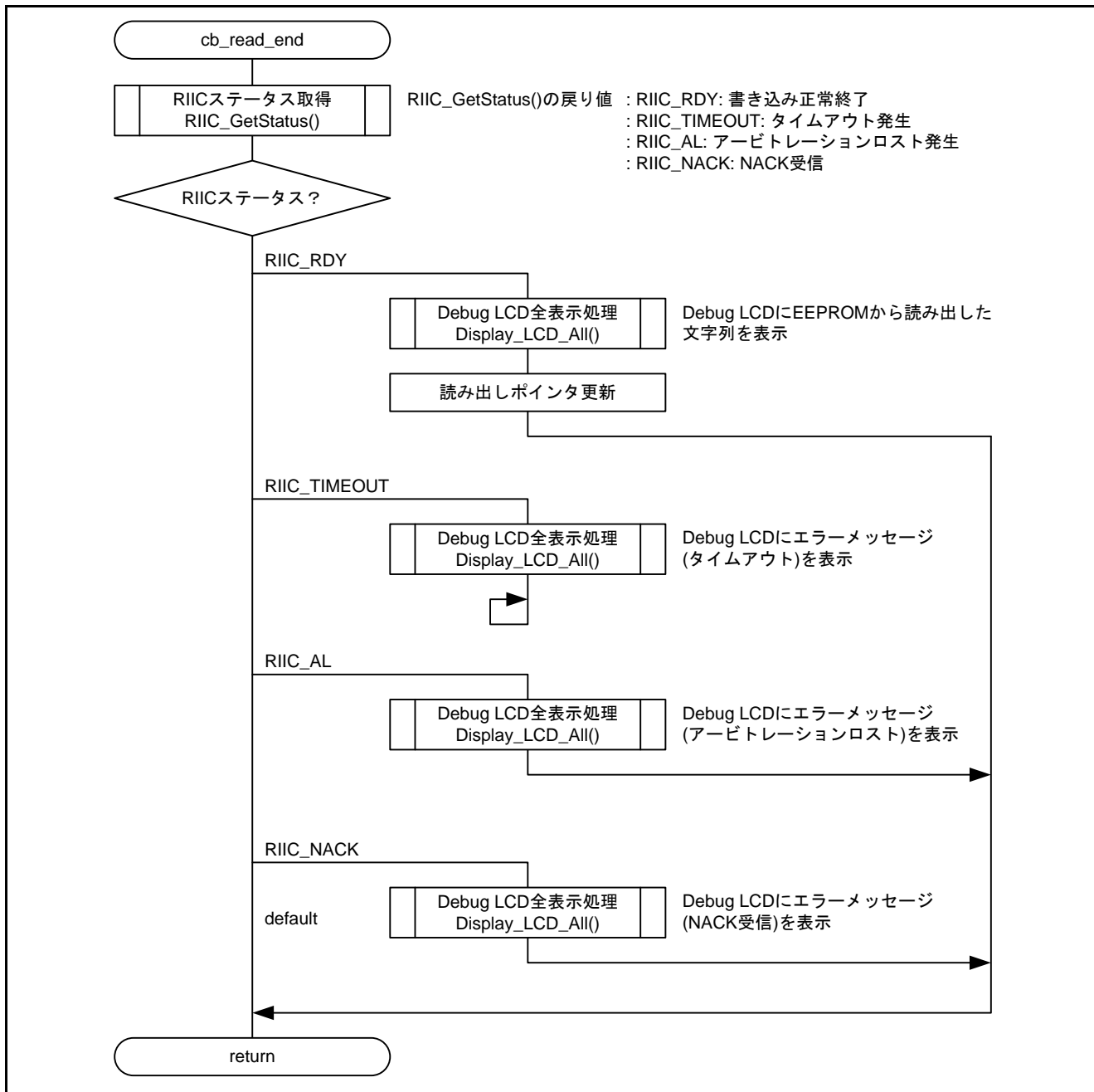


図5.8 EEPROM 読み出し後のコールバック関数

5.8.7 Debug LCD 全表示処理

図 5.9にDebug LCD 全表示処理のフローチャートを示します。

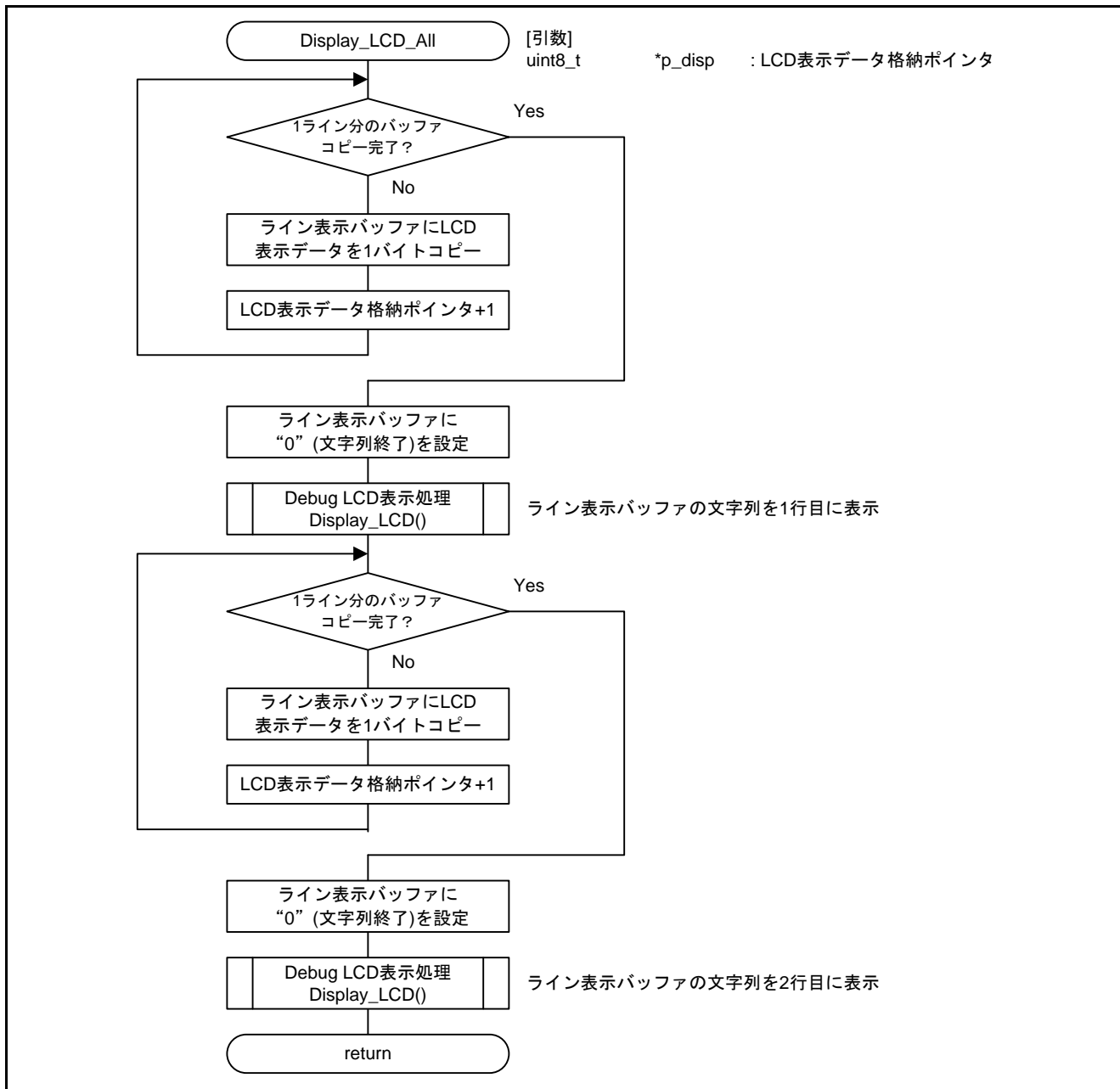


図5.9 Debug LCD 全表示処理

5.8.8 RIIC 初期設定

図 5.10、図 5.11にRIIC 初期設定のフローチャートを示します。



図5.10 RIIC 初期設定(1/2)



図5.11 RIIC 初期処理(2/2)

5.8.9 EEPROM への書き込み開始処理

図 5.12にEEPROM への書き込み開始処理のフローチャートを示します。

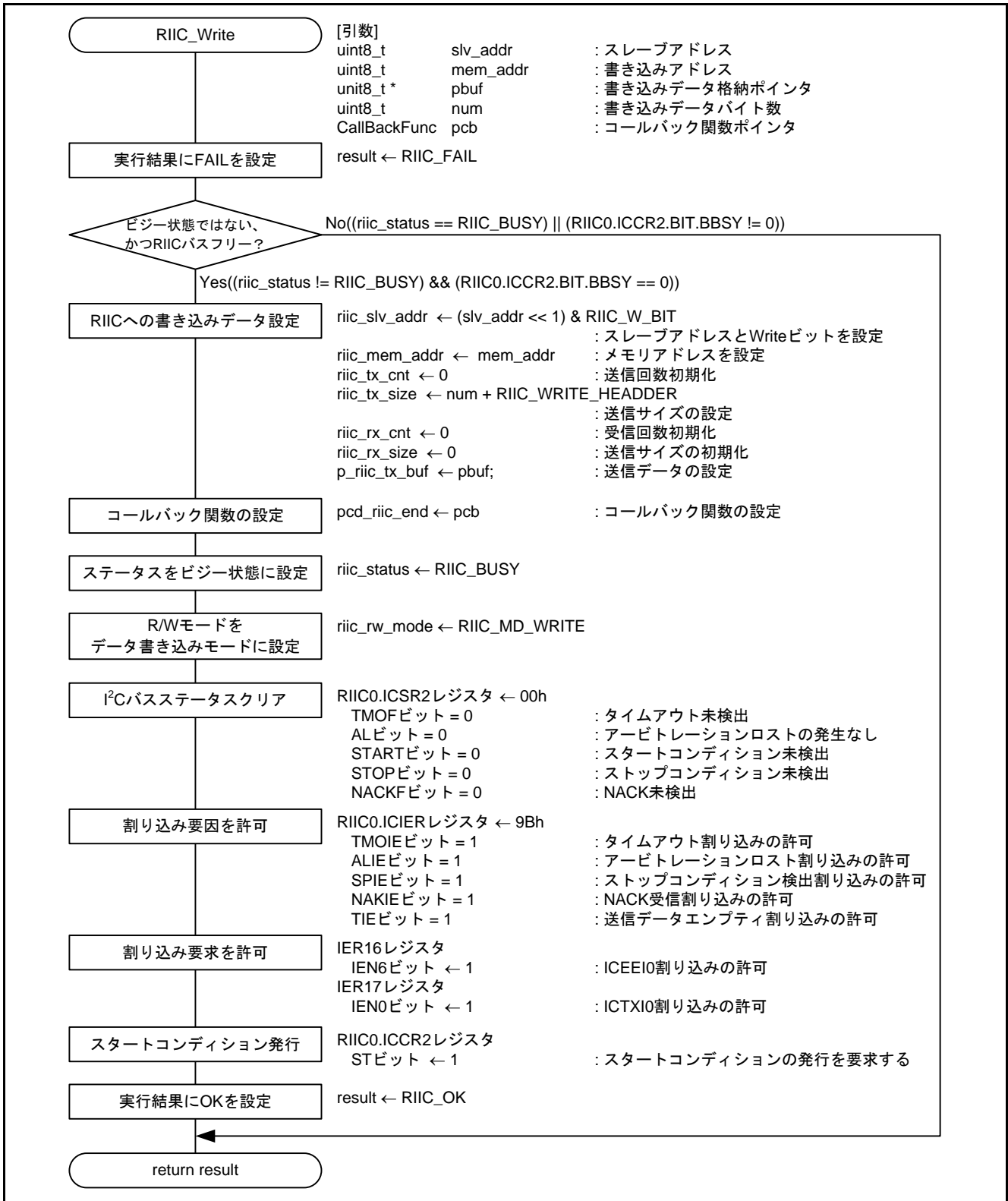


図5.12 EEPROM への書き込み開始処理

5.8.10 EEPROM からの読み出し開始処理

図 5.13にEEPROM からの読み出し開始処理のフローチャートを示します。

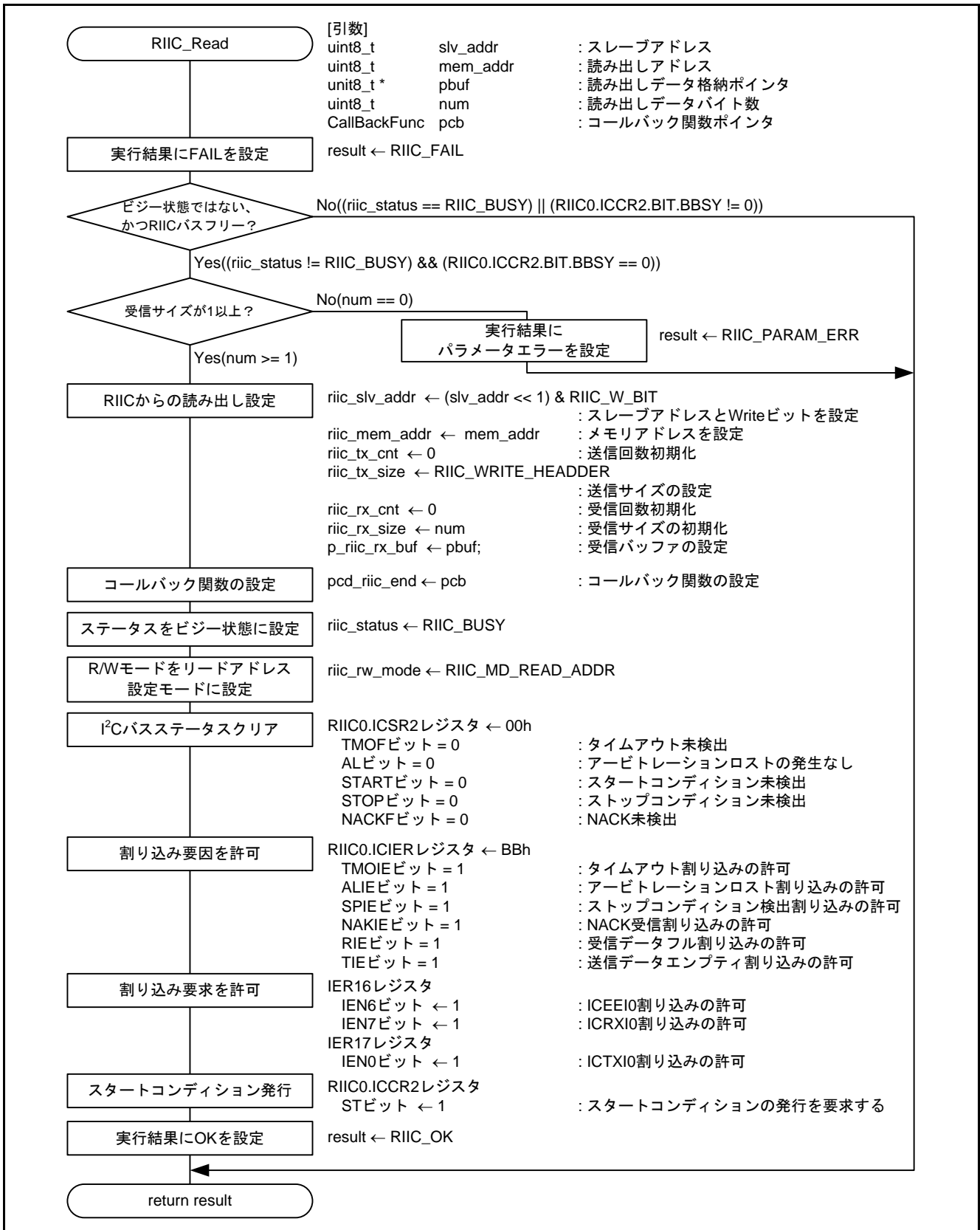


図5.13 EEPROM からの読み出し開始処理

5.8.11 RIIC0 ICTXIO 割り込み処理

図 5.14にRIIC0 ICTXIO 割り込み処理のフローチャートを示します。

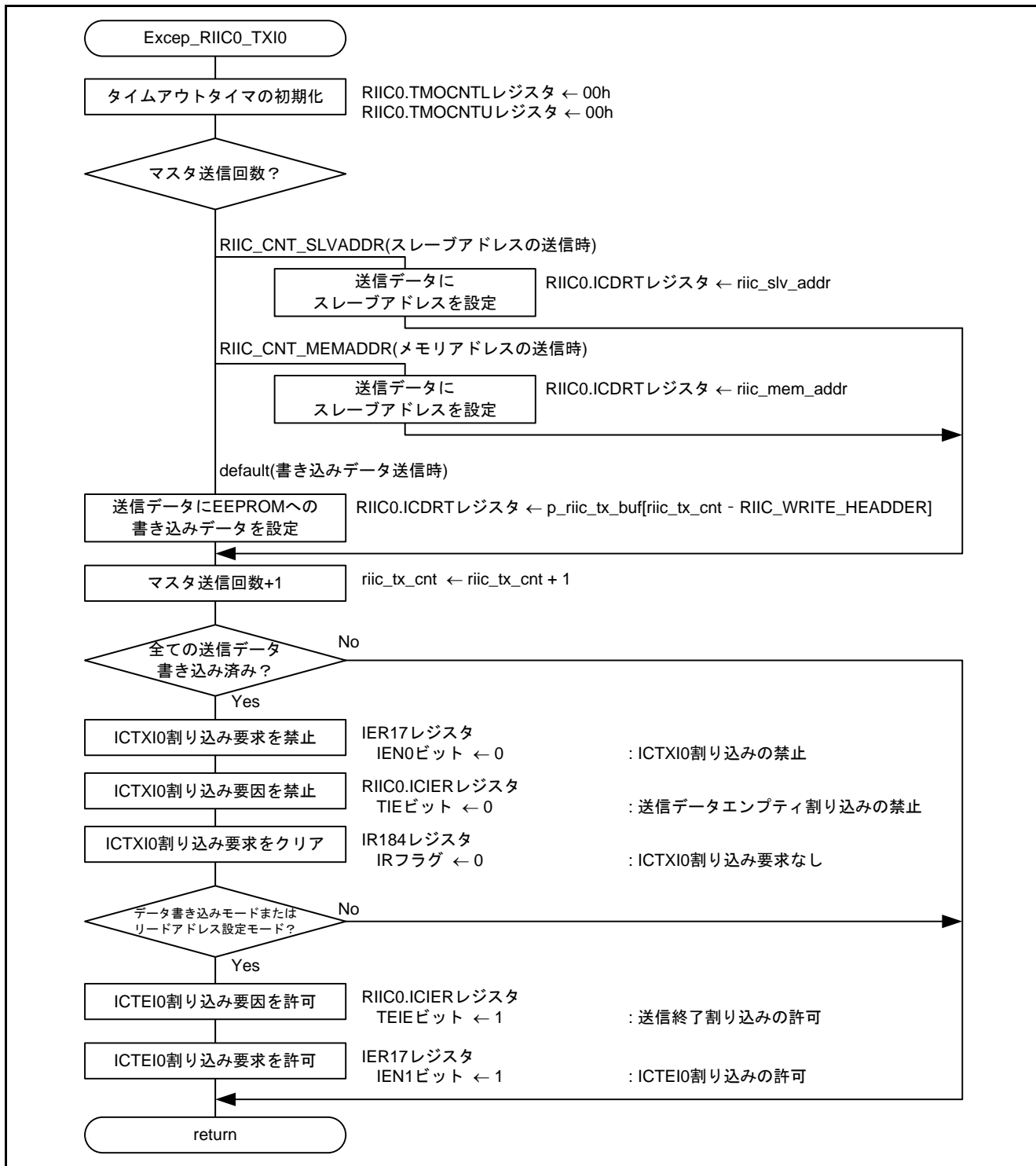


図5.14 RIIC0 ICTXIO 割り込み処理

5.8.12 RIIC0 ICRX10 割り込み処理

図 5.15にRIIC0 ICRX10 割り込み処理のフローチャートを示します。

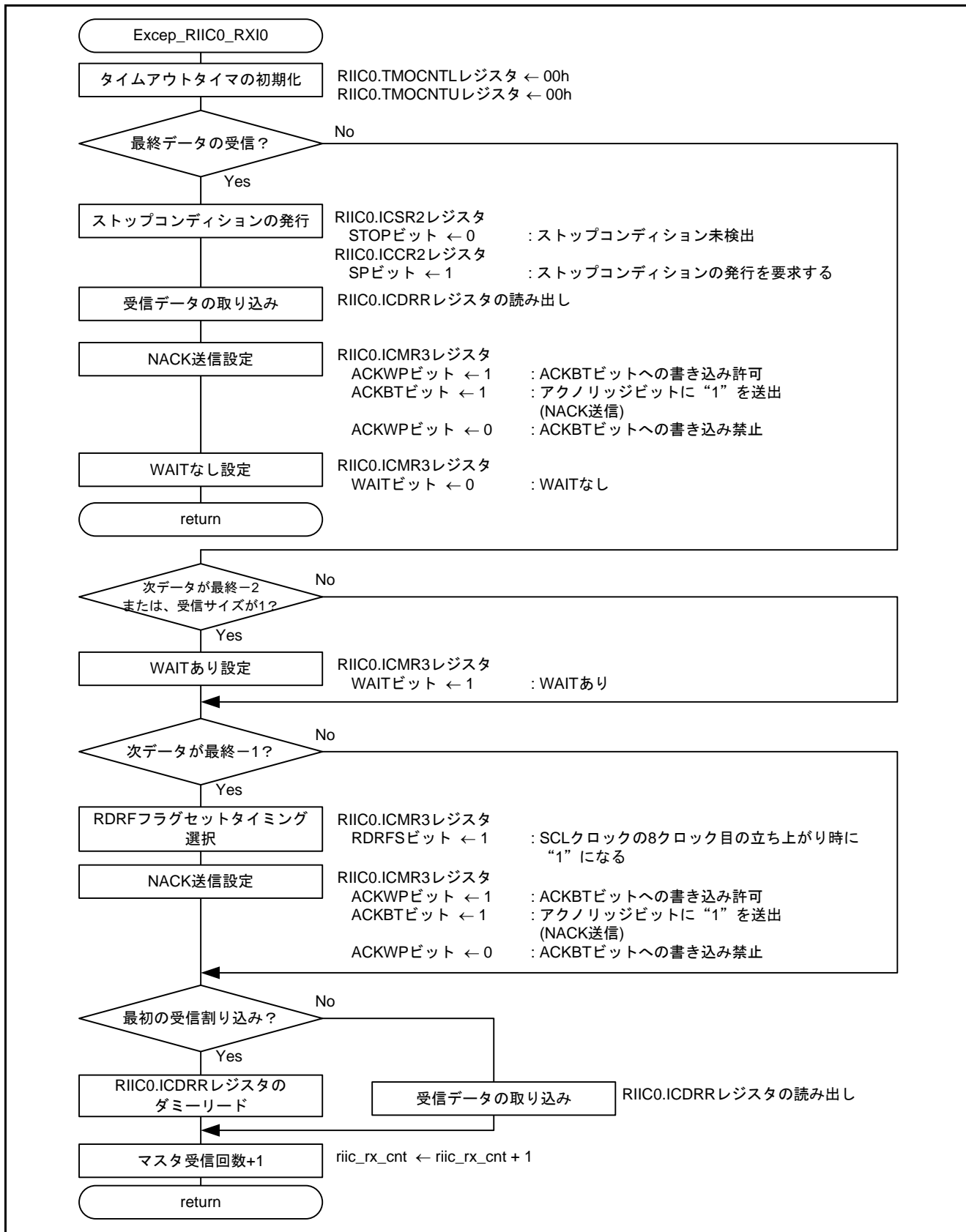


図5.15 RIIC0 ICRX10 割り込み処理

5.8.13 RIIC0 ICTEIO 割り込み処理

図 5.16にRIIC0 ICTEIO 割り込み処理のフローチャートを示します。

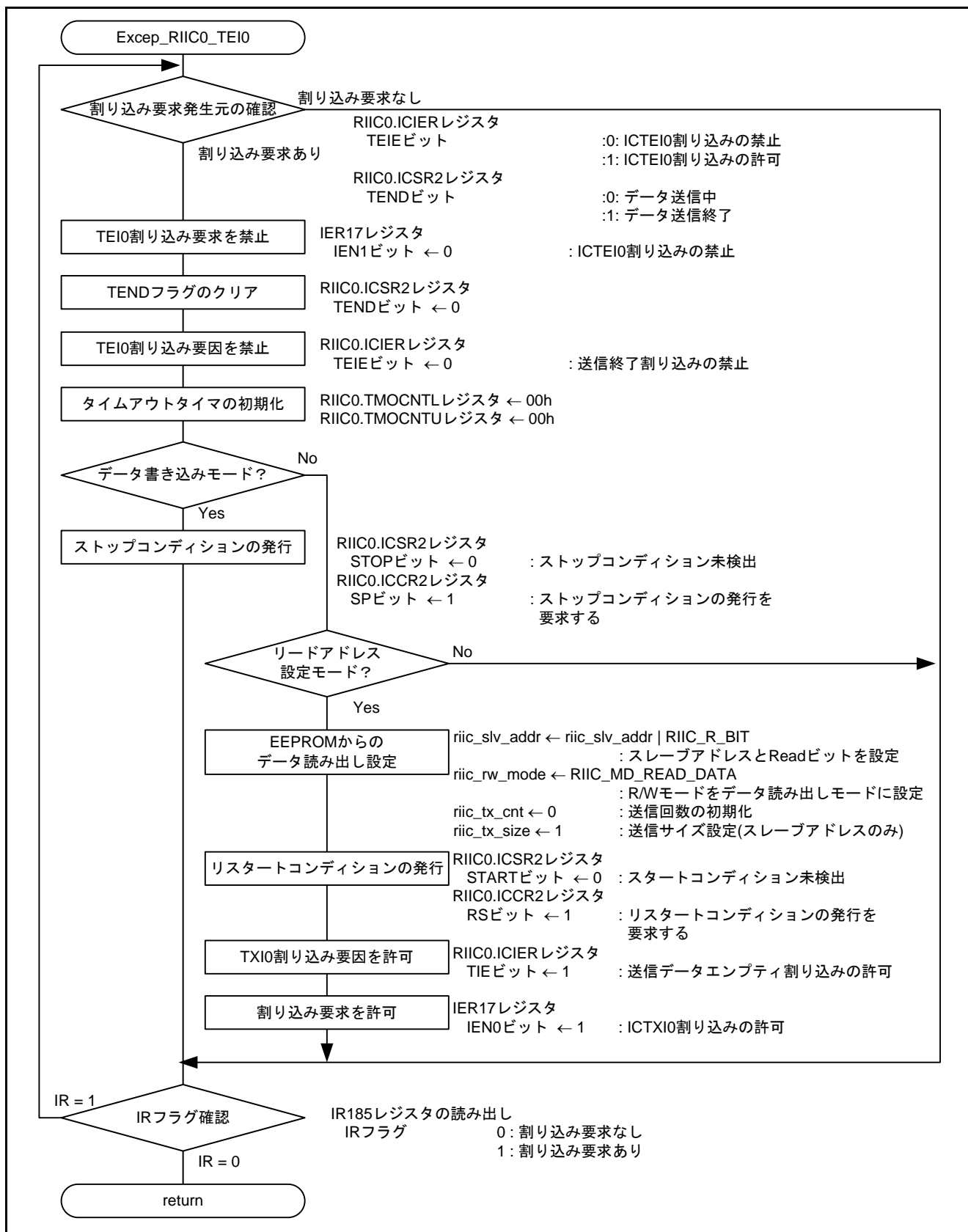


図5.16 RIIC0 ICTEIO 割り込み処理

5.8.14 RIIC0 ICEEIO 割り込み処理

図 5.17、図 5.18にRIIC0 ICEEIO 割り込み処理のフローチャートを示します。

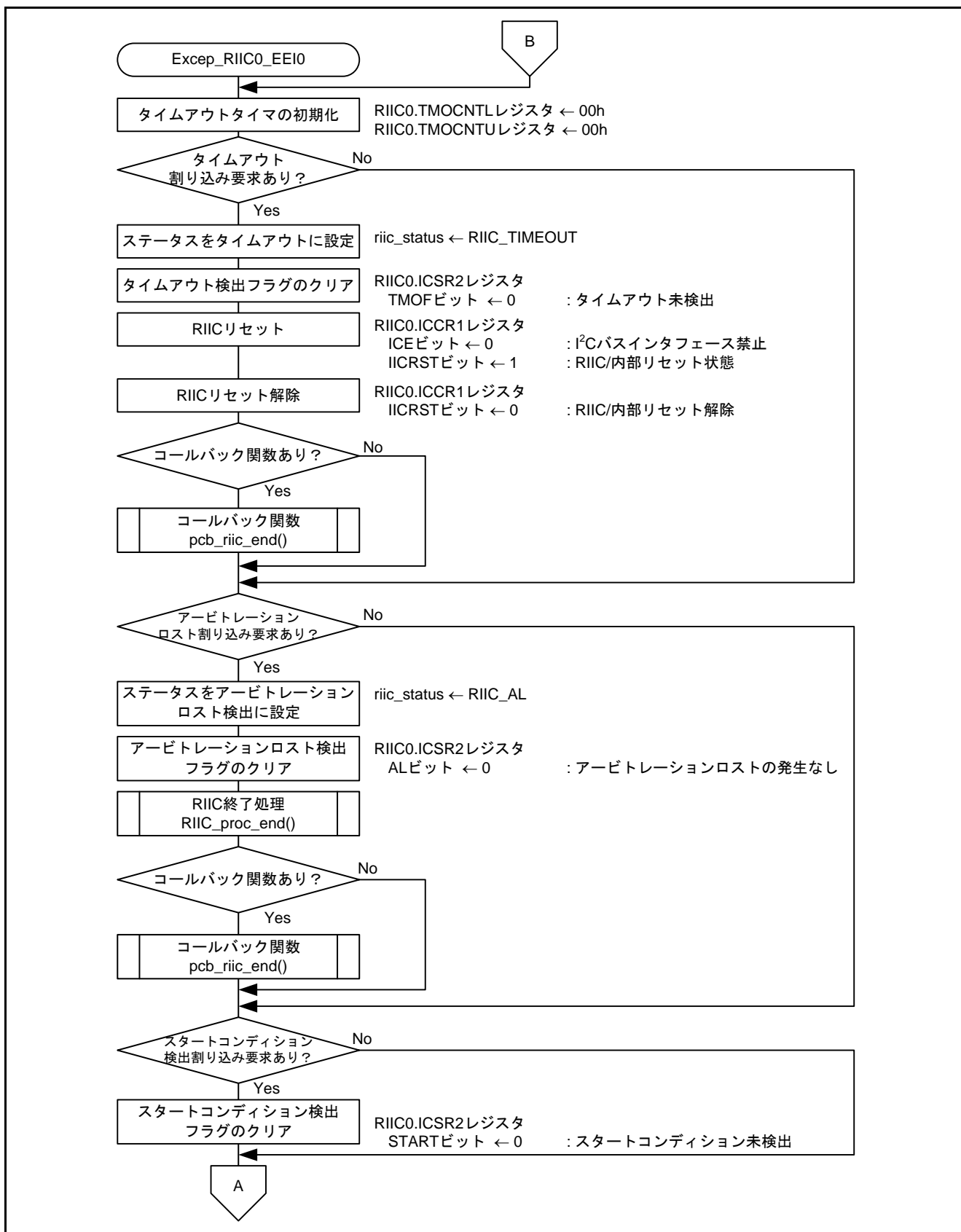


図5.17 RIIC0 ICEEIO 割り込み処理(1/2)

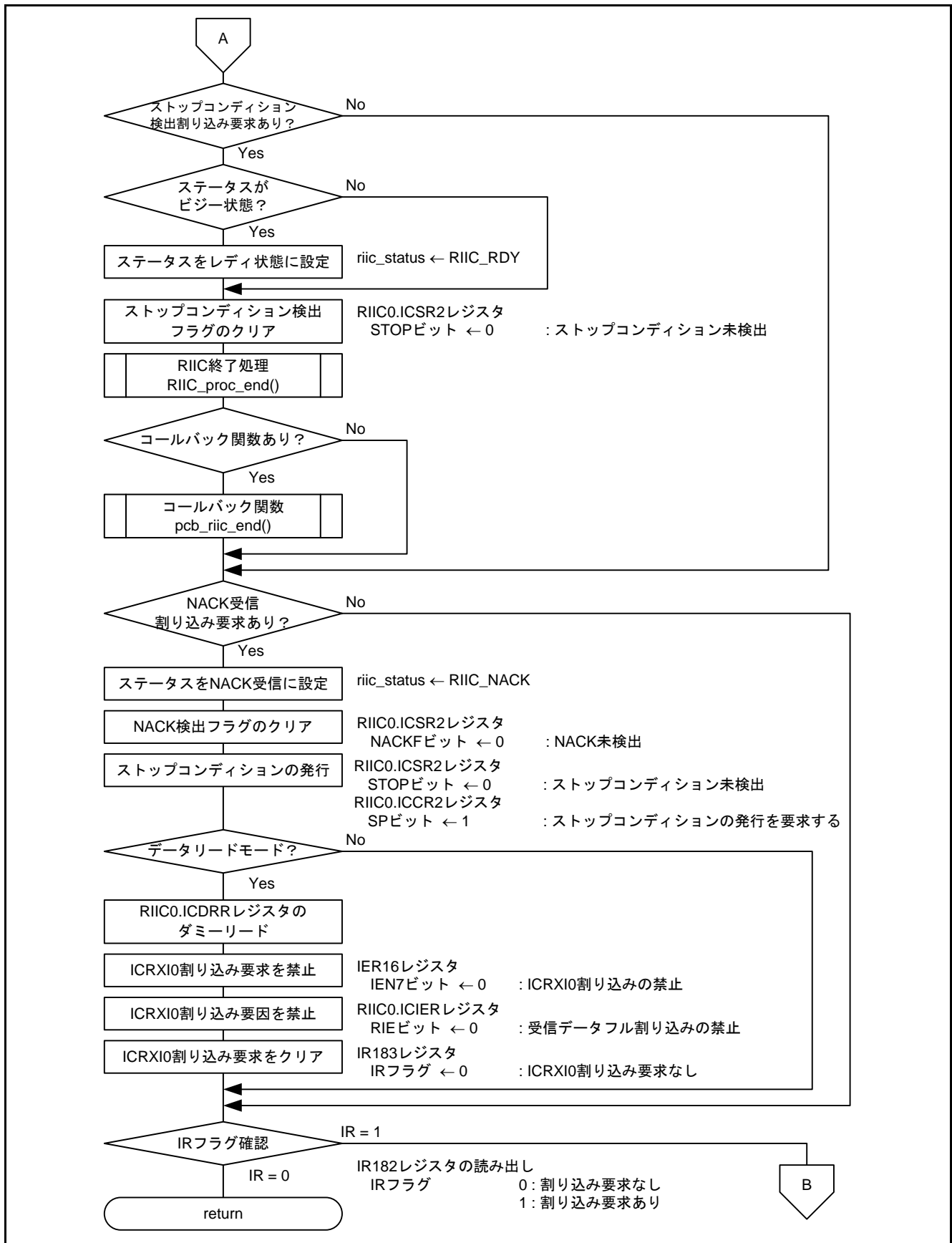


図5.18 RIIC0 ICCEI0 割り込み処理(2/2)

5.8.15 RIIC ステータス取得

図 5.19にRIIC ステータス取得のフローチャートを示します。

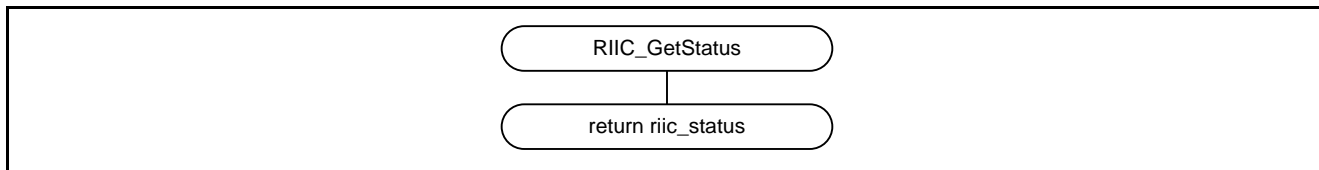


図5.19 RIIC ステータス取得

5.8.16 EEPROM への書き込み/読み出し終了時の処理

図 5.20にEEPROM への書き込み/読み出し終了時の処理のフローチャートを示します。

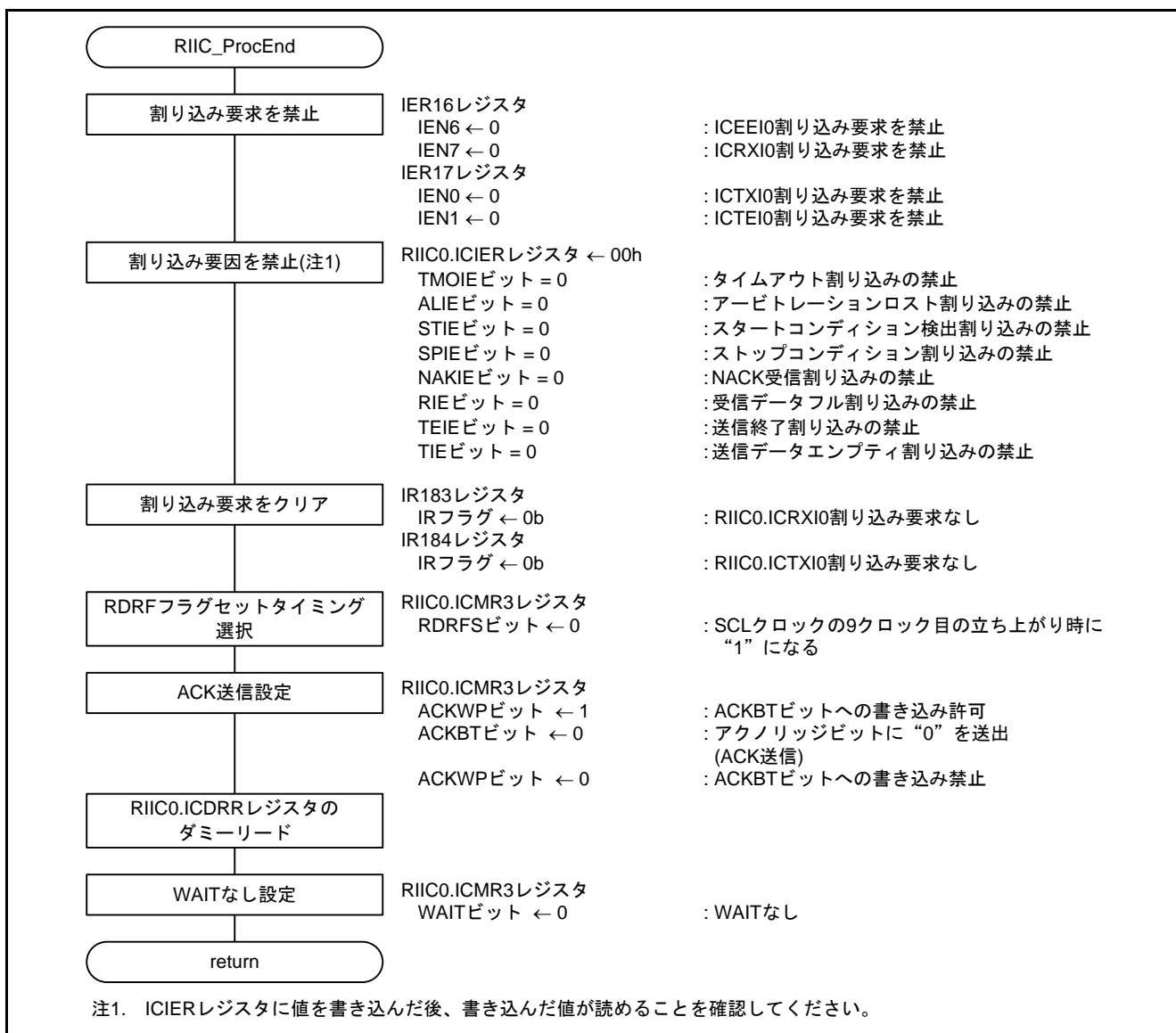


図5.20 EEPROM への書き込み/読み出し終了時の処理

6. 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。

7. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX63N グループ、RX631グループ ユーザーズマニュアル ハードウェア編 Rev.1.70 (R01UH0041JJ)
(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

ユーザーズマニュアル：開発環境

RX ファミリー C/C++コンパイラパッケージ V.1.01 ユーザーズマニュアル Rev.1.00 (R20UT0570JJ)
(最新版をルネサス エレクトロニクスホームページから入手してください。)

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com>

お問合せ先

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

改訂記録	RIIC を使用した EEPROM 制御
------	----------------------

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2014.04.01	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、
家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、
防災・防犯装置、各種安全装置等
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問い合わせください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍用用途に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/contact/>