RL78/G22

シリアル・インターフェース IICA (マスタ送受信)

要旨

本アプリケーションノートでは、シリアル・インタフェース IICA によるマスタ送受信機能の使用方法を説明します。IICA を使用してシングルマスタ･システムでのマスタ動作（アドレス送信、データ送受信）を行います。

対象デバイス

RL78/G22

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
1.1 IIC 通信のタイミング・チャート 4
1.2 シリアル RAM の制御 18
1.2.1 コマンド設定 18
1.2.2 連続データ書き込み 19
1.2.3 連続データ読み込み 20

2. 動作確認条件 21

3. 関連アプリケーションノート 21

4. ハードウェア説明 22
4.1 ハードウェア構成例 22
4.2 使用端子一覧 22

5. ソフトウェア説明 23
5.1 動作概要 23
5.2 オプション・バイトの設定一覧 24
5.3 定数一覧 24
5.4 変数一覧 25
5.5 関数一覧 26
5.6 関数仕様 27
5.7 フローチャート 31
5.7.1 メイン処理 31
5.7.2 メイン・ユーザ初期化設定 34
5.7.3 スイッチの押下待ち処理 34
5.7.4 ms 単位の時間待ち処理 35
5.7.5 タイマ・アレイ・ユニット チャネル 0 割り込み処理 36
5.7.6 スレーブへのデータ送信処理 37
5.7.7 スレーブからのデータ受信処理 39
5.7.8 IICA0 マスタ送信起動関数 41
5.7.9 IICA0 マスタ受信起動関数 42
5.7.10 IICA0 通信状況確認関数 42
5.7.11 通信完了待ち関数 43
5.7.12 ストップ・コンディション発行処理 44
5.7.13 バス状態確認関数 45
5.7.14 IICA0 マスタ通信処理 46
5.7.15 us 単位での時間待ち処理 49
5.7.16 us 単位での待ち時間設定処理 50
5.7.17 タイマ・アレイ・ユニット チャネル 1 割り込み処理 51
5.7.18 IICA0 割り込み処理 52

6. サンプルコード 53

7. 参考ドキュメント 53

1. 仕様

本アプリケーションノートでは、シリアル・インタフェース IICA を使用してシングルマスタ構成でのマスタ送受信（アドレス送信、データ送受信）を行う方法を説明します。スレーブにはスレーブ内のアドレスを指定するレジスタが搭載されていることを想定しています。
マスタは、スレーブ・アドレスを指定してスレーブを選択します。スレーブとの通信が成立すると、データの送受信ができるようになります。

本アプリケーションノートで対象とするスレーブの概略仕様を以下に示します。

- スレーブ・アドレス：0b1010000 注
- 指定したアドレスから、任意のバイト数のデータの読み出しや書き込みが可能です。
- スレーブのシリアル RAM 領域は、レジスタ・アドレス 0x80～0xFF（128 バイト）です。レジスタ・アドレス 0x00 に配置されたコマンド・レジスタでスレーブの動作を指定します。
- レジスタ・アドレスとして 0x01～0x7F が指定された場合、NACK 応答して通信から退避します。
- レジスタ・アドレスが 0xFF を越えた場合は、シリアル RAM 領域を選択し、指定したレジスタ・アドレスの下位 7 ビットのみを有効として扱います。

注　RL78 ファミリでは、自局アドレス（7 ビット）を SVA0 レジスタの上位 7 ビットで表現します。SVA0 レジスタの最下位ビットは 0 固定です。アドレス送信は、スレーブのアドレスと転送方向（R / W）を合わせて 8 ビットとして IICA シフト・レジスタ 0（IICA0）に書き込みます。

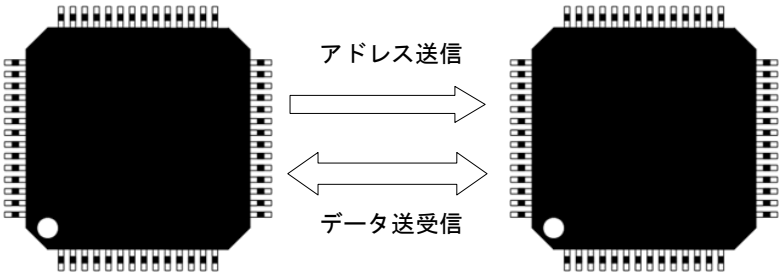
注意　本サンプルコードは、RL78/G22 シリアル・インタフェース IICA（スレーブ送受信）（R01AN7345J）アプリケーションノートに対応しています。

使用する周辺機能と用途を表 1.1 に示し、IIC 通信の概要を図 1.1 に示します。IIC 通信のタイミング・チャートを図 1.2 から図 1.8 に示します。

表 1.1　使用する周辺機能と用途

周辺機能	用途
シリアル・インタフェース IICA	シングルマスタ・システムでの IIC 通信を行う（SCLA0 端子と SDAA0 端子を使用）。

図 1.1 IIC 通信の概要

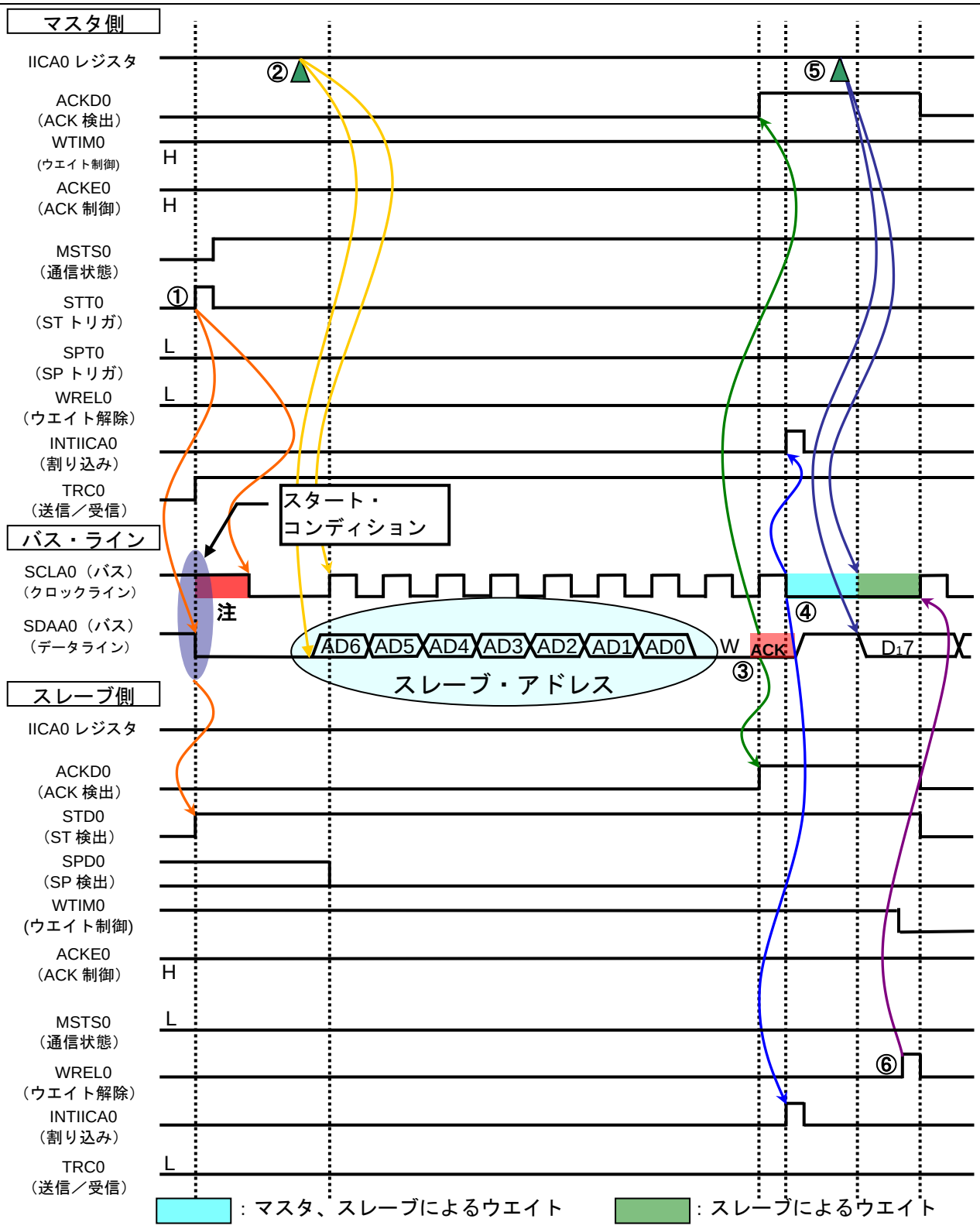


アドレス送信：IIC バスに接続された各デバイスはそれぞれ固有のアドレスを持っています。マスタは転送対象のスレーブ・アドレスを送信します。アドレス（7ビット）と転送方向（1ビット）を1バイトのデータとして送信します。スレーブは1バイトのデータ受信後、アクノリッジを生成します。
データ送受信：アドレス受信後、マスタへデータ送信、またはマスタからデータ受信する動作です。

1.1 IIC 通信のタイミング・チャート

(1) マスタ→スレーブ通信 1（スタート・コンディション～アドレス～データ）

図 1.2 IIC 通信タイミング・チャート（マスタ→スレーブ通信例）(1/4)



注 SDAA0 が立ち下がってから SCLA0 が立ち下がるまで、標準モード設定時 4.0μs 以上／ファースト・モード設定時 0.6μs 以上です。

① マスタ側でスタート・コンディション・トリガがセット（STT0=1）されると、SDAA0 ラインが立ち下がり、スタート・コンディションが生成されます。その後、スタート・コンディションを検出（STD0=1）すると、マスタ側はマスタ通信状態（MSTS0=1）になります。ホールド時間経過後、SCLA0 ラインが立ち下がり、通信準備が完了となります。

② マスタ側で IICA0 レジスタにアドレス＋転送方向 W（送信）が書き込まれると、スレーブ・アドレスが送信されます。

③ 受信したアドレスとスレーブのアドレスが一致した場合[注]、スレーブ側のハードウェアより ACK0 がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

④ 9 クロック目の立ち下がりで、マスタ側の割り込み（INTIICA0:アドレス送信完了の割り込み）が発生し、スレーブはアドレスが一致した場合、割り込み（INTIICA0:アドレス一致の割り込み）が発生します。さらに、マスタ側およびアドレスが一致したスレーブ側はウエイト（SCLA0 ライン：ロウ・レベル）を発生させます。[注]。

⑤ マスタ側が IICA0 レジスタに送信データを書き込み、マスタ側はウエイトを解除します。

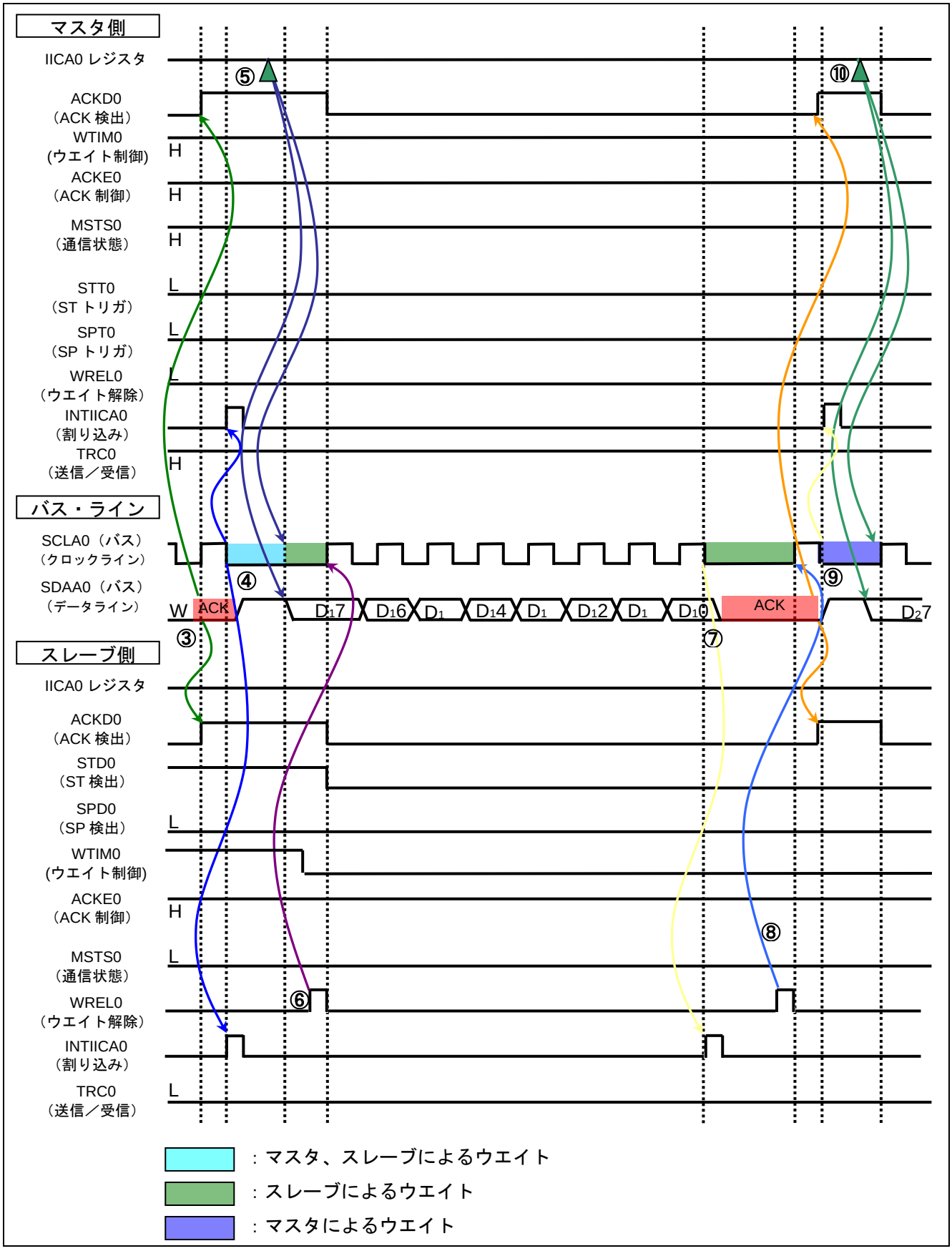
⑥ スレーブ側は受信動作となるため、8 クロック・ウエイト（WTIM0=0）を選択します。スレーブ側がウエイトを解除（WREL0=1）すると、マスタ側からスレーブ側にデータ転送が開始されます。

注　受信アドレスと自局アドレスが不一致の場合は、スレーブ側は ACK をマスタ側へ返しません（NACK）。また、スレーブ側の INTIICA0 割り込み（アドレス一致の割り込み）は発生せず、スレーブ側からのウエイトは入りません。ただし、マスタ側は ACK、NACK のどちらの場合でも、INTIICA0 割り込み（アドレス送信完了の割り込み）が発生します。

RL78 ファミリでは、自局アドレス（7 ビット）を SVA0 レジスタの上位 7 ビットで表現します。SVA0 レジスタの最下位ビットは 0 固定です。アドレス送信は、スレーブのアドレスと転送方向（R / W）を合わせて 8 ビットとして IICA シフト・レジスタ 0（IICA0）に書き込みます。

（2）マスタ→スレーブ通信 2（アドレス～データ～データ）

図 1.3 IIC 通信タイミング・チャート（マスタ→スレーブ通信例）(2/4)



③ 受信したアドレスとスレーブのアドレスが一致した場合、スレーブ側のハードウェアより ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

④ 9 クロック目の立ち下がりで、マスタ側の割り込み（INTIICA0:アドレス送信完了の割り込み）が発生し、スレーブはアドレスが一致した場合、割り込み（INTIICA0:アドレス一致の割り込み）が発生します。さらに、マスタ側およびアドレスが一致したスレーブ側はウエイト（SCLA0 ライン：ロウ・レベル）を発生させます。

⑤ マスタ側が IICA0 レジスタに送信データを書き込み、マスタ側はウエイトを解除します。

⑥ スレーブ側は受信動作となるため、8 クロック・ウエイト（WTIM0=0）を選択します。スレーブ側がウエイトを解除（WREL0=1）すると、マスタ側からスレーブ側にデータ転送が開始されます。

⑦ データ転送完了後、8 クロック目の立ち下がり時にスレーブ側のハードウェアによるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側で割り込み（INTIICA0:転送完了割り込み）が発生します。

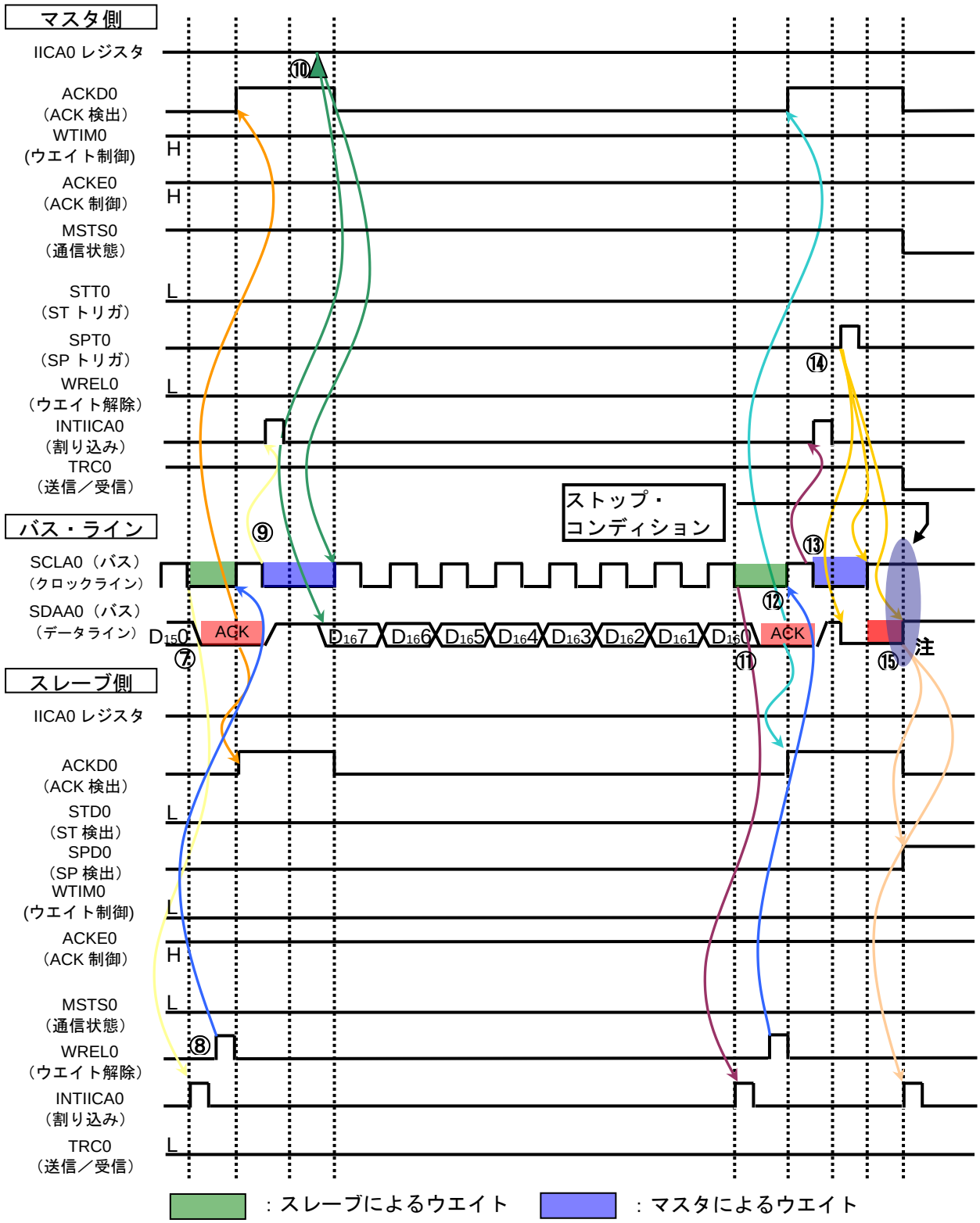
⑧ スレーブ側が受信データを読み出して、ウエイトを解除（WREL0=1）すると、ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

⑨ 9 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側で割り込み（INTIICA0:転送完了割り込み）が発生します。

⑩ マスタ側が IICA0 レジスタに送信データを書き込み、マスタ側がウエイトを解除すると、マスタ側からスレーブ側にデータ転送が開始されます。

（3）マスタ→スレーブ通信 3（データ～データ～ストップ・コンディション）

図 1.4 IIC 通信タイミング・チャート（マスタ→スレーブ通信例）(3/4)



注 ストップ・コンディション発行後、SCLA0 が立ち上がってからストップ・コンディションが生成されるまでに標準モード設定時 4.0μs 以上／ファースト・モード設定時 0.6μs 以上です。

⑦ データ転送完了後、8 クロック目の立ち下がり時にスレーブ側のハードウェアよるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側で割り込み（INTIICA0:転送完了割り込み）が発生します。

⑧ スレーブ側が受信データを読み出して、ウエイトを解除（WREL0=1）すると、ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

⑨ 9 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側の割り込み（INTIICA0:アドレス送信完了の割り込み）が発生します。

⑩ マスタ側が IICA0 レジスタに送信データを書き込み、マスタ側はウエイトを解除すると、マスタ側からスレーブ側にデータ転送が開始されます。

⑪ データ転送完了後、8 クロック目の立ち下がり時にスレーブ側のハードウェアよるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側で割り込み（INTIICA0:転送完了割り込み）が発生します。

⑫ スレーブ側が受信データを読み出して、ウエイトを解除（WREL0=1）すると、ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

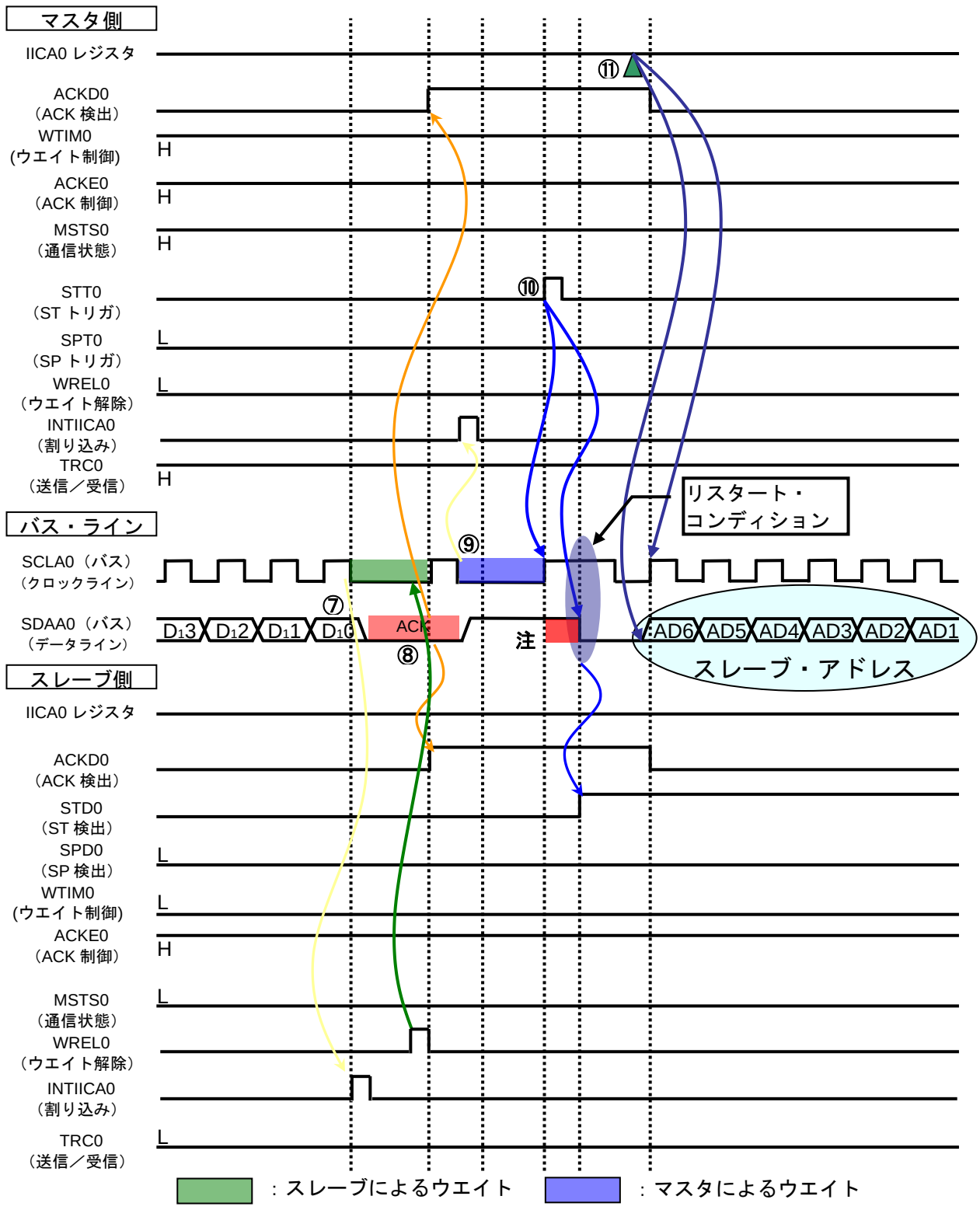
⑬ 9 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側で割り込み（INTIICA0:転送完了の割り込み）が発生します。

⑭ ストップ・コンディション・トリガをセット（SPT0=1）すると、SDAA0 ラインを立ち下げ、SCLA0 ラインを立ち上げ、ストップ・コンディションセットアップ時間経過後、SDAA0 ラインを立ち上げることでストップ・コンディションが生成されます。

⑮ ストップ・コンディションが生成されると、スレーブ側でストップ・コンディションが検出され（SPD0=1）、スレーブ側で IICA0 割り込み（ストップ・コンディションの割り込み）が発生します。

（4）マスタ→スレーブ通信 4（データ～リスタート・コンディション～アドレス）

図 1.5 IIC 通信タイミング・チャート（マスタ→スレーブ通信例）(4/4)



注 リスタート・コンディション発行後、SCLA0 が立ち上がってからスタート・コンディションが生成されるまでに標準モード設定時 4.7μs 以上／ファースト・モード設定時 0.6μs 以上です。

⑦ データ転送完了後、8 クロック目の立ち下がり時にスレーブ側のハードウェアよるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側で割り込み（INTIICA0:転送完了割り込み）が発生します。

⑧ スレーブ側が受信データを読み出して、ウエイトを解除（WREL0=1）すると、ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

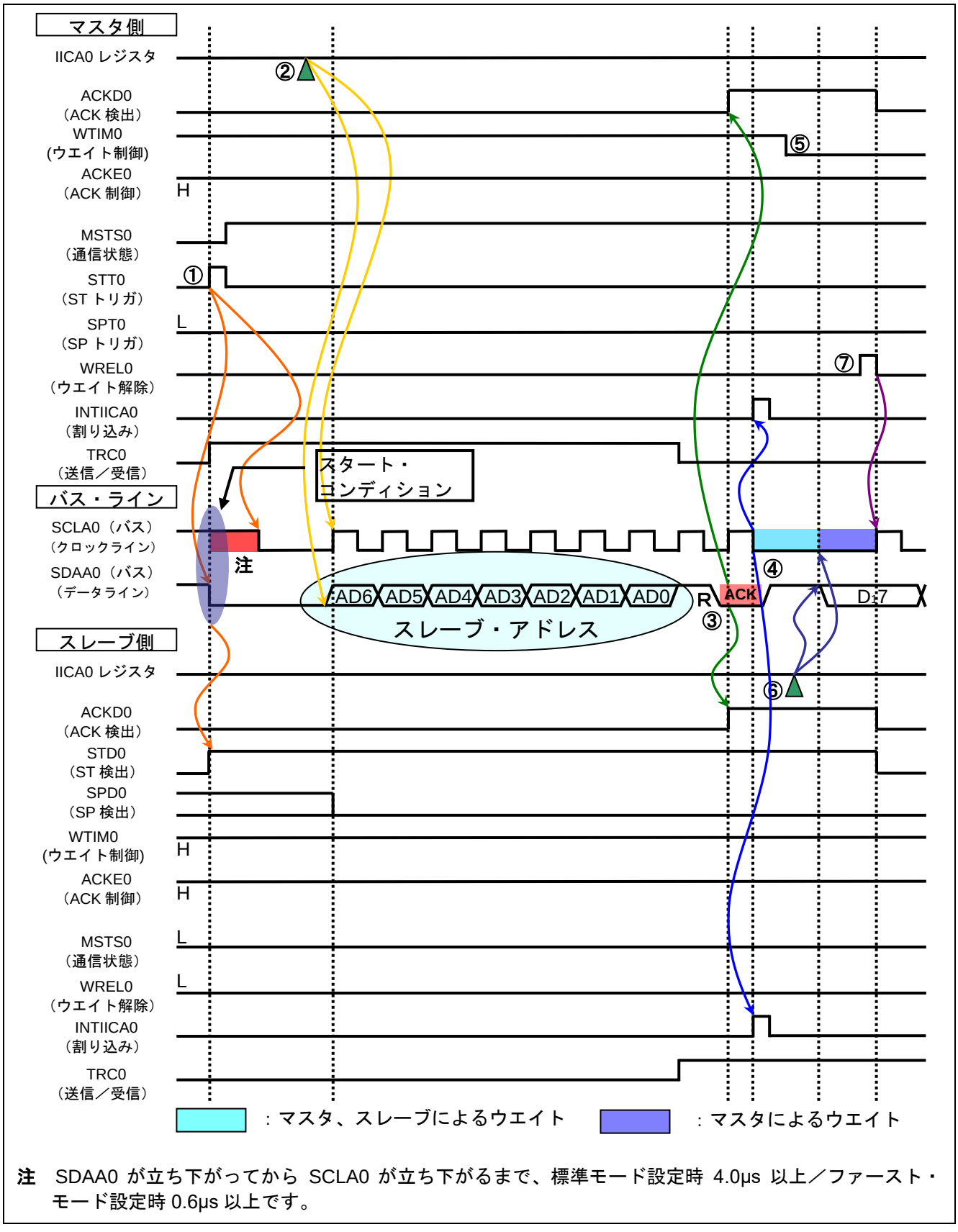
⑨ 9 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側で割り込み（INTIICA0:転送完了の割り込み）が発生します。

⑩ マスタ側で再度スタート・コンディション・トリガがセット（STT0=1）されると、SCLA0 ラインが立ち上がり、リスタート・コンディションセットアップ時間後、SDAA0 ラインが立ち下がり、スタート・コンディションが生成されます。その後、スタート・コンディションを検出（STD0=1）すると、ホールド時間経過後、バス・クロックラインが立ち下がり、通信準備が完了となります。

⑪ マスタ側が IICA0 レジスタにスレーブ･アドレスを書き込み、マスタ側からスレーブ側にアドレス転送が開始されます。

（5）スレーブ→マスタ通信 1（スタート・コンディション～アドレス～データ）

図 1.6 IIC 通信タイミング・チャート（スレーブ→マスタ通信例）(1/3)



注 SDAA0 が立ち下がってから SCLA0 が立ち下がるまで、標準モード設定時 4.0μs 以上／ファースト・モード設定時 0.6μs 以上です。

① マスタ側でスタート・コンディション・トリガがセット（STT0=1）されると、SDAA0 ラインが立ち下がり、スタート・コンディションが生成されます。その後、スタート・コンディションを検出（STD0=1）すると、マスタ側はマスタ通信状態（MSTS0=1）になります。ホールド時間経過後、SCLA0 ラインが立ち下がり、通信準備が完了となります。

② マスタ側で IICA0 レジスタにアドレス＋転送方向 R（受信）が書き込まれると、スレーブ・アドレスが送信されます。

③ 受信したアドレスとスレーブのアドレスが一致した場合[注]、スレーブ側のハードウェアより ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

④ 9 クロック目の立ち下がりで、マスタ側の INTIICA0 割り込み（アドレス送信完了の割り込み）が発生し、スレーブはアドレスが一致した場合、INTIICA0 割り込み（アドレス一致の割り込み）が発生します。さらに、マスタ側およびアドレスが一致したスレーブ側はウエイト（SCLA0 ライン：ロウ・レベル）を発生させます。

⑤ マスタ側は受信動作となるため、8 クロック・ウエイト（WTIM0=0）を選択します。

⑥ スレーブ側が IICA0 レジスタに送信データを書き込み、スレーブ側はウエイトを解除します。

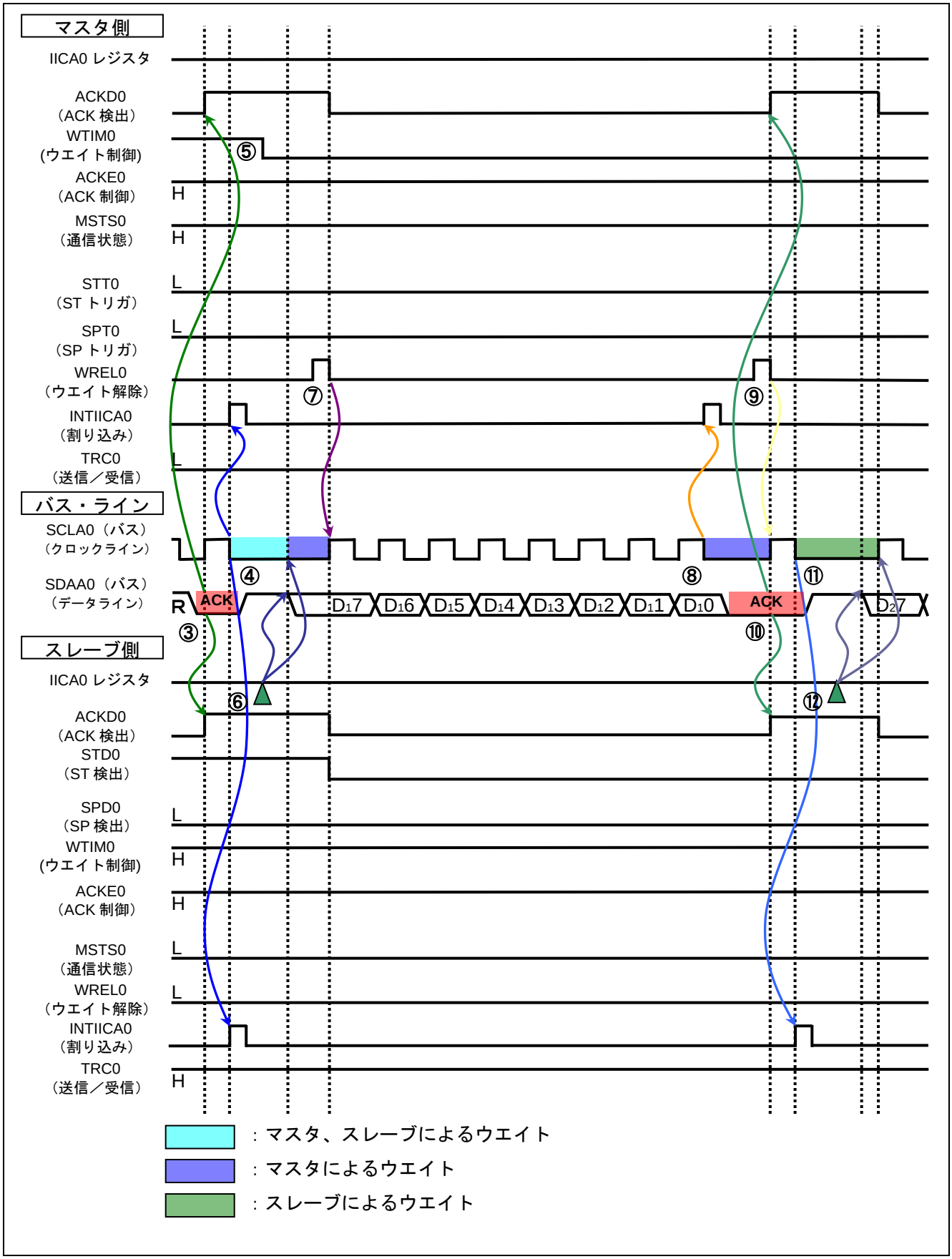
⑦ マスタ側がウエイトを解除（WREL0=1）すると、スレーブ側からマスタ側にデータ転送が開始されます。

注 受信アドレスと自局アドレスが不一致の場合は、スレーブ側は ACK をマスタ側へ返しません（NACK）。また、スレーブ側の INTIICA0 割り込み（アドレス一致の割り込み）は発生せず、スレーブ側からのウエイトも入りません。ただし、マスタ側は ACK、NACK のどちらの場合でも、INTIICA0 割り込み（アドレス送信完了の割り込み）が発生します。

RL78 ファミリでは、自局アドレス（7 ビット）を SVA0 レジスタの上位 7 ビットで表現します。SVA0 レジスタの最下位ビットは 0 固定です。

（6）スレーブ→マスタ通信 2（アドレス～データ～データ）

図 1.7 IIC 通信タイミング・チャート（スレーブ→マスタ通信例）(2/3)



③ 受信したアドレスとスレーブのアドレスが一致した場合、スレーブ側のハードウェアより ACK がマスタ側へ送信され、9 クロック目の立ち上がり時に、マスタ側で ACK が検出（ACKD0=1）されます。

④ 9 クロック目の立ち下がりで、マスタ側の INTIICA0 割り込み（アドレス送信完了の割り込み）が発生し、スレーブはアドレスが一致した場合、INTIICA0 割り込み（アドレス一致の割り込み）が発生します。さらに、マスタ側とアドレスが一致したスレーブ側によるウエイト（SCLA0 ライン：ロウ・レベル）を発生させます。

⑤ マスタ側は受信動作となるため、8 クロック・ウエイト（WTIM0=0）を選択します。

⑥ スレーブ側が IICA0 レジスタに送信データを書き込み、スレーブ側はウエイトを解除します。

⑦ マスタ側がウエイトを解除（WREL0=1）すると、スレーブ側からマスタ側にデータ転送が開始されます。

⑧ 8 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側の INTIICA0 割り込み（転送完了の割り込み）が発生します。また、マスタ側のハードウェアより ACK がスレーブ側へ送信されます。

⑨ マスタ側は受信データを読み出して、ウエイトを解除（WREL0=1）します。

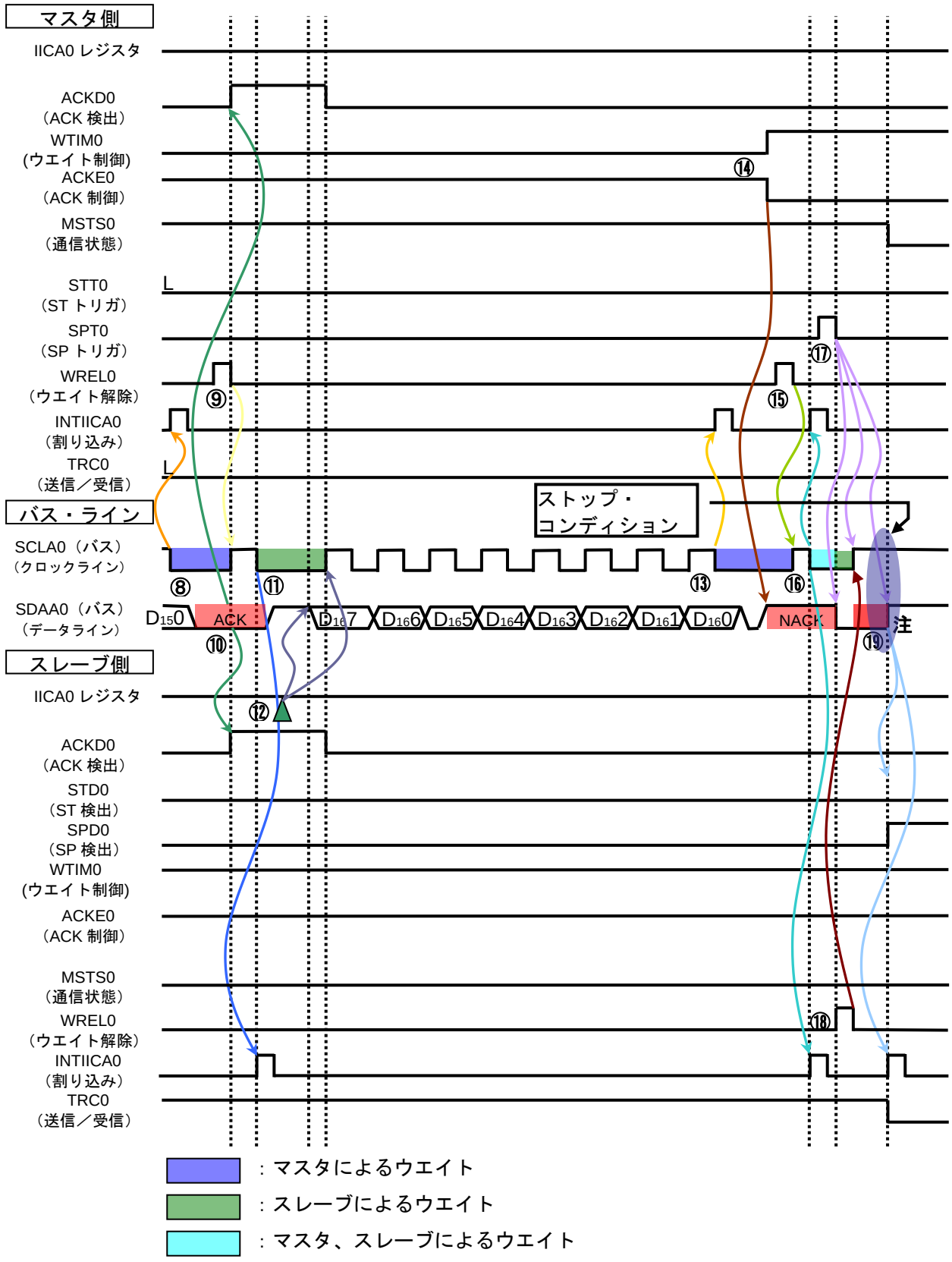
⑩ 9 クロック目の立ち上がり時に、スレーブ側で ACK が検出（ACKD0=1）されます。

⑪ 9 クロック目の立ち下がりで、スレーブ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側は INTIICA0 割り込み（転送完了の割り込み）が発生します。

⑫ スレーブ側の IICA0 レジスタに送信データを書き込むと、スレーブ側のウエイトは解除され、スレーブ側からマスタ側にデータ転送が開始されます。

（7）スレーブ→マスタ通信 3（データ～データ～ストップ・コンディション）

図 1.8 IIC 通信タイミング・チャート（スレーブ→マスタ通信例）(3/3)



注 ストップ・コンディション発行後、SCLA0 が立ち上がってからストップ・コンディションが生成されるまでに標準モード設定時 4.0μs 以上／ファースト・モード設定時 0.6μs 以上です。

⑧ 8 クロック目の立ち下がりで、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側の INTIICA0 割り込み（転送完了の割り込み）が発生します。さらに、マスタ側のハードウェアより ACK がスレーブ側へ送信されます。

⑨ マスタ側は受信データを読み出して、ウエイトを解除（WREL0=1）します。

⑩ 9 クロック目の立ち上がり時に、スレーブ側で ACK が検出（ACKD0=1）されます。

⑪ 9 クロック目の立ち下がりで、スレーブ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、スレーブ側は INTIICA0 割り込み（転送完了の割り込み）が発生します。

⑫ スレーブ側の IICA0 レジスタに送信データを書き込むと、スレーブ側のウエイトは解除され、スレーブ側からマスタ側にデータ転送が開始されます。

⑬ 8 クロック目の立ち下がりで、マスタ側 INTIICA0 の割り込み（転送完了の割り込み）が発生し、マスタ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生します。また、マスタ側のハードウェアより ACK がスレーブ側へ送信されます

⑭ マスタ側はスレーブ側に最終データであること（通信終了）を通知するため、NACK 応答に設定（ACKE0=0）し、ウエイト・タイミングを 9 クロック・ウエイト（WTIM0=1）に変更します。

⑮ マスタ側がウエイトを解除（WREL0=1）すると、スレーブ側は 9 クロック目の立ち上がりで NACK を検出（ACKD0=0）します。

⑯ 9 クロック目の立ち下がりで、マスタ側とスレーブ側によるウエイト（SCLA0 ライン：ロウ・レベル）が発生し、マスタ側とスレーブ側で INTIICA0 割り込み（転送完了の割り込み）が発生します。

⑰ マスタ側でストップ・コンディション発行（SPT0=1）すると、SDAA0 ラインが立ち下がり、マスタ側のウエイトが解除されます。その後、マスタ側は SCLA0 ラインが立ち上がるまで待機します。

⑱ スレーブ側は通信を終了するためにウエイトを解除（WREL0=1）します。スレーブによるウエイトが解除されると、SCLA0 ラインが立ち上がります。

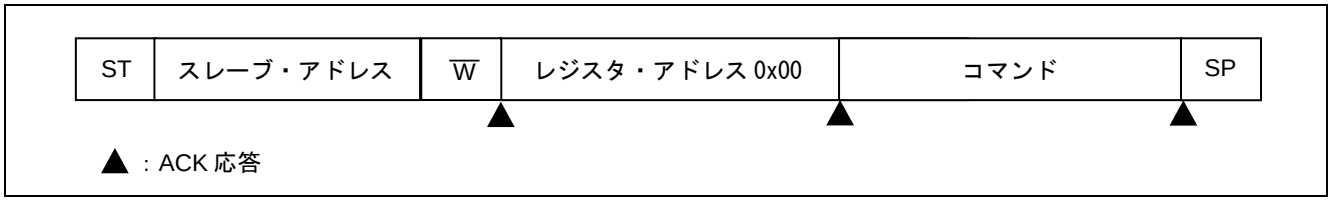
⑲ マスタ側は SCLA0 ラインが立ち上がったことを確認すると、ストップ・コンディションセットアップ時間経過後、SDAA0 ラインを立ち上げ、ストップ・コンディションを発行します。ストップ・コンディションが生成されると、スレーブ側でストップ・コンディションが検出（SPD0=1）され、マスタ側、スレーブ側で INTIICA0 割り込み（ストップ・コンディションの割り込み）が発生します。

1.2 シリアル RAM の制御

1.2.1 コマンド設定

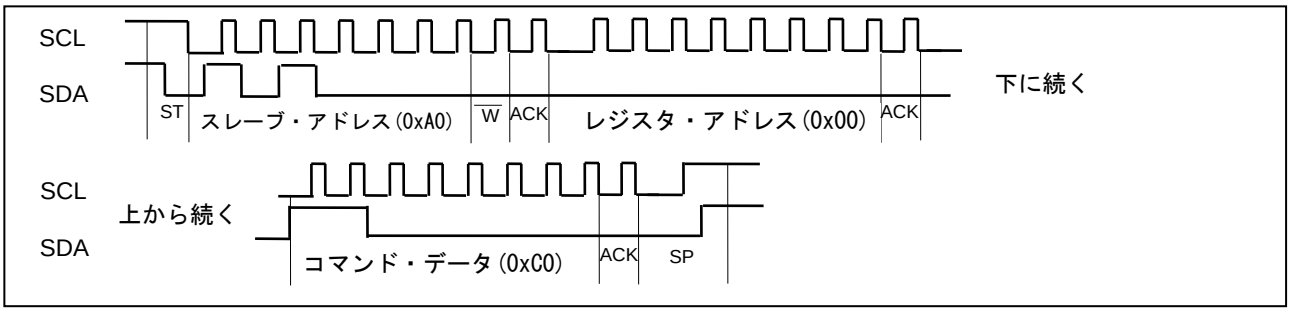
本アプリケーションノートでは、スレーブの動作をコマンドで指定します。コマンド設定のシーケンスを図 1.9 に示し、コマンド設定のタイミング・チャートを図 1.10 に示します。

図 1.9 コマンド設定のシーケンス



マスタは、スタート・コンディション（ST）に続いて、スレーブ・アドレス 0xA0（0b1010000 と転送方向 Wを合わせた 8 ビット）を送信します。このスレーブ・アドレスでスレーブが選択されます。次に、マスタは、レジスタ・アドレス（0x00）を送信して、次にコマンドが送信されることをスレーブに示します。コマンド送信後、ストップ・コンディション（SP）を送信して、通信を終了します。

図 1.10 コマンド設定のタイミング・チャート



コマンド機能の一覧を表 1.2 に示します。

レジスタ・アドレス 0x00（コマンド・レジスタ）に書き込むデータのビット 7（MSB）は、コマンドが有効か無効かを示します。ビット 7 が 1 のとき、スレーブは有効なコマンドであると判断します。ビット 7 が 0 のとき、スレーブは無効なコマンドと判断して無視します。ビット 6 は、メモリ機能を使用するか使用しないかを示します。メモリ機能を使用する場合は、ビット 6 に 1 を設定します。ビット 5～3 は未使用のため、全て 0 に設定します。ビット 2 は、書き込みを禁止するか許可するかを示します。ビット 2 が 1 の期間（書き込みを禁止してから再び書き込みを許可にするまで）、スレーブは書き込みデータ受信時に NACK 応答して通信から退避します。ビット 1 は、メモリを初期化するか何もしないかを示します。ビット 1 が 1 のとき、ビット 0 で指定した初期化データでスレーブのメモリを初期化します。初期化完了後、スレーブはコマンド・レジスタのビット 1 を 0 に設定します。

表 1.2 コマンド機能

ビット	意味	説明
7	コマンド設定	1：コマンドは有効、0：コマンドは無効
6	メモリ機能選択	1：メモリ機能を使用する、0：メモリ機能を使用しない
5～3	未使用	0 に固定
2	書き込み選択	1:書き込みを禁止する、0：書き込みを許可する
1	初期化選択	1：メモリ（シリアル RAM 領域）を初期化する、0：何もしない
0	初期化データ選択	1：レジスタ・アドレスの下位 7 ビットの値[注]、0：0x00

注. シリアル RAM 領域のアドレス毎に、そのアドレスの下位 7 ビットの値を書き込む。 例えば、レジスタ・アドレス 0x80、0x81、0x82 の場合は、それぞれ 0x00、0x01、0x02 を書き込む。

1.2.2 連続データ書き込み

レジスタ・アドレスを指定して連続してスレーブのシリアル RAM にデータを書き込む場合について、シーケンスを図 1.11 に示し、タイミング・チャートを図 1.12 に示します。

マスタからはスタート・コンディション（ST）に続いて、スレーブ・アドレス 0xA0（0b1010000 と転送方向 Wを合わせた 8 ビット）を送信します。スレーブ・アドレスに続いて、シリアル RAM の内部アドレスを指定するレジスタ・アドレスを送信します。その後、書き込みデータを順番に送信します。最終のデータ送信後、ストップ・コンディション（SP）を生成して通信を完了します。

図 1.11 レジスタ・アドレス指定での連続データ書き込みシーケンス

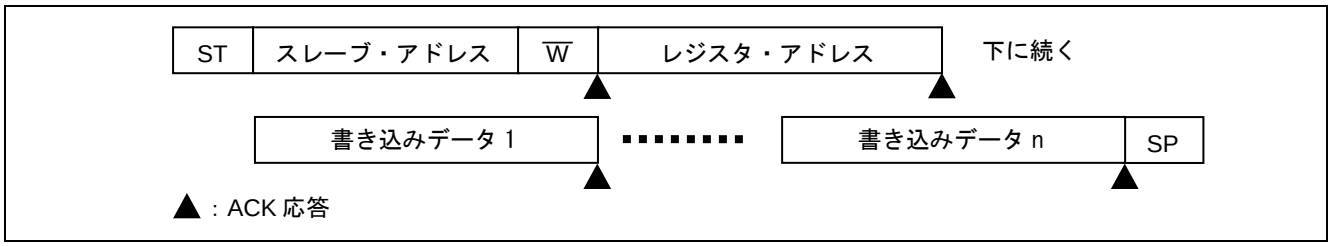
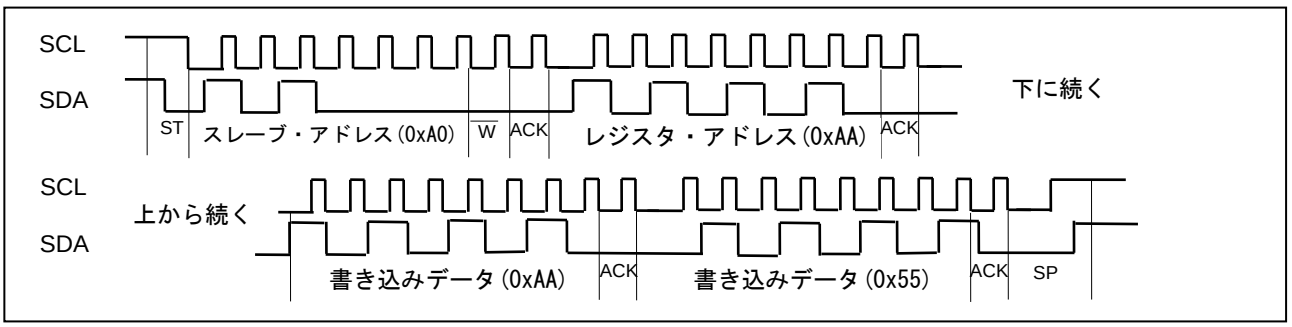
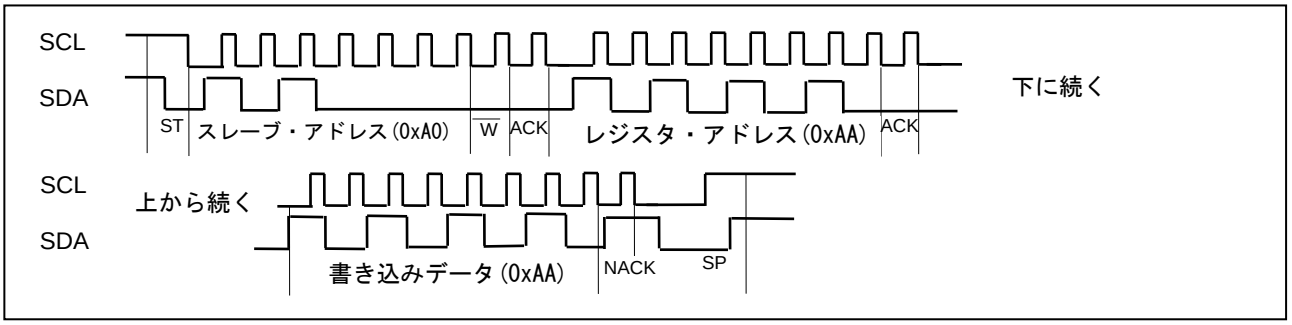


図 1.12 レジスタ・アドレス指定での連続データ書き込みのタイミング・チャート



書き込み禁止時のデータ書き込みのタイミング・チャートを図 1.13 に示します。スレーブは、スレーブ・アドレス、レジスタ・アドレスに対して ACK 応答しますが、書き込みデータに対しては NACK 応答して通信から退避します。マスタは、NACK 応答を確認したら、ストップ・コンディション生成して通信を終了します。

図 1.13 書き込み禁止時のデータ書き込みのタイミング・チャート



1.2.3 連続データ読み込み

レジスタ・アドレスを指定して連続してスレーブからデータを読み込む場合について、シーケンスを図 1.14 に示し、タイミング・チャートを図 1.15 に示します。

マスタからはスタート・コンディション（ST）に続いて、スレーブ・アドレス 0xA0（0b1010000 と転送方向 Wを合わせた 8 ビット）を送信します。スレーブ・アドレスに続いて、シリアル RAM の内部アドレスを指定するレジスタ・アドレスを送信します。その後、リスタート・コンディション（ST）に続いて、スレーブ・アドレス 0xA1（0b1010000 と転送方向 R を合わせた 8 ビット）を送信します。その後は、指定されたレジスタ・アドレスから順番にデータが送信されます（シーケンシャルリード）。受信データに対してマスタが NACK 応答すると、スレーブは送信を中止します。最後にストップ・コンディション（SP）を生成して通信を完了します。

図 1.14 レジスタ・アドレス指定での連続データ読み出しシーケンス

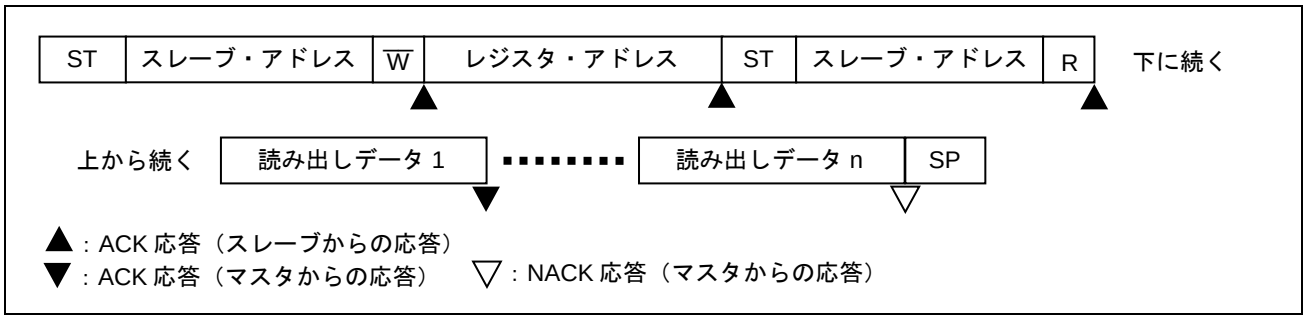
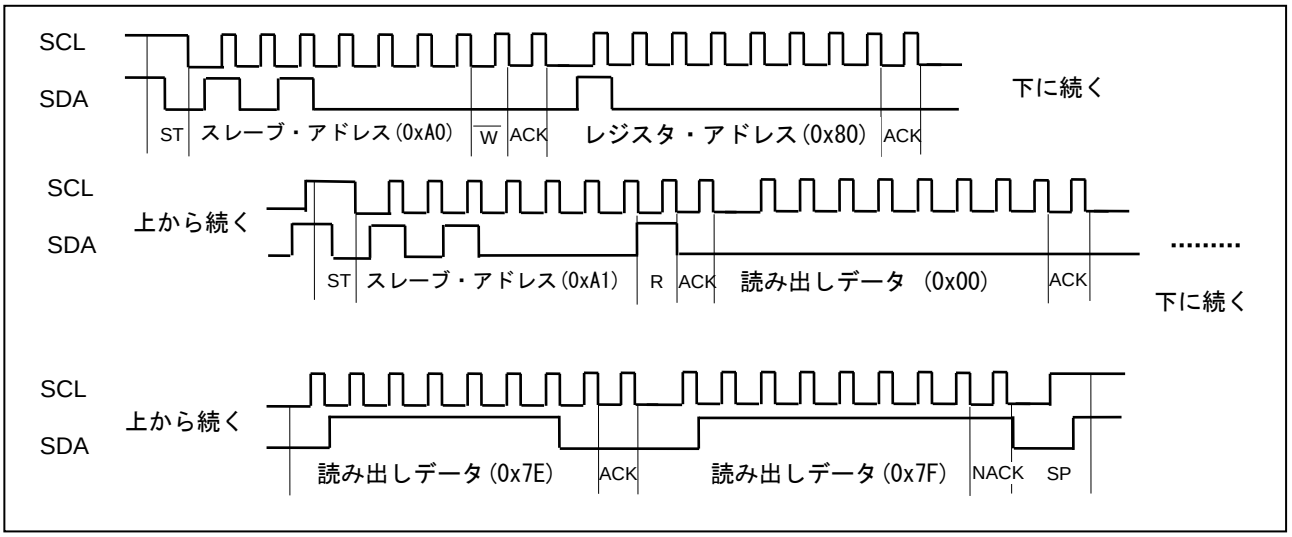


図 1.15 レジスタ・アドレス指定での連続データ読み出しのタイミング・チャート



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	RL78/G22（R7F102GGE2DFB）
使用ボード	RL78/G22-48p Fast Prototyping Board (RTK7RLG220C00000BJ)
動作周波数	高速オンチップ・オシレータ・クロック：32MHz
動作電圧	5.0V（1.8V～5.5V で動作可能）
統合開発環境（CS+）	ルネサス エレクトロニクス製 CS+ V8.11.00
C コンパイラ（CS+）	ルネサス エレクトロニクス製 CC-RL V1.13.01
統合開発環境（e^2 studio）	ルネサス エレクトロニクス製 e^2 studio V2024-04
C コンパイラ（e^2 studio）	ルネサス エレクトロニクス製 CC-RL V1.13.01
統合開発環境（IAR）	IAR Systems 製 IAR Embedded Workbench for Renesas RL78 V5.10.1
C コンパイラ（IAR）	IAR Systems 製 IAR C/C++ Compiler for Renesas RL78 V5.10.1
スマート・コンフィグレータ（SC）	ルネサス エレクトロニクス製 V1.10.0
ボードサポートパッケージ（BSP）	ルネサス エレクトロニクス製 V1.6.2

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。

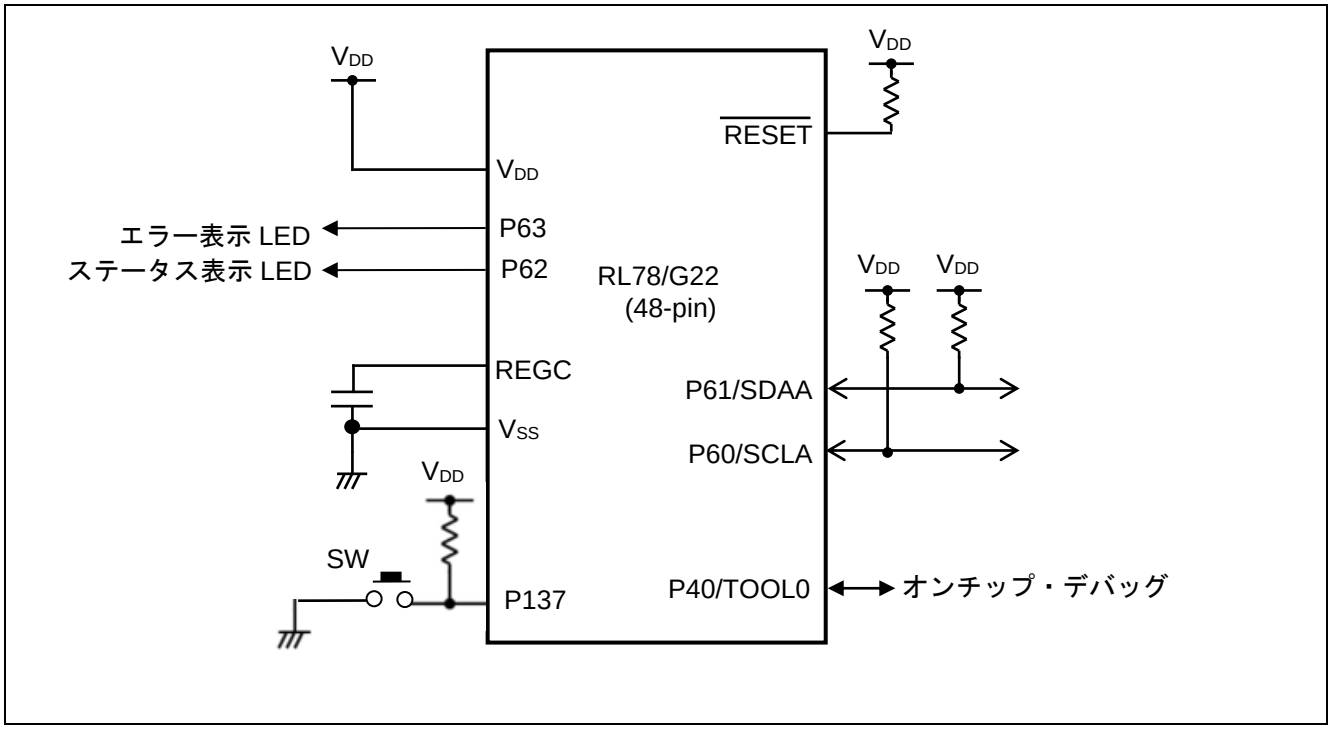
RL78/G22 シリアル・インタフェース IICA (スレーブ送受信) (R01AN7345J) アプリケーションノート

4. ハードウェア説明

4.1 ハードウェア構成例

図 4.1 に本アプリケーションノートで使用するハードウェア構成例を示します。

図 4.1 ハードウェア構成



注意 1 この回路イメージは接続の概要を示す為に簡略化しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください（入力専用ポートは個別に抵抗を介して V_{DD} 又は V_{SS} に接続して下さい）。

2 V_{DD} は LVD0 にて設定したリセット解除電圧 (V_{LVD0}) 以上にしてください。

4.2 使用端子一覧

表 4.1 に使用端子と機能を示します。

表 4.1 使用端子と機能

端子名	入出力	内容
P60/SCLA0	入出力	IICA0 のシリアル・クロック入出力端子
P61/SDAA0	入出力	IICA0 のシリアル・データ送受信用端子
P62	出力	ステータス表示 LED ドライブ信号
P63	出力	エラー表示 LED ドライブ信号
P137	入力	動作開始指定用 SW 入力信号

5. ソフトウェア説明

5.1 動作概要

本アプリケーションノートでは、シリアル・インタフェース IICA を利用して IICA マスタ送受信（アドレス送信、データ送受信）動作を行います。

(1) シリアル・インタフェース IICA の初期設定を行います。

＜設定条件＞

- 動作モードをファースト・モードに設定します。
- 転送クロックを 400kHz に設定します。
- 自局アドレスを 0x10 に設定します。
- デジタル・フィルタの動作をオンに設定します。
- 9 クロック目で割り込みが発生するように設定します。
- ストップ・コンディション割り込み禁止に設定します。
- P60/SCLA0 端子をシリアル・クロック（入出力設定)に、P61/SDAA0 端子をデータ送受信（入出力設定）に設定します。

(2) 通信用バッファ（受信用 8×16＝128 バイト、送信用 65 バイト）を準備します。

(3) P137 に接続されている SW が押されると、マスタからスレーブにコマンド 0xC0 を送信します。その後、コマンド 0xC3 を送信して、スレーブのシリアル RAM 領域を「レジスタ・アドレスの下位 7 ビットの値」で初期化します。

(4) スレーブのレジスタ・アドレス 0x80 から 16 バイト単位で 8 回（計 128 バイト）の連続データ読み出しを行います。
スレーブからの NACK 応答があった場合、P63 に接続された LED を点灯します。

(5) スレーブから読み出したデータ（128 バイト分）が事前に用意している期待値と一致しているか確認します。期待値と異なる場合、P63 に接続された LED を 500ms 周期で点滅させます。

(6) 書き込むデータを変更して、スレーブにデータ送信します。
スレーブからの NACK 応答があった場合、P63 に接続された LED を点灯します。

送信データ：
上記(4)の受信データが 0x00～0x7F（単調増加）の場合、0xFF～0x80（単調減少）
上記(4)の受信データが 0xFF～0x80（単調減少）の場合、0x00～0x7F（単調増加）

(7) P62 に接続されたステータス表示用の LED を反転させ、10ms 待ってから(4)から繰り返します。

注意 本サンプルコードは、RL78/G22 シリアル・インタフェース IICA（スレーブ送受信）（R01AN7345J）アプリケーションノートに対応しています。

5.2 オプション・バイトの設定一覧

表 5.1 にオプション・バイト設定を示します。

表 5.1 オプション・バイト設定

アドレス	設定値	内容
000C0H / 020C0H	11101111B	ウォッチドッグ・タイマ動作停止 (リセット解除後、カウント停止)
000C1H / 020C1H	11111100B	LVD0 検出電圧: リセット・モード 立ち上がり時 TYP. 1.90 V (1.84 V ～ 1.95 V) 立ち下がり時 TYP. 1.86 V (1.80 V ～ 1.91 V)
000C2H / 020C2H	11101000B	HSモード、 高速オンチップ・オシレータ・クロック : 32MHz
000C3H / 020C3H	10000101B	オンチップ・デバッグ許可

5.3 定数一覧

表 5.2 に　サンプルコードで使用する定数を示します。

表 5.2 サンプルコードで使用する定数

定数名	設定値	内容
MAX_DATA	64	IIC 送信データ長
SLAVE_ADDR	0xA0	スレーブのアドレス
RAM_TOP	0x80	スレーブの RAM 領域の先頭アドレス
LED_ON	0	LED を点灯するデータ
RETRY	10	リトライ回数の上限値
WAITTIME	1000	スタート・コンディション検出待ちループ
STS_MASK	0x03	IICA0 ステータスのマスクデータ
DUMMY_DATA	0xFF	受信起動用のダミーデータ
BUS_BUSY	0x8C	IIC バス・ビジー
ON_COMMU	0x01	通信中を示すデータ
IIC_SUCCESS	0x00	通信の正常終了
IIC_USING	0x01	通信中
NO_ACK	0x84	送信データへの NACK 応答
NO_SLAVE	0x88	スレーブ・アドレスへの NACK 応答
RAM_COMMAND[4]	-	スレーブへのコマンドのリスト
RAM_DATA[4][64]	-	スレーブに書き込むデータのテーブル

5.4 変数一覧

表 5.3 にグローバル変数を示します。

表 5.3 グローバル変数

型	変数名	内容	使用関数
uint8_t	g_read_ram[8][16]	IICA0 受信バッファ	main()
uint8_t	g_write_data	IICA0 送信バッファ	r_IICA0_put_data()、 r_IICA0_get_data()、main()
uint8_t	g_1ms_status	1ms のカウントフラグ	wait_ms()、 r_Config_IT_interrupt()
uint8_t	g_1ms_count	1ms 単位のカウンタ	wait_ms()、 r_Config_IT_interrupt()
uint8_t	g_iica0_status	ICA0 のステータス	R_IICA0_check_comstate() R_IICA0_wait_comend() R_IICA0_bus_check() r_Config_IICA0_interrupt()
uint8_t *	gp_iica0_rx_address	IICA0 受信バッファ アドレス	R_IICA0_Master_Receive() r_Config_IICA0_interrupt()
uint16_t	g_iica0_rx_len	IICA0 受信データ長	R_IICA0_Master_Receive() r_Config_IICA0_interrupt()
uint16_t	g_iica0_rx_cnt	IICA0 受信データ カウント	R_IICA0_Master_Receive() r_Config_IICA0_interrupt()
uint8_t *	gp_iica0_tx_address	IICA0 送信バッファ アドレス	R_IICA0_Master_Send () r_Config_IICA0_interrupt()
uint16_t	g_iica0_tx_cnt	IICA0 送信データ カウント	R_IICA0_Master_Send () r_Config_IICA0_interrupt()
uint16_t	g_us_rest_time	μs 単位の待ち時間	set_delay_us() r_Config_TAU0_0_interrupt()
uint16_t	g_ms_timer1S	1ms のカウンタ	r_Config_TAU0_1_interrupt()

5.5 関数一覧

表 5.4 に関数を示します。

表 5.4 関数

関数名	概要
wait_ms	ms 単位の時間待ち処理
r_wait_SW	スイッチの押下待ち処理
r_IICA0_put_data	スレーブへのデータ送信処理
r_IICA0_get_data	スレーブからのデータ受信処理
R_IICA0_Master_Send	マスタ送信起動処理
R_IICA0_Master_Receive	マスタ受信起動処理
R_IICA0_wait_comend	通信完了待ち処理
R_IICA0_check_comstate	IICA0 通信状態確認処理
R_IICA0_StopCondition	IICA0 ストップ・コンディション発行処理
R_IICA0_bus_check	IIC バスの状態確認及びスタート・コンディション発行処理
r_Config_IICA0_interrupt	IICA0 割り込み処理
delay_us	μs 単位での時間待ち処理
set_delay_us	μs 単位での待ち時間設定処理
r_Config_TAU0_0_interrupt	タイマ・アレイ・ユニット チャネル 0 割り込み処理
r_Config_TAU0_1_interrupt	タイマ・アレイ・ユニット チャネル 1 割り込み処理

5.6 関数仕様

サンプルコードの関数仕様を示します。

[関数名] wait_ms

概要	ms 単位の時間待ち処理	
ヘッダ	Config_IT.h	
宣言	void wait_ms(uint8_t)	
説明	引数で指定された時間（ms 単位）待ちます。	
引数	待ち時間	-
リターン値	なし	
備考	なし	

[関数名] r_wait_SW

概要	スイッチの押下待ち処理	
ヘッダ	-	
宣言	void r_wait_SW(void)	
説明	P137 に接続された SW が押下されるのを待ちます。	
引数	なし	-
リターン値	なし	
備考	なし	

[関数名] r_IICA0_put_data

概要	スレーブへのデータ送信処理	
ヘッダ	Config_IICA0.h	
宣言	uint8_t r_IICA0_put_data(uint8_t s_addr、 uint8_t r_addr、 uint8_t __far * const buffer、 uint8_t tx_num)	
説明	指定したスレーブの指定アドレスに送信バッファから指定した数のデータを送信	
引数	s_addr	スレーブ・アドレス
	r_addr	レジスタ・アドレス
	buffer	送信データバッファのアドレス
	txnum	送信データ数
リターン値	通信ステータス（0x00 なら正常終了、それ以外はエラー）	
備考	なし	

[関数名] r_IICA0_get_data

概要	スレーブからのデータ受信処理	
ヘッダ	Config_IICA0.h	
宣言	uint8_t r_IICA0_get_data(uint8_t s_addr、 uint8_t r_addr、 uint8_t * const buffer、 uint8_t rx_num);	
説明	指定したスレーブの指定アドレスから受信バッファに指定した数のデータを受信	
引数	s_addr	スレーブ・アドレス
	r_addr	レジスタ・アドレス
	buffer	受信データバッファのアドレス
	rxnum	受信データ数
リターン値	通信ステータス（0x00 なら正常終了、それ以外はエラー）	
備考	なし	

[関数名] R_IICA0_Master_Send

概要	マスタ送信起動処理	
ヘッダ	Config_IICA0.h	
宣言	uint8_t R_IICA0_Master_Send(uint8_t adr、 uint8_t * const tx_buf、 uint16_t tx_num)	
説明	マスタ送信を設定します。	
引数	adr	スレーブ・アドレス
	tx_buf	送信データ・バッファ・アドレス
	tx_num	送信データ長
リターン値	通信ステータス（0x00 なら正常にバスを獲得、それ以外はエラー）	
備考	なし	

[関数名] R_IICA0_Master_Receive

概要	マスタ受信起動処理	
ヘッダ	Config_IICA0.h	
宣言	uint8_t R_IICA0_Master_Receive(uint8_t adr、 uint8_t * const rx_buf、 uint16_t rx_num)	
説明	マスタ受信を設定します。	
引数	adr	スレーブ・アドレス
	rx_buf	受信データ・バッファ・アドレス
	rx_num	受信データ長
リターン値	通信ステータス（0x00 なら正常にバスを獲得、それ以外はエラー）	
備考	なし	

[関数名] R_IICA0_wait_comend

概要	通信完了待ち処理	
ヘッダ	Config_IICA0.h	
宣言	uint8_t R_IICA0_wait_comend (void)	
説明	IIC 通信完了を待ちます。エラー検出で、ストップ・コンディションを発行します。	
引数	なし	
リターン値	通信ステータス（0x00 なら正常終了、それ以外はエラー）	
備考	なし	

[関数名] R_IICA0_check_comstate

概要	IICA0 通信状態確認処理
ヘッダ	Config_IICA0.h
宣言	uint8_t R_IICA0_check_comstate(void)
説明	IIC 通信状態のステータス（g_iica0_status）を戻します。
引数	なし
リターン値	通信ステータス（g_iica0_status の値）
備考	なし

[関数名] R_IICA0_StopCondition

概要	IICA0 ストップ・コンディション発行処理
ヘッダ	Config_IICA0.h
宣言	uint8_t R_IICA0_StopCondition(void)
説明	IIC バスを開放します。
引数	なし
リターン値	通信ステータス（0x00 なら正常終了、それ以外はエラー）
備考	なし

[関数名] R_IICA0_bus_check

概要	IIC バスの状態確認及びスタート・コンディション発行処理
ヘッダ	-
宣言	uint8_t R_IICA0_bus_check(void)
説明	IIC バスが使用可能かをチェックし、可能ならスタート・コンディションを発行し、スタート・コンディションを発行できたら g_iica0_status を通信中（0x01）に設定します。発行できなかったならエラーで戻ります。
引数	なし
リターン値	通信ステータス（0x00 なら正常終了、それ以外はエラー）
備考	なし

[関数名] r_Config_IICA0_interrupt

概要	IICA0 割り込み処理
ヘッダ	-
宣言	static void __near r_Config_IICA0_interrupt(void)
説明	IICA0 の割り込み処理を行います。
引数	なし
リターン値	なし
備考	本サンプルコードでは、コードジェネレータは初期設定のみ使用し、割り込み処理は全て新規に作成しています。

[関数名] delay_us

概要	μs 単位での時間待ち処理
ヘッダ	Config_TAU0_2.h
宣言	void delay_us(uint16_t us)
説明	引数で指定した時間（μs 単位）を待ち合わせます。
引数	us 待ち合わせ時間
リターン値	なし
備考	なし

[関数名] set_delay_us

概要	μs 単位での待ち時間設定処理	
ヘッダ	-	
宣言	void set_delay_us(uint16_t us)	
説明	μs 単位の待ち合わせ時間を設定します。	
引数	us	待ち合わせ時間
リターン値	なし	
備考	なし	

[関数名] r_Config_TAU0_0_interrupt

概要	タイマ・アレイ・ユニット チャネル 0 割り込み処理
ヘッダ	-
宣言	static void __near r_Config_TAU0_0_interrupt(void)
説明	タイマ・アレイ・ユニット チャネル 0（最大 100us のインターバル）の割り込み処理を行います。
引数	なし
リターン値	なし
備考	なし

[関数名] r_Config_TAU0_1_interrupt

概要	タイマ・アレイ・ユニット チャネル 1 割り込み処理
ヘッダ	-
宣言	static void __near r_Config_TAU0_1_interrupt(void)
説明	タイマ・アレイ・ユニット チャネル 1（最大 100us のインターバル）の割り込み処理を行います。
引数	なし
リターン値	なし
備考	なし

5.7 フローチャート

5.7.1 メイン処理

図 5.1～図 5.3 にメイン処理のフローチャートを示します。

図 5.1 メイン処理（1/3）

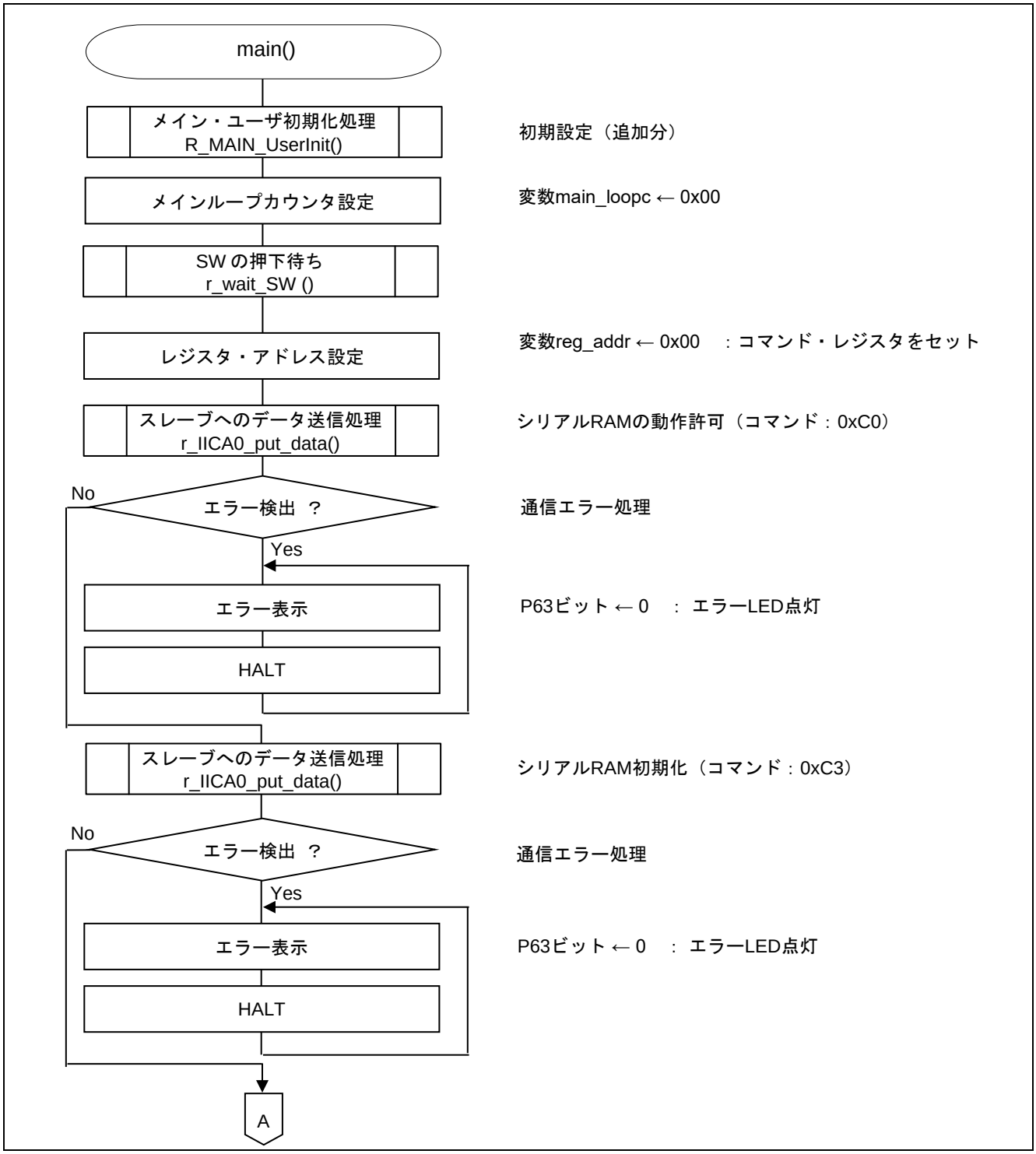


図 5.2　メイン処理（2/3）

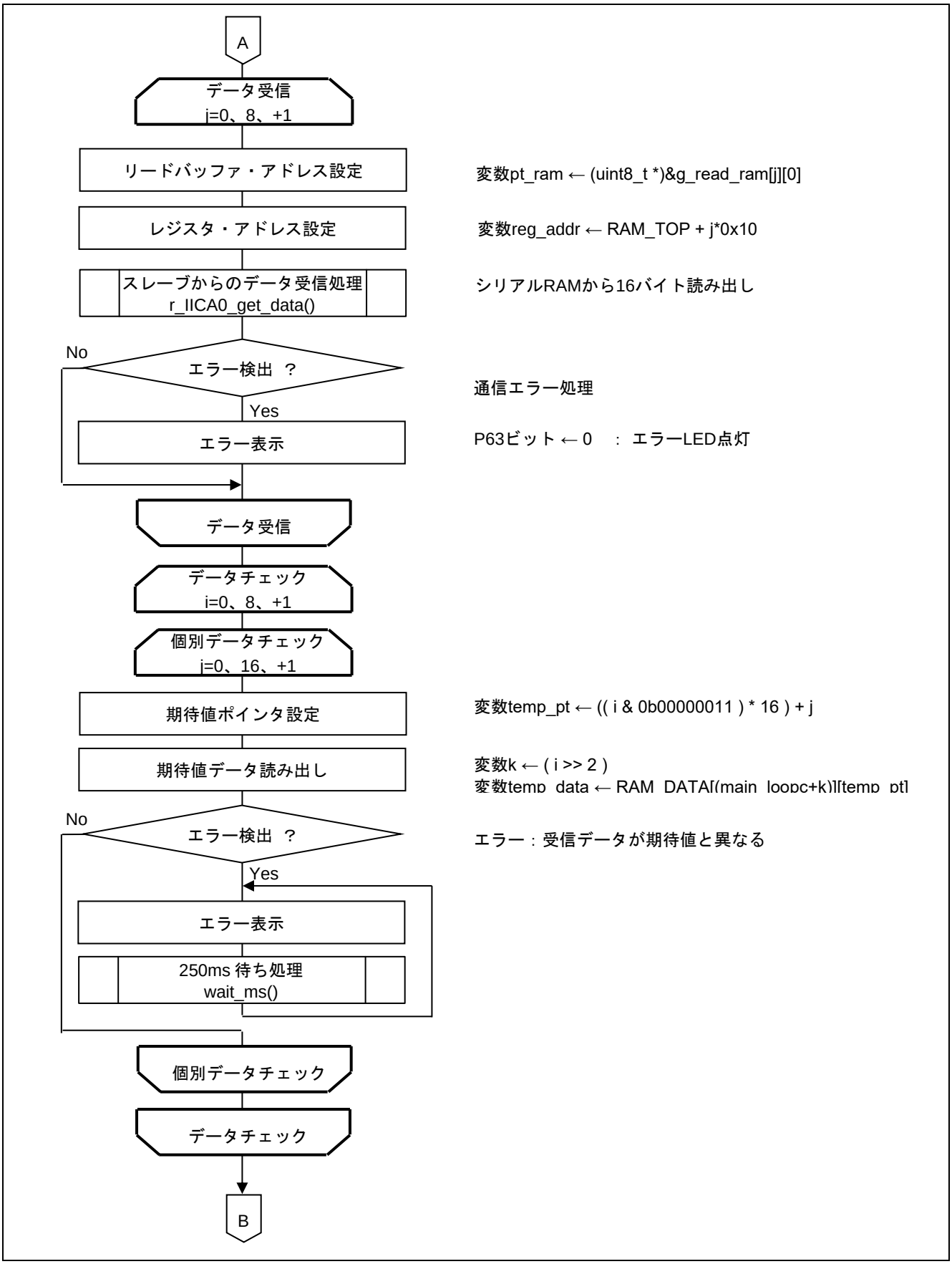
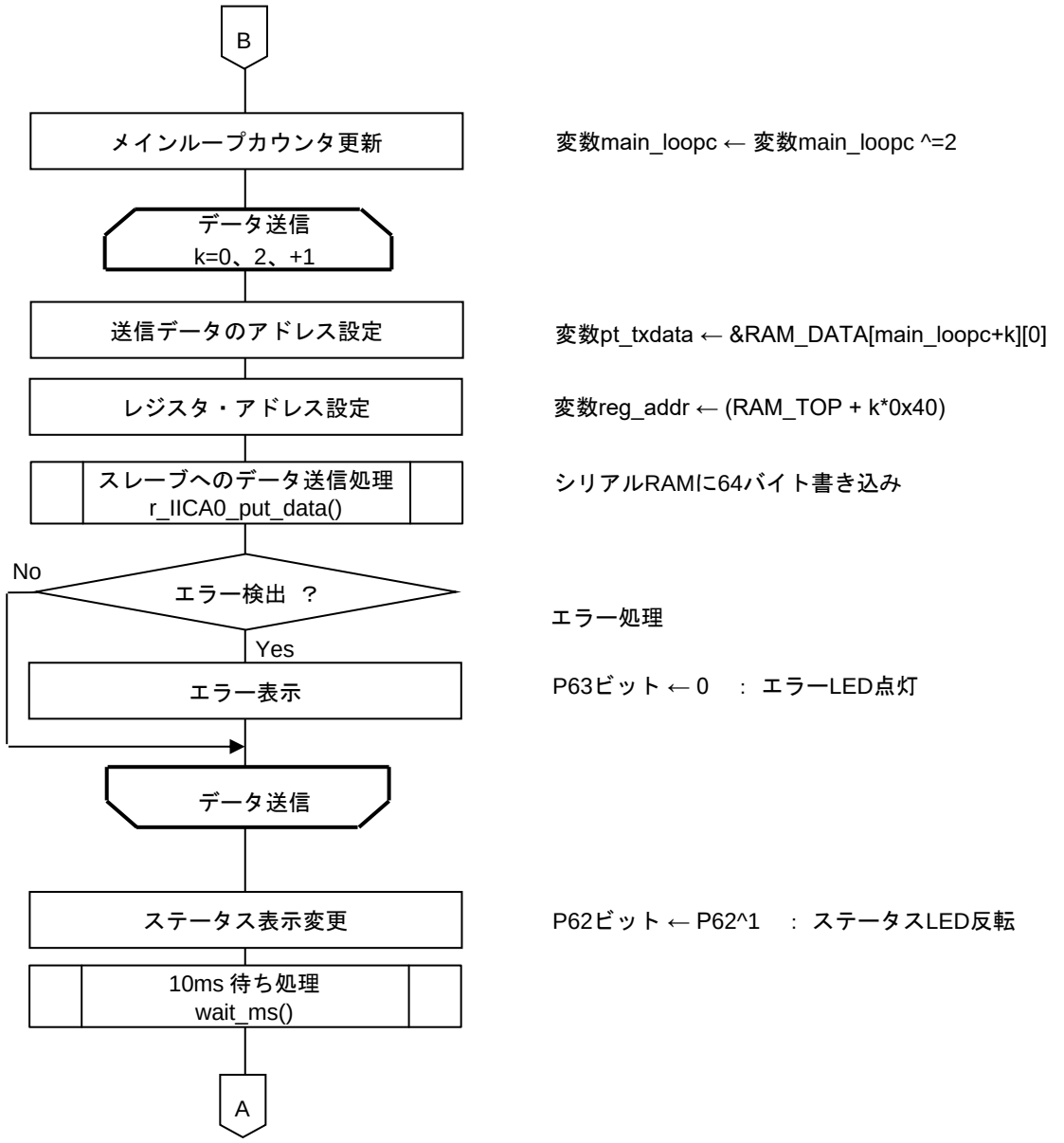

A
データ受信
j=0、8、+1
リードバッファ・アドレス設定
変数pt_ram ← (uint8_t *)&g_read_ram[j][0]
レジスタ・アドレス設定
変数reg_addr ← RAM_TOP + j*0x10
スレーブからのデータ受信処理
r_IICA0_get_data()
シリアルRAMから16バイト読み出し
No
エラー検出 ？
通信エラー処理
Yes
エラー表示
P63ビット ← 0　：エラーLED点灯
データ受信
データチェック
i=0、8、+1
個別データチェック
j=0、16、+1
期待値ポインタ設定
変数temp_pt ← ((i & 0b00000011) * 16) + j
期待値データ読み出し
変数k ← (i >> 2)
変数temp_data ← RAM_DATA[(main_loopc+k)][temp_pt]
No
エラー検出 ？
エラー：受信データが期待値と異なる
Yes
エラー表示
250ms 待ち処理
wait_ms()
個別データチェック
データチェック
B

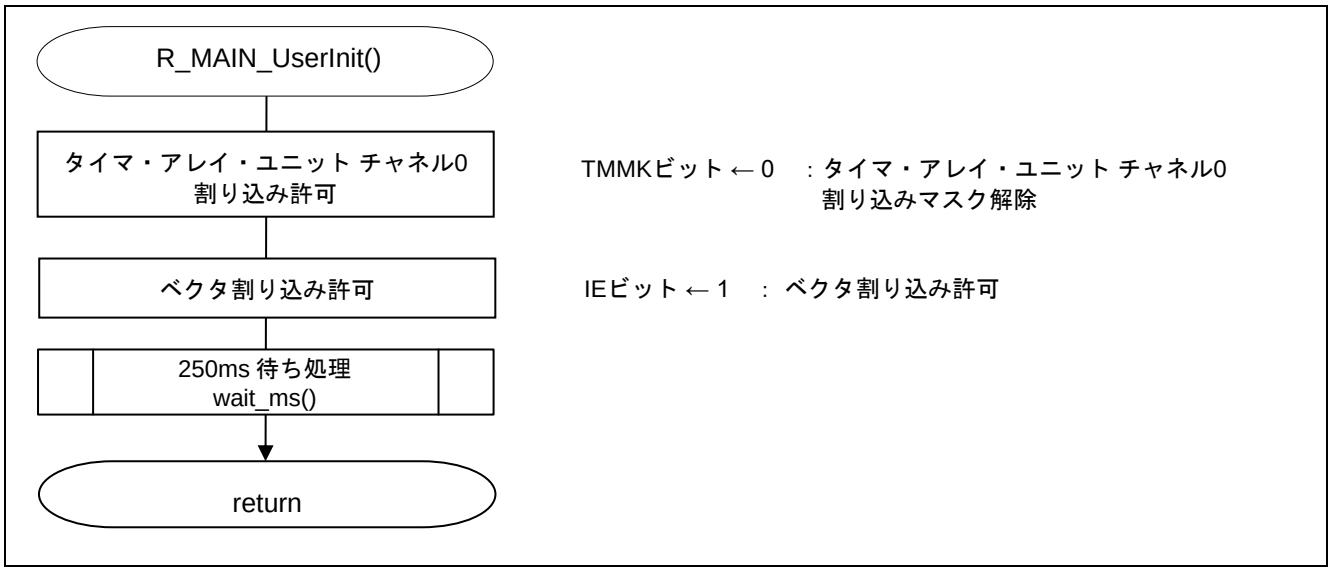
図 5.3　メイン処理（3/3）


B
メインループカウンタ更新
変数main_loopc ← 変数main_loopc ^=2
データ送信
k=0、2、+1
送信データのアドレス設定
変数pt_txdata ← &RAM_DATA[main_loopc+k][0]
レジスタ・アドレス設定
変数reg_addr ← (RAM_TOP + k*0x40)
スレーブへのデータ送信処理
r_IICA0_put_data()
シリアルRAMに64バイト書き込み
No
エラー検出 ？
エラー処理
Yes
エラー表示
P63ビット ← 0　：エラーLED点灯
データ送信
ステータス表示変更
P62ビット ← P62^1　：ステータスLED反転
10ms 待ち処理
wait_ms()
A

5.7.2 メイン・ユーザ初期化設定

図 5.4 にメイン・ユーザ初期化設定のフローチャートを示します。

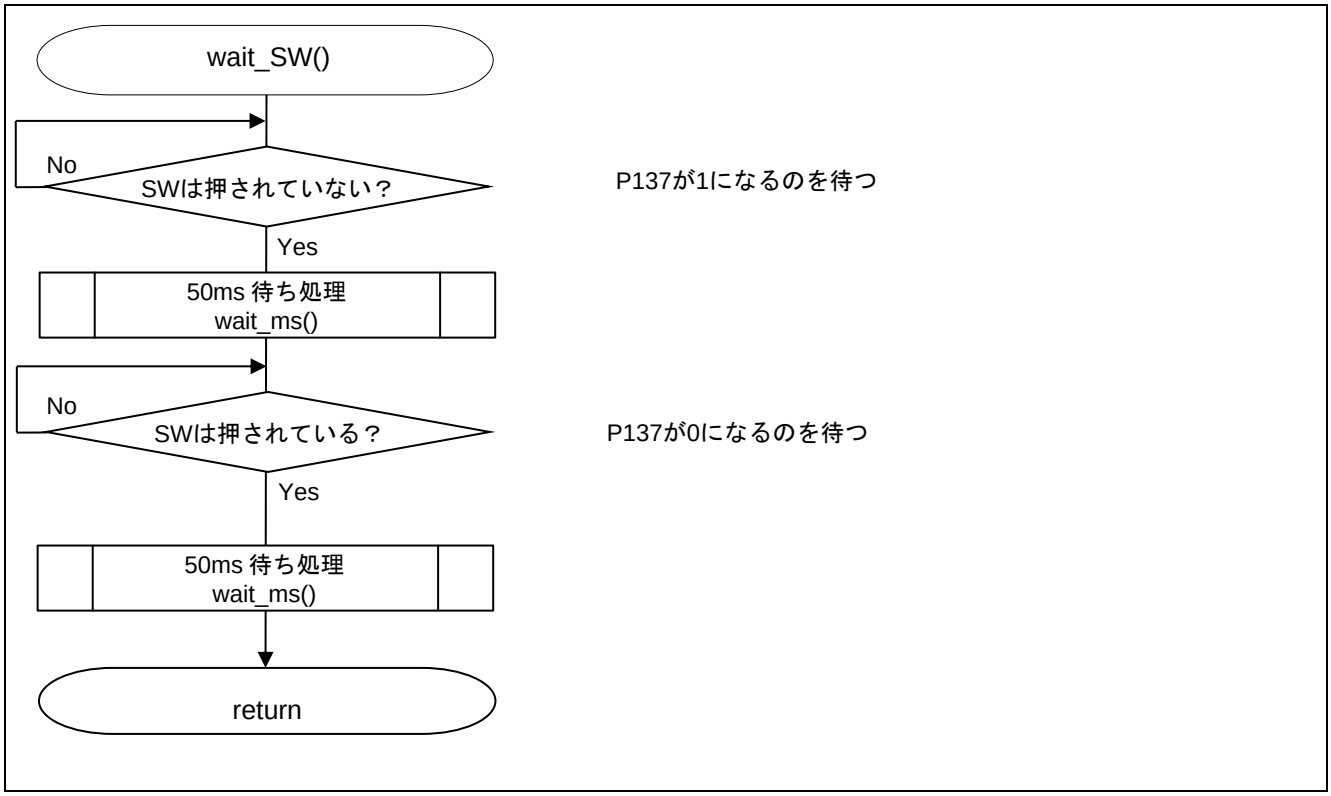
図 5.4 メイン・ユーザ初期化設定



5.7.3 スイッチの押下待ち処理

図 5.5 にスイッチの押下待ち処理のフローチャートを示します。

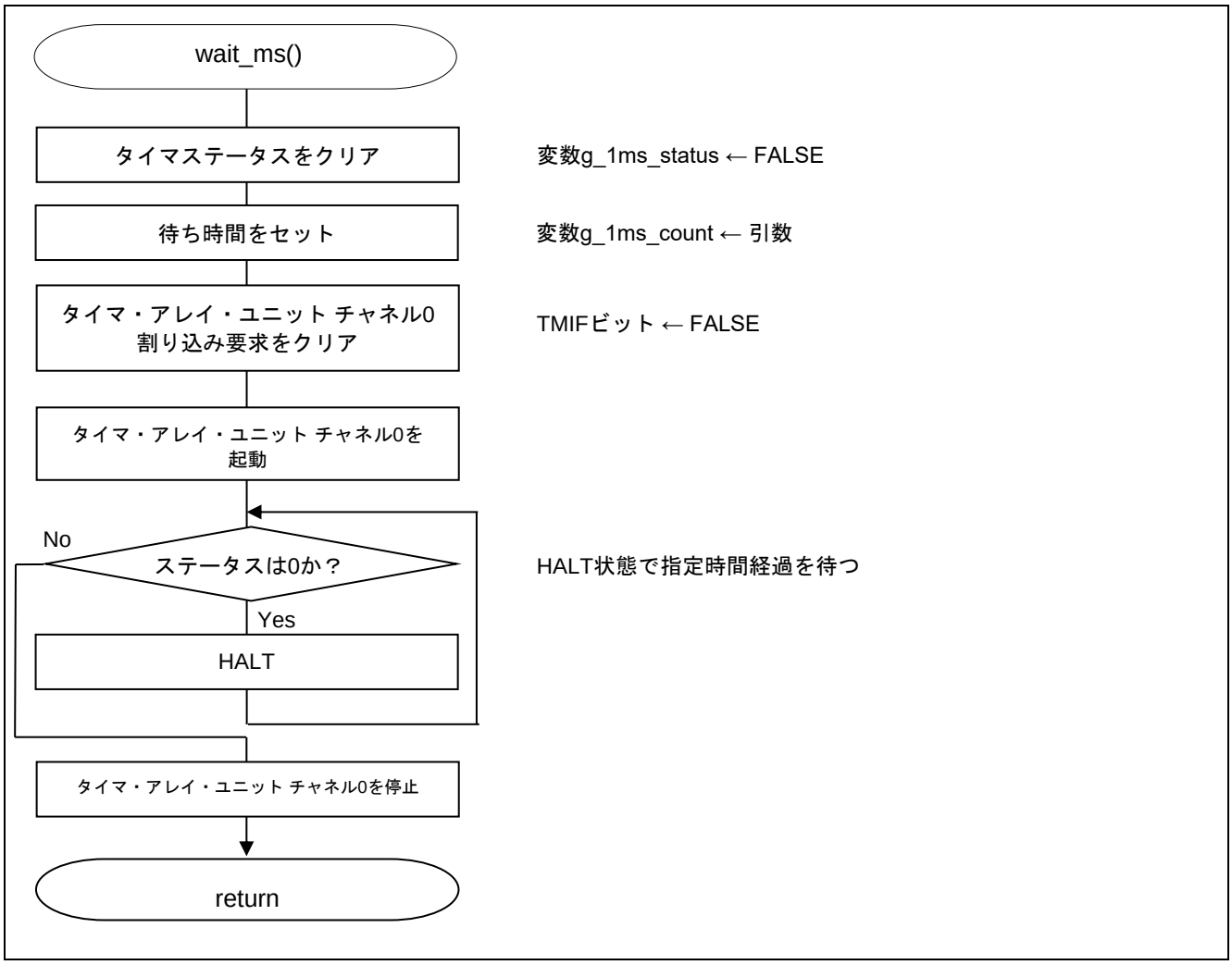
図 5.5 スイッチの押下待ち処理



5.7.4 ms 単位の時間待ち処理

図 5.6 に ms 単位の時間待ち処理のフローチャートを示します。

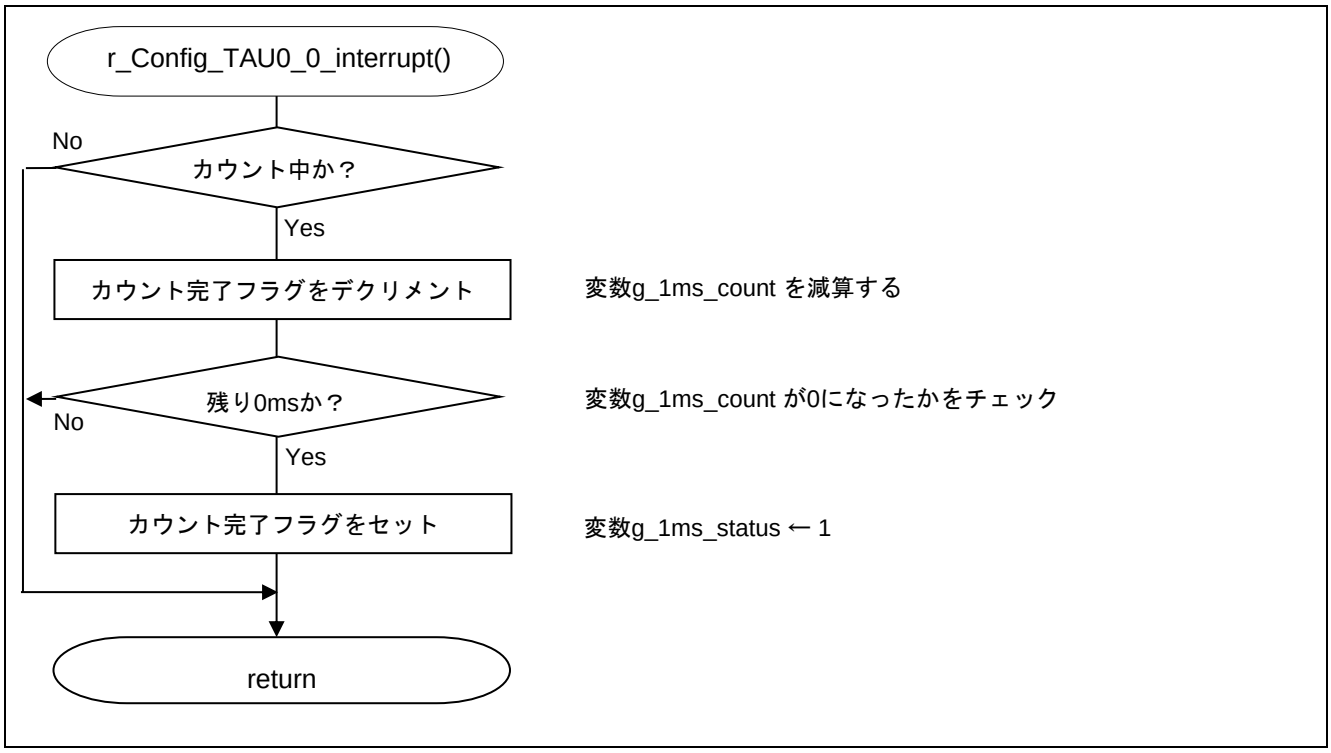
図 5.6 ms 単位の時間待ち処理



5.7.5 タイマ・アレイ・ユニット チャネル 0 割り込み処理

図 5.7 にタイマ・アレイ・ユニット チャネル 0 割り込み処理のフローチャートを示します。

図 5.7 タイマ・アレイ・ユニット チャネル 0 割り込み処理



5.7.6 スレーブへのデータ送信処理

図 5.8～図 5.9 にスレーブへのデータ送信処理のフローチャートを示します。

図 5.8 スレーブへのデータ送信処理（1/2）

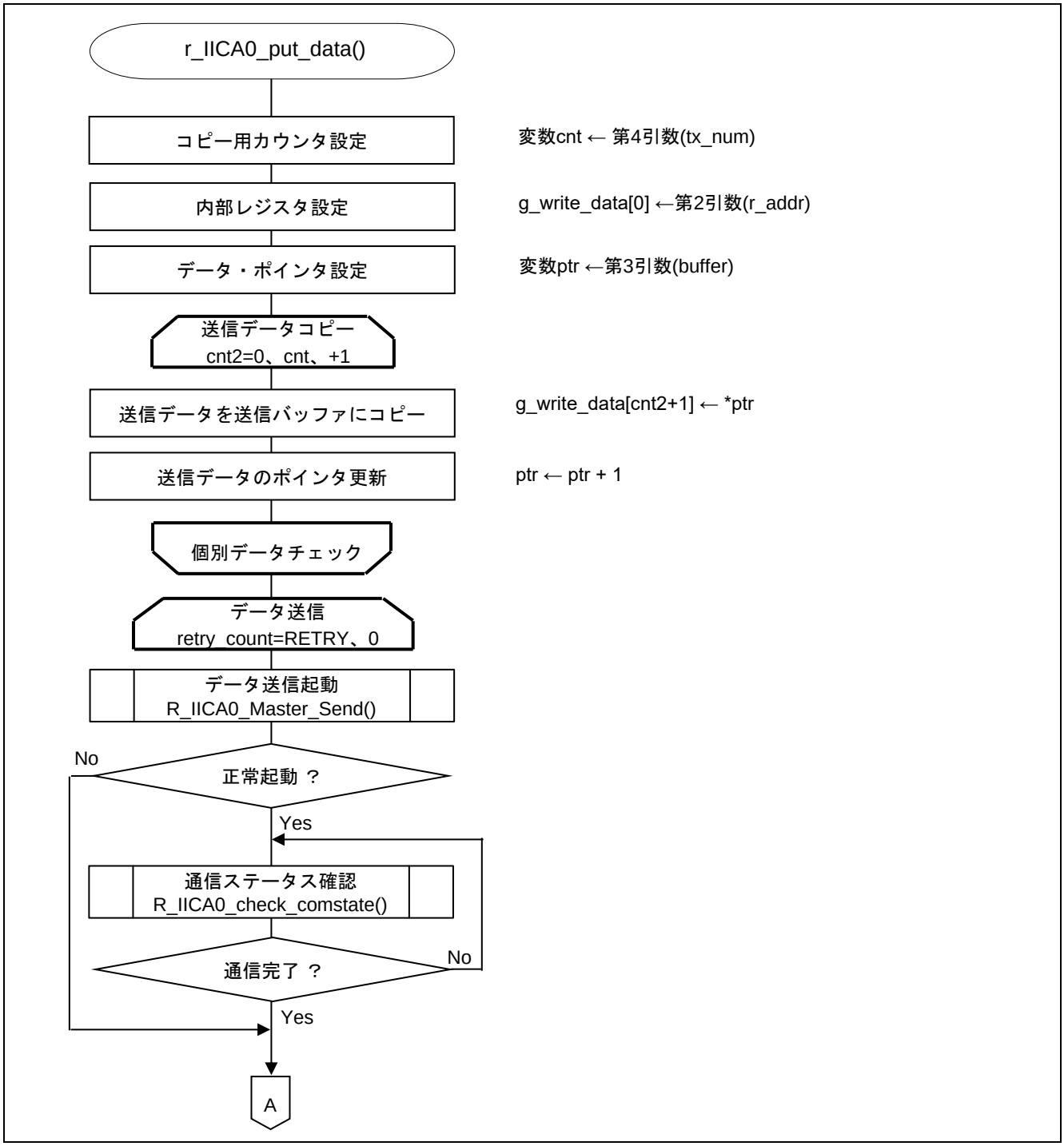
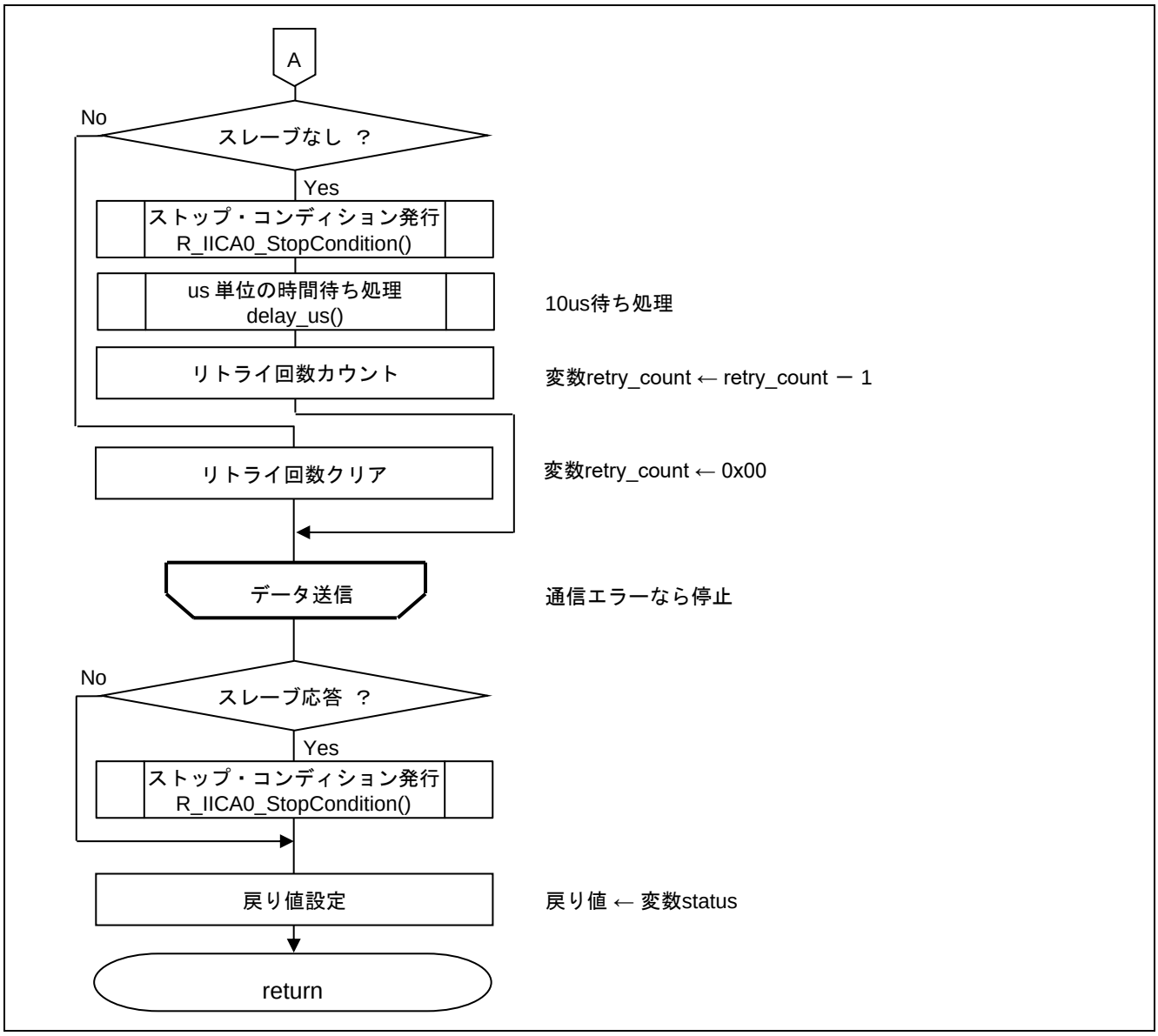


図 5.9　スレーブへのデータ送信処理（2/2）



5.7.7 スレーブからのデータ受信処理

図 5.10～図 5.11 にスレーブからのデータ受信処理のフローチャートを示します。

図 5.10 スレーブからのデータ受信処理（1/2）

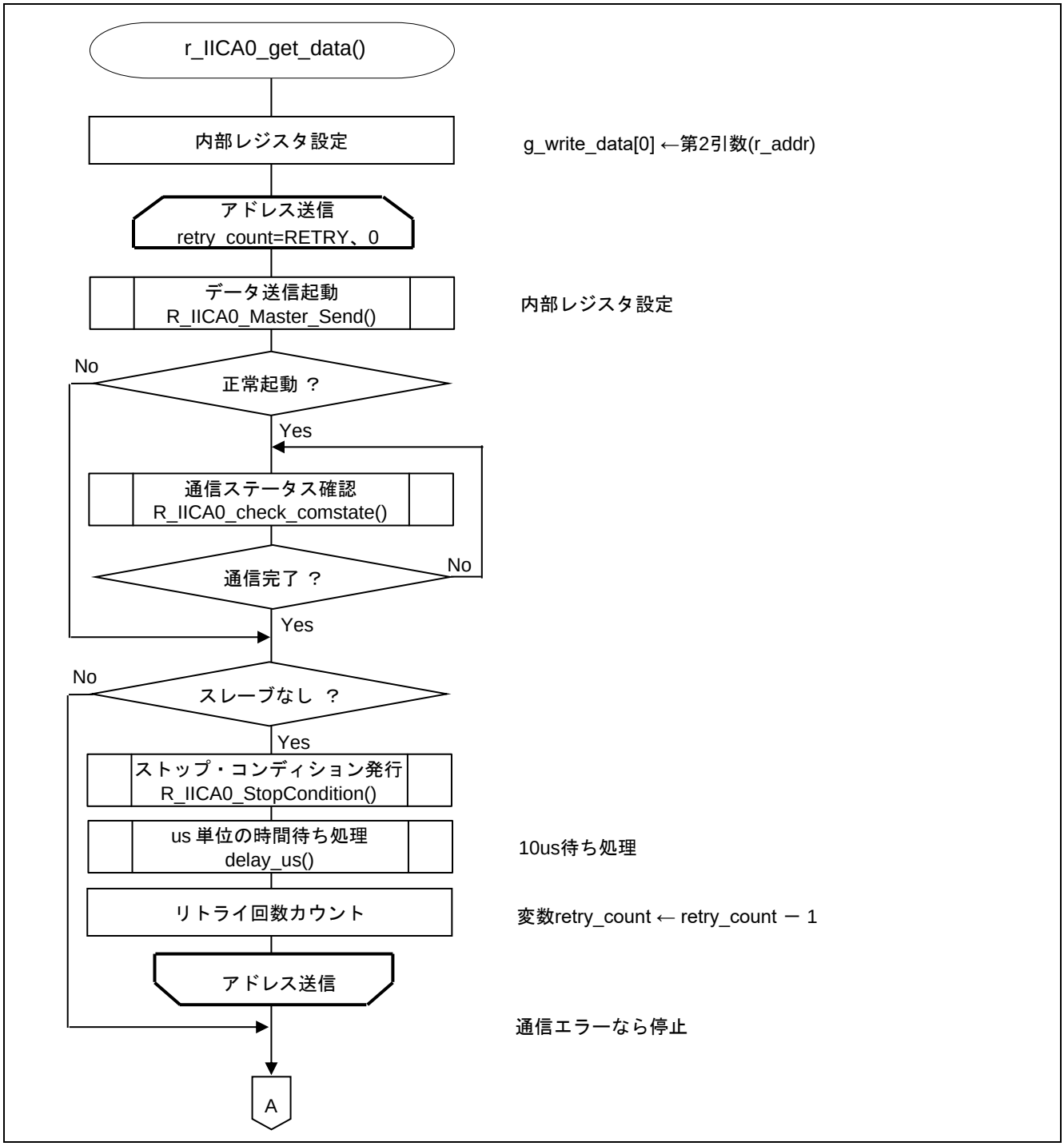
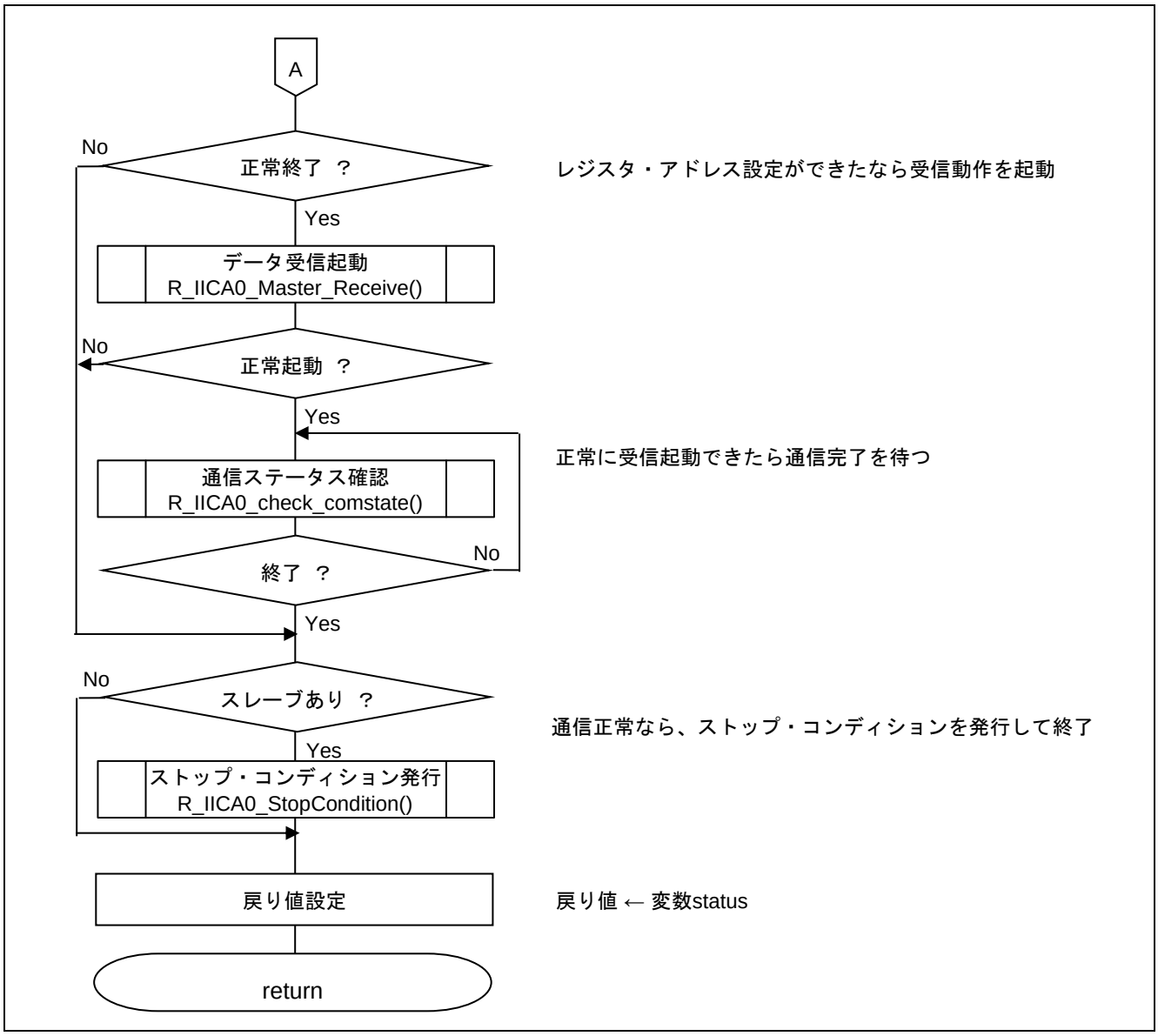


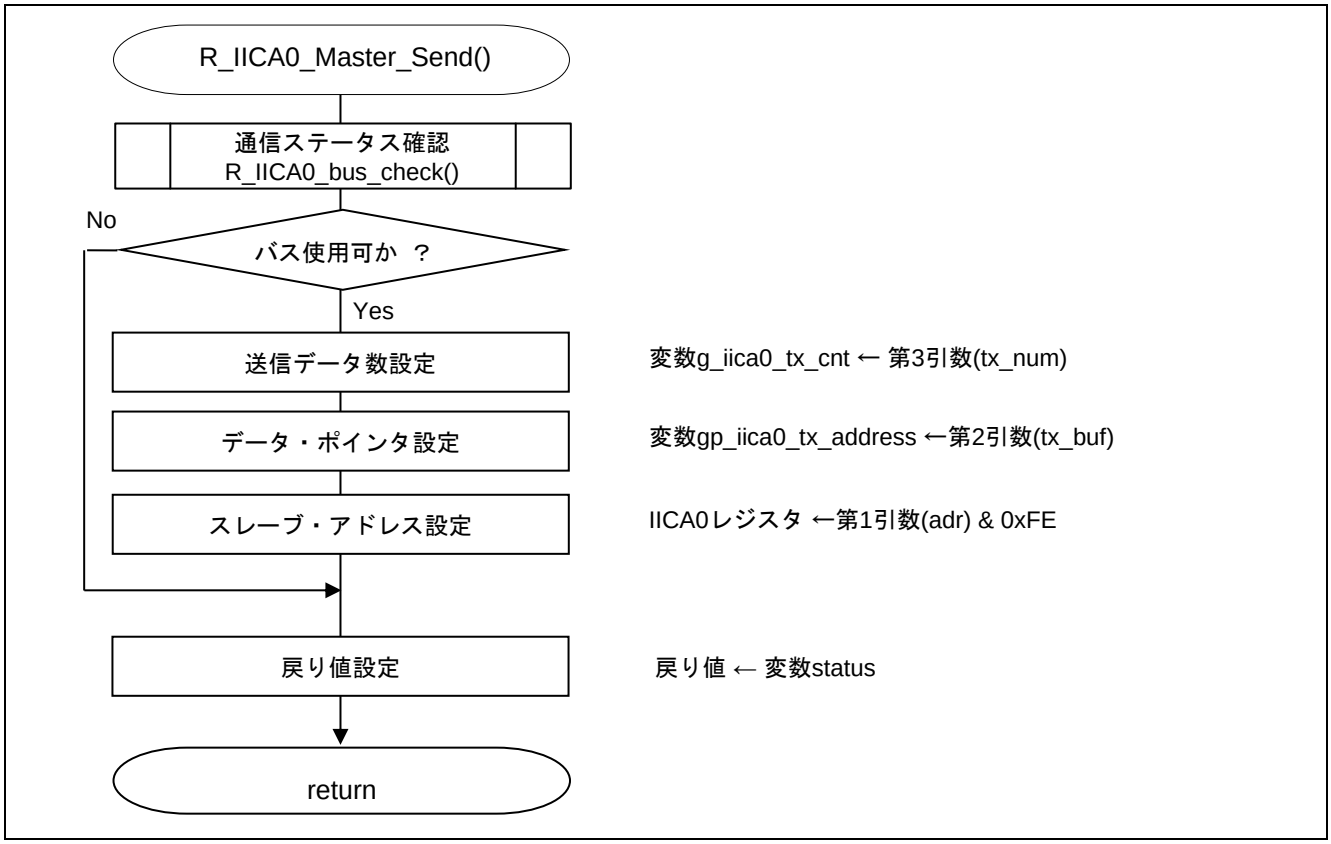
図 5.11　スレーブからのデータ受信処理（2/2）



5.7.8 IICA0 マスタ送信起動関数

図 5.12 に IICA0 マスタ送信起動関数のフローチャートを示します。

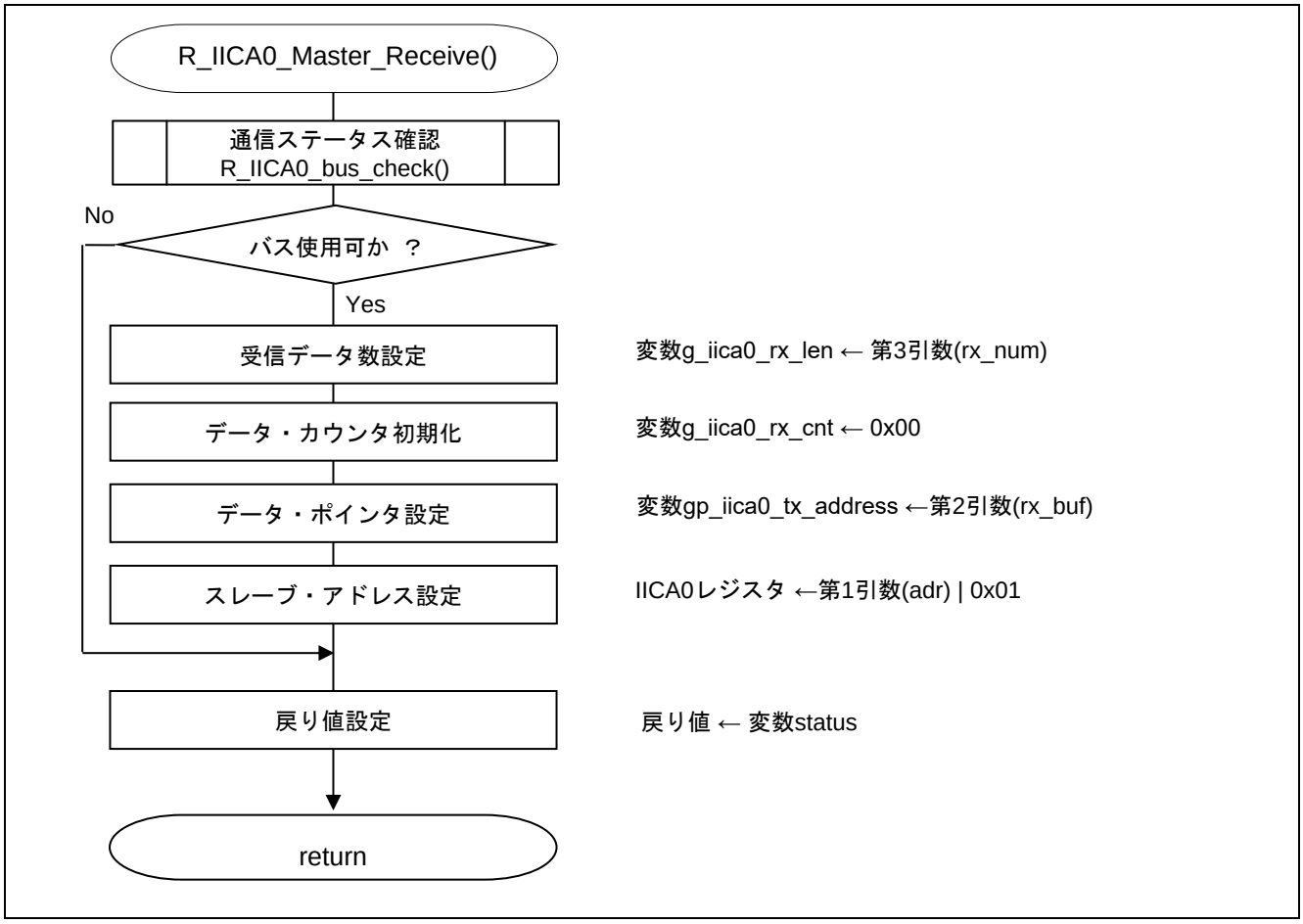
図 5.12 IICA0 マスタ送信起動関数



5.7.9 IICA0 マスタ受信起動関数

図 5.13 に IICA0 マスタ受信起動関数のフローチャートを示します。

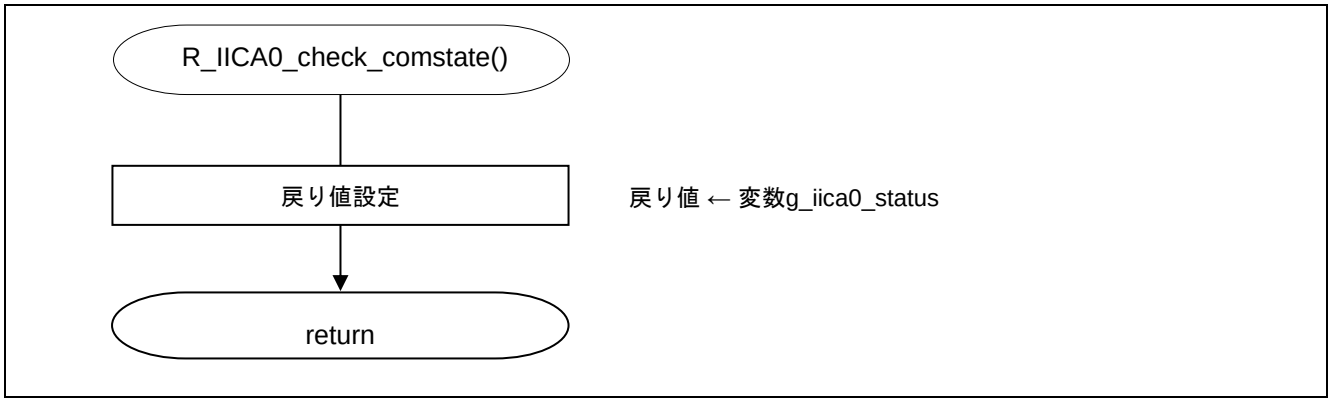
図 5.13 IICA0 マスタ受信起動関数



5.7.10 IICA0 通信状況確認関数

図 5.14 に IICA0 通信状況確認関数のフローチャートを示します。

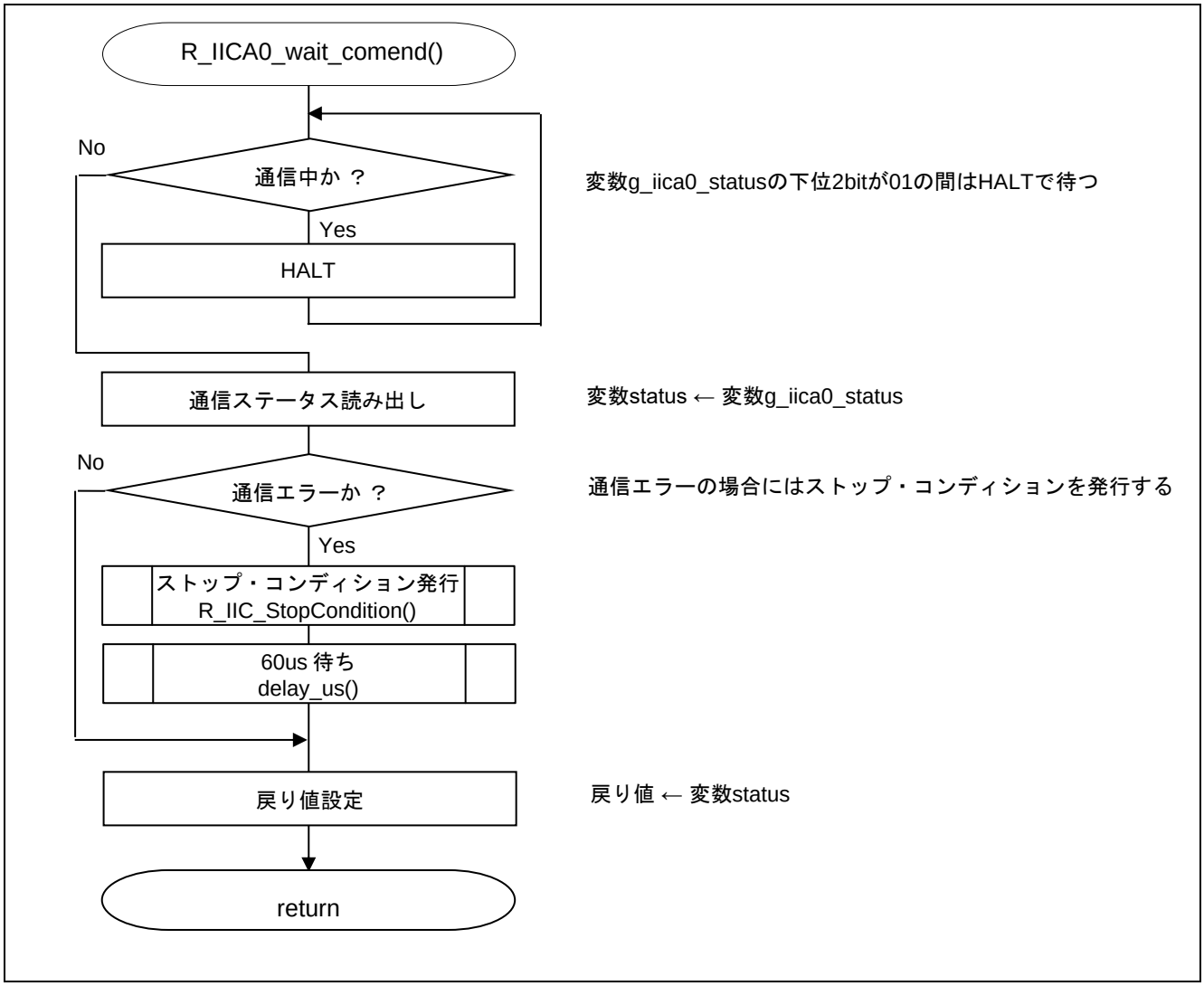
図 5.14 IICA0 通信状況確認関数



5.7.11 通信完了待ち関数

図 5.15 に通信完了待ち関数のフローチャートを示します。

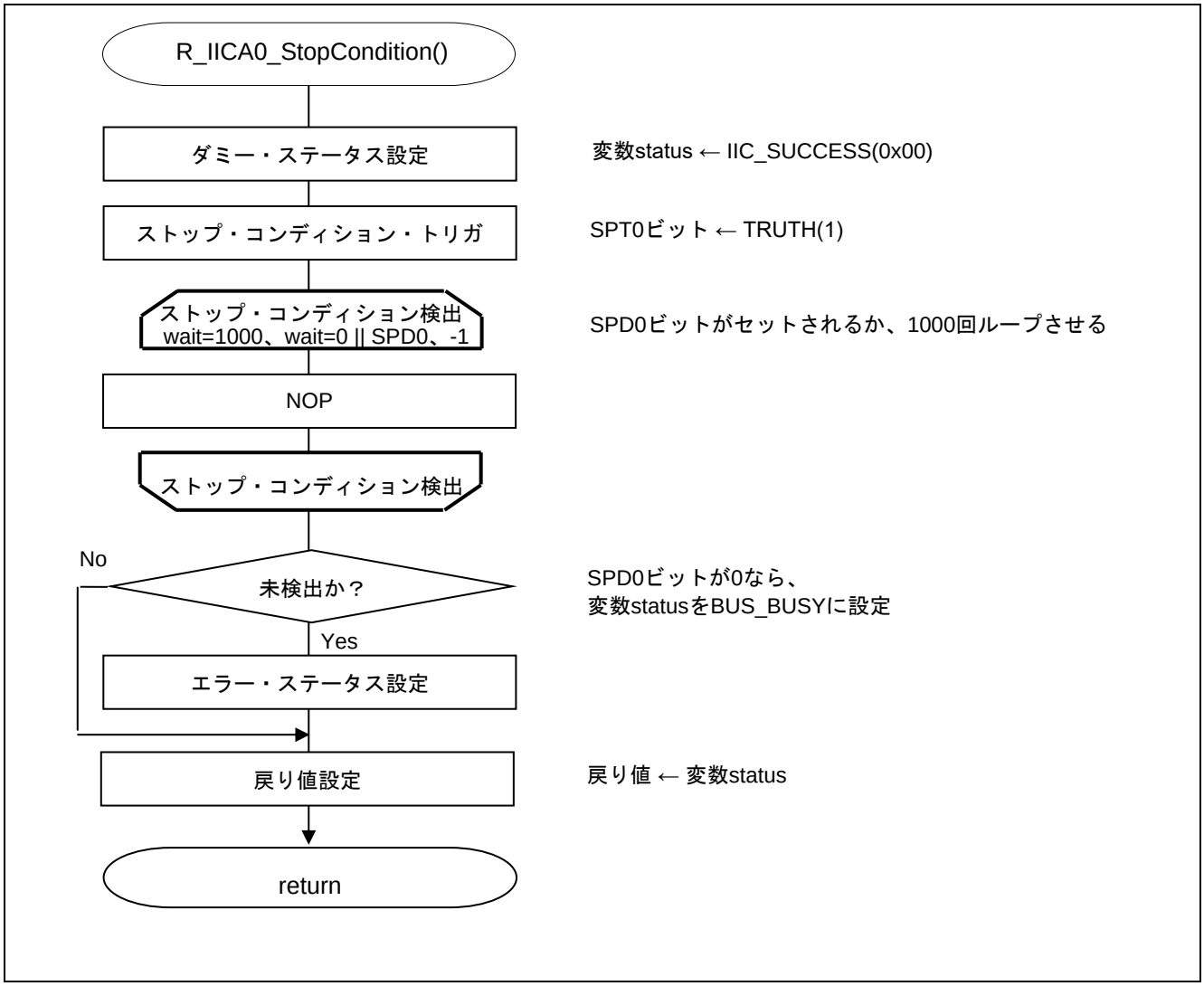
図 5.15　通信完了待ち関数



5.7.12 ストップ・コンディション発行処理

図 5.16 にストップ・コンディション発行処理のフローチャートを示します。

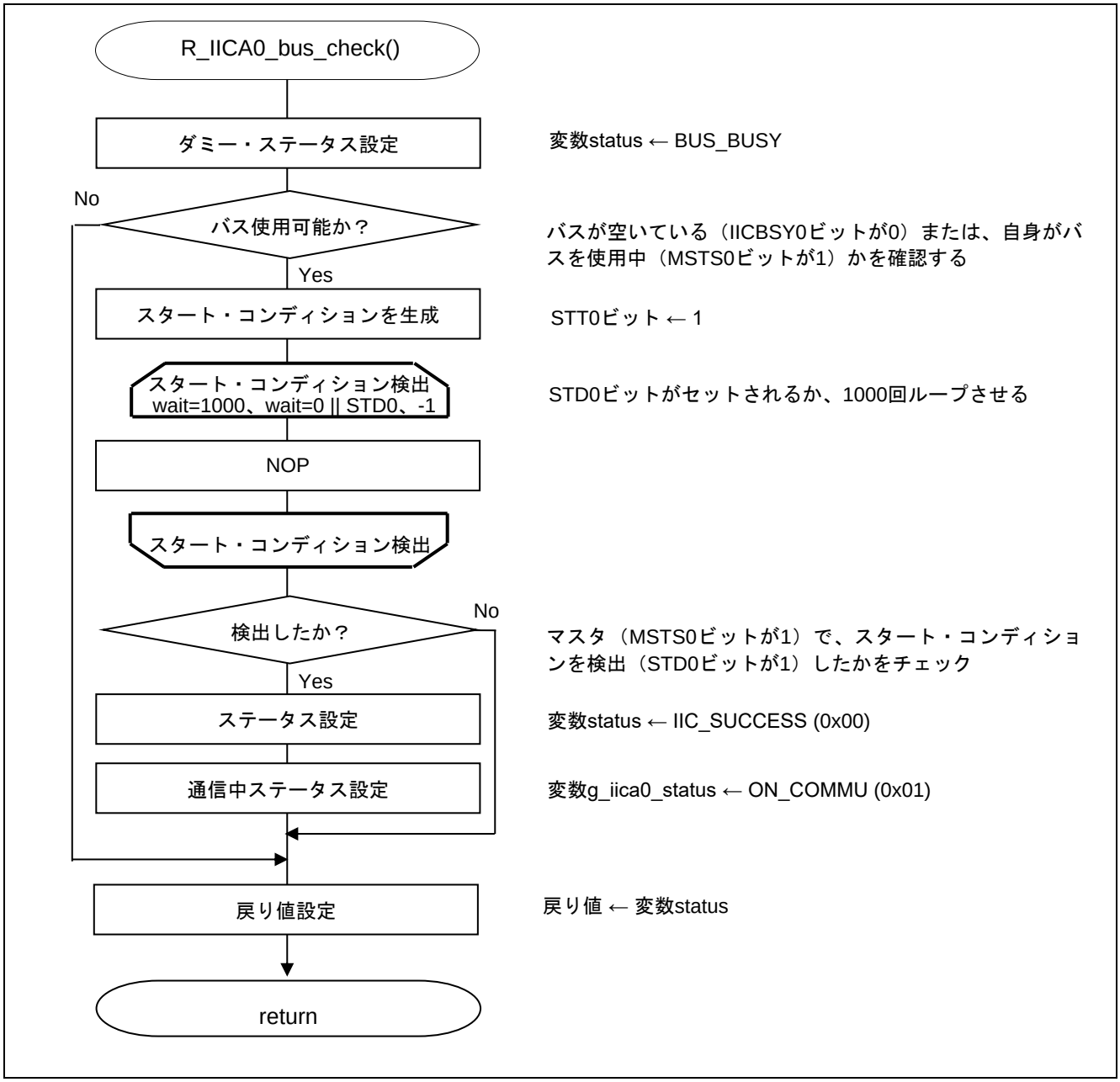
図 5.16　ストップ・コンディション発行処理



5.7.13 バス状態確認関数

図 5.17 にバス状態確認関数のフローチャートを示します。

図 5.17　バス状態確認関数



5.7.14 IICA0 マスタ通信処理

図 5.18～図 5.20 に IICA0 マスタ通信処理のフローチャートを示します。

図 5.18　IICA0 マスタ通信処理（1/3）

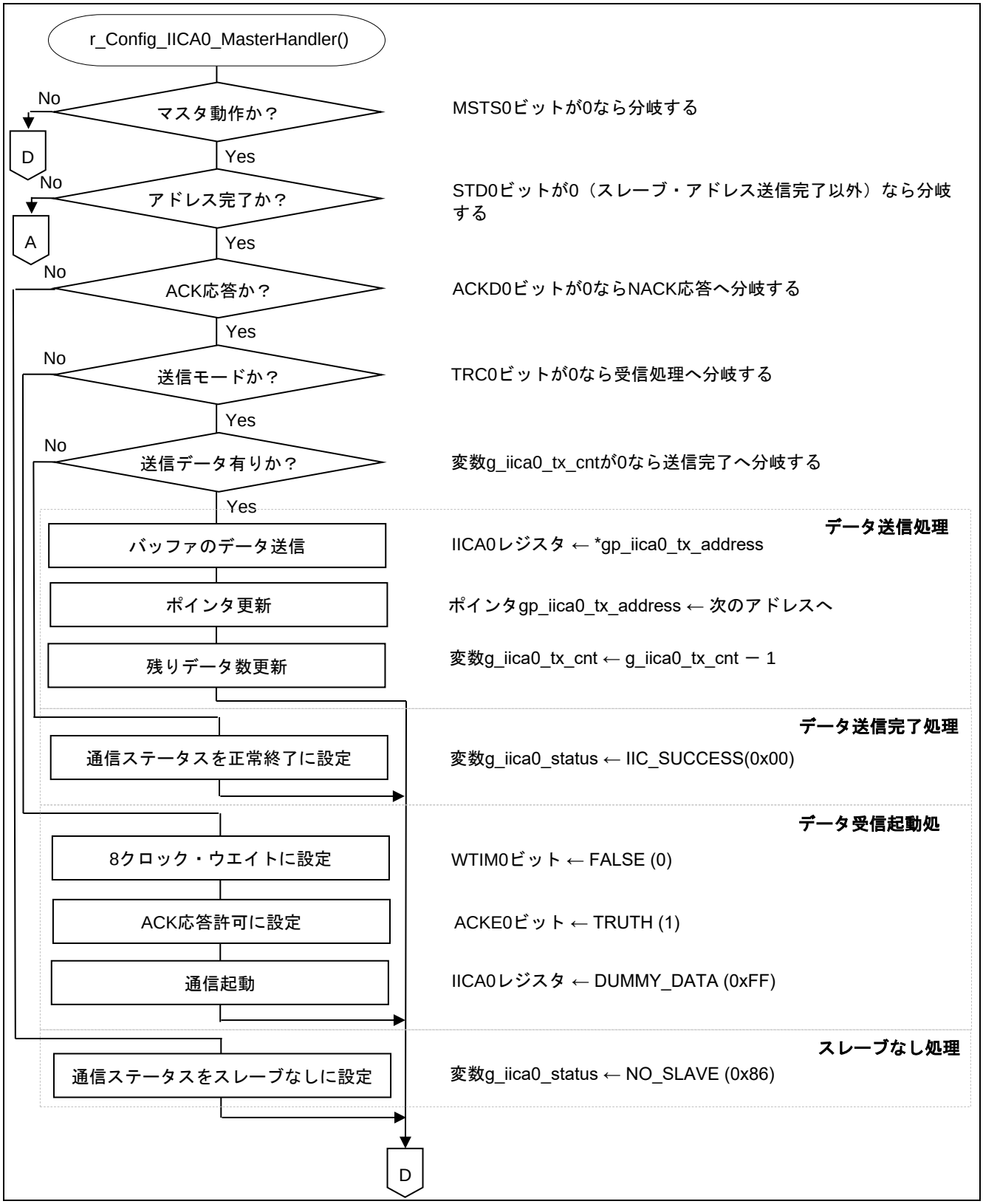


図 5.19　IICA0 マスタ通信処理（2/3）

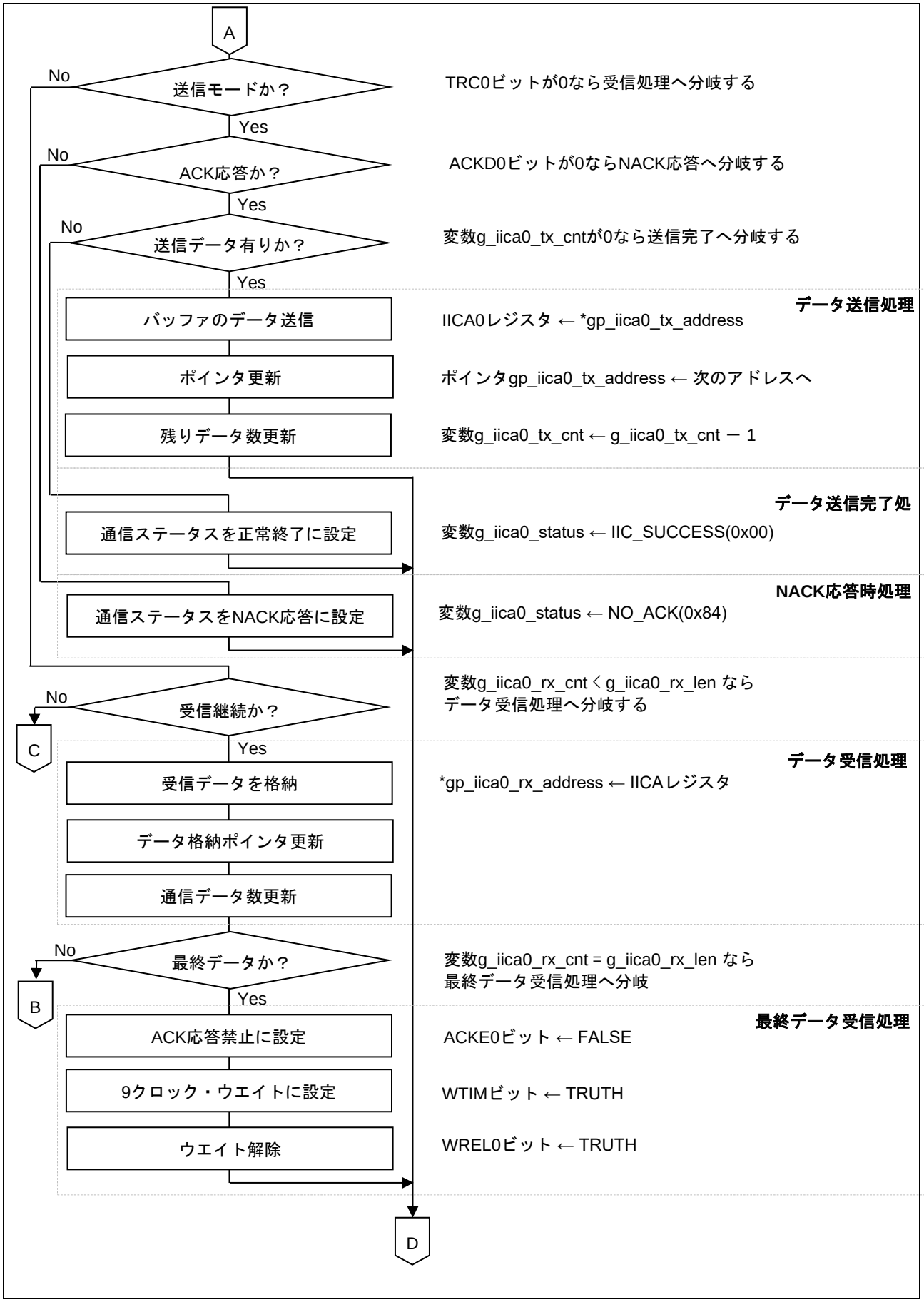
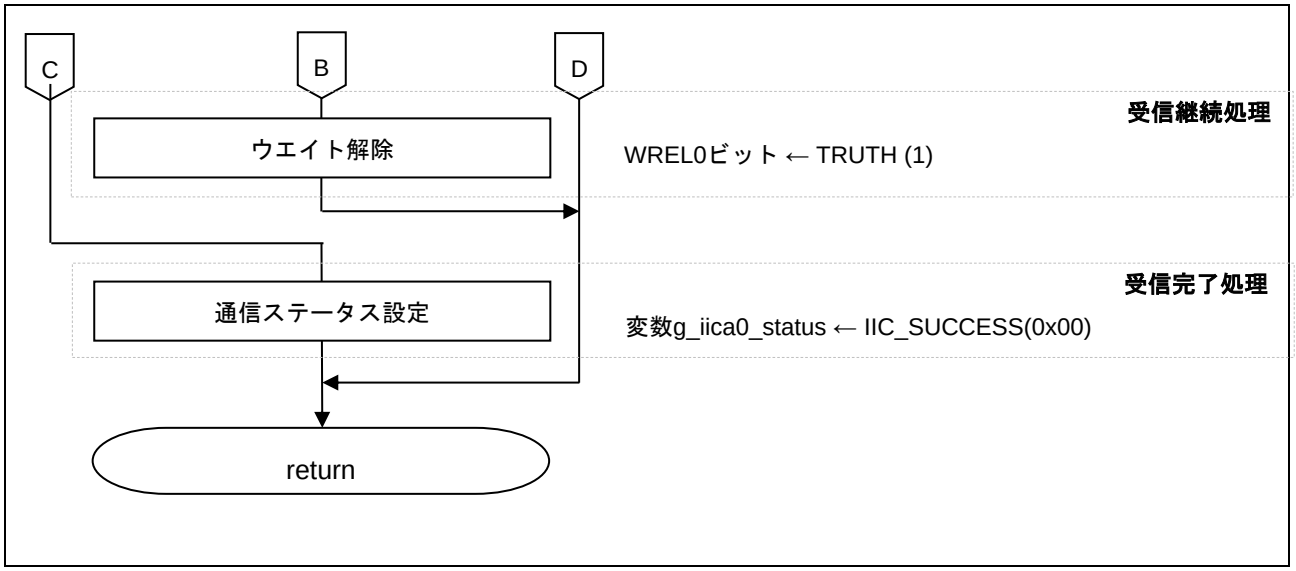


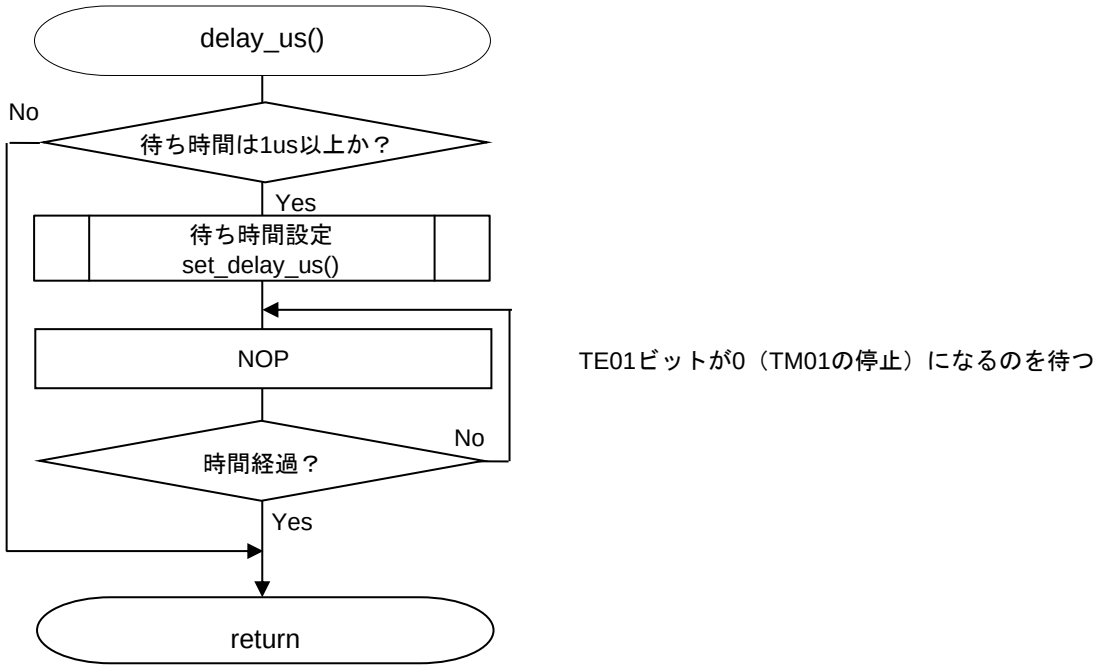
図 5.20　IICA0 マスタ通信処理（3/3）



5.7.15 us 単位での時間待ち処理

図 5.21 に us 単位での時間待ち処理のフローチャートを示します。

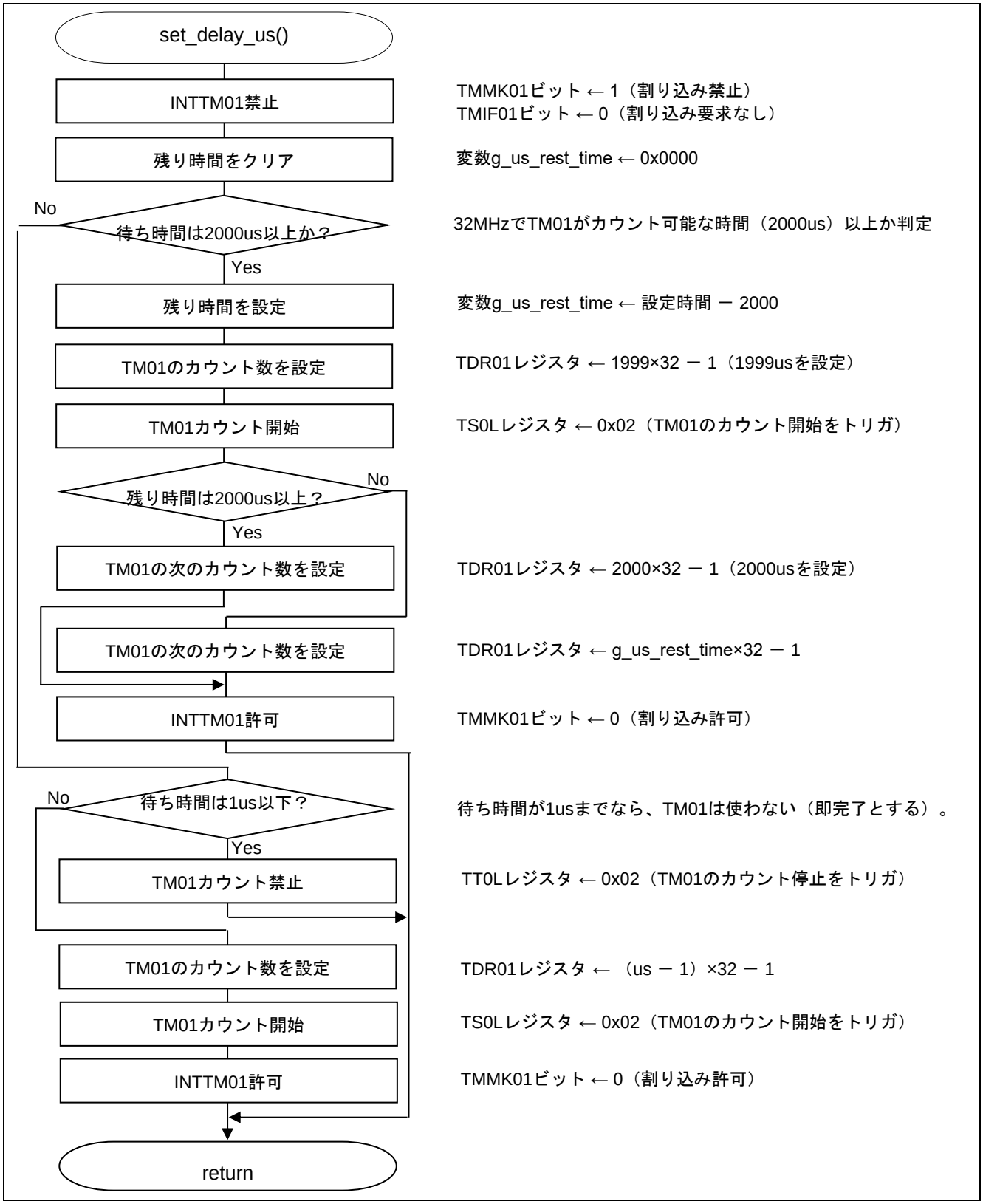
図 5.21　us 単位での時間待ち処理



5.7.16 us 単位での待ち時間設定処理

図 5.22 に us 単位での待ち時間設定処理のフローチャートを示します。

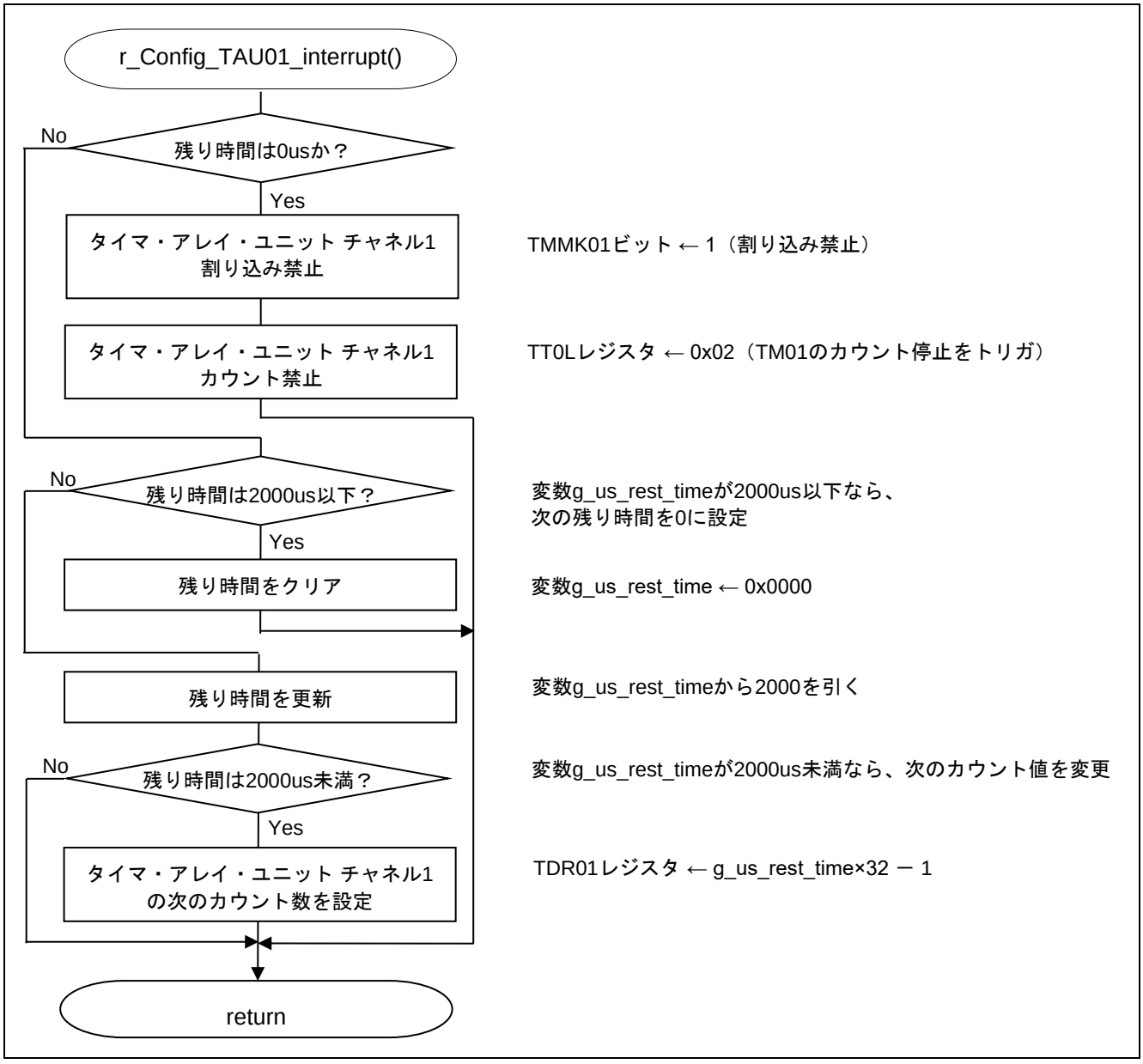
図 5.22　us 単位での待ち時間設定処理



5.7.17 タイマ・アレイ・ユニット チャネル 1 割り込み処理

図 5.23 にタイマ・アレイ・ユニット チャネル 1 割り込み処理のフローチャートを示します。

図 5.23　タイマ・アレイ・ユニット チャネル 1 割り込み処理



5.7.18 IICA0 割り込み処理

図 5.24 に IICA0 割り込み処理のフローチャートを示します。

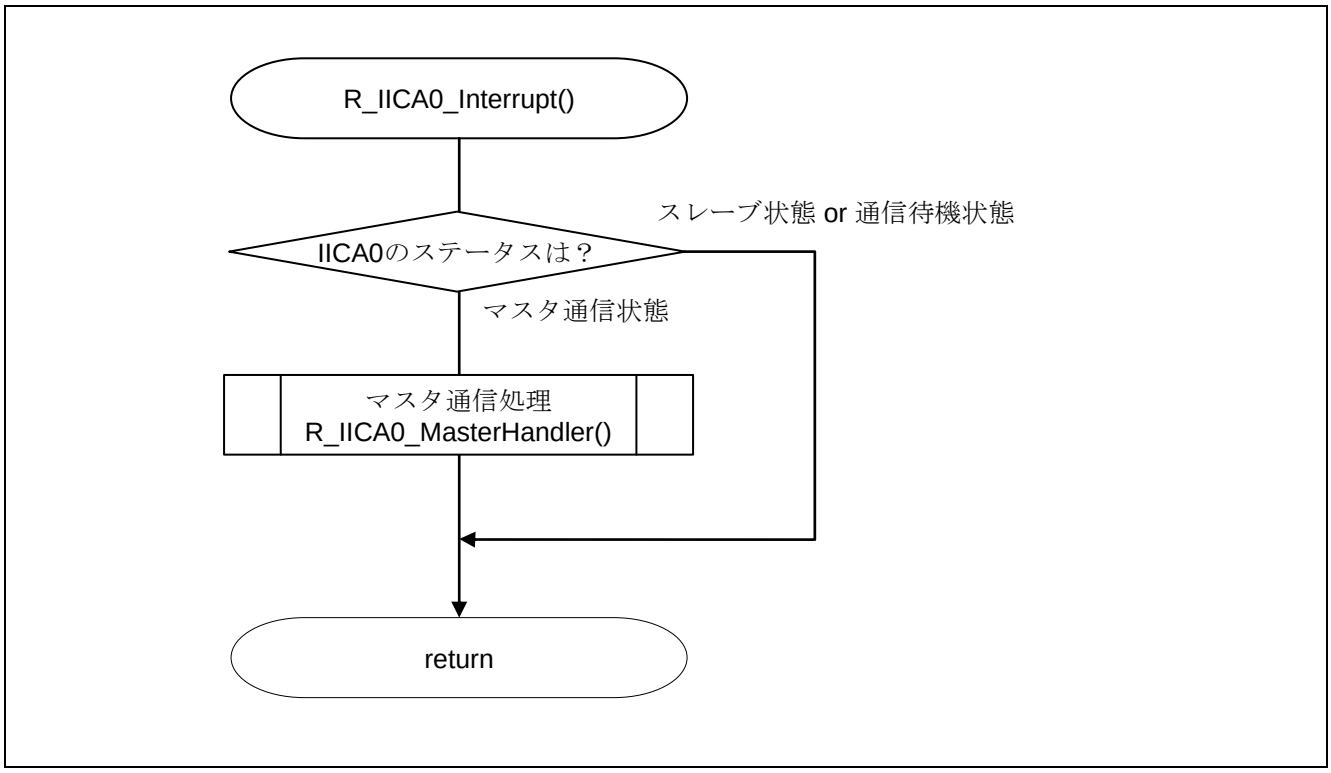


図 5.24 IICA0 割り込み処理

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

RL78/G22 ユーザーズマニュアル ハードウェア編（R01UH0978J）

RL78 ファミリ ユーザーズマニュアル ソフトウェア編（R01US0015J）

（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。）

すべての商標および登録商標は，それぞれの所有者に帰属します。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	June.28.24	—	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. 静電気対策
   CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。
2. 電源投入時の処置
   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. 電源オフ時における入力信号
   当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。
4. 未使用端子の処理
   未使用端子は、「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。
5. クロックについて
   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。
6. 入力端子の印加波形
   入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、V_{IL}（Max.）から V_{IH}（Min.）までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、V_{IL}（Max.）から V_{IH}（Min.）までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。
7. リザーブアドレス（予約領域）のアクセス禁止
   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、将来の拡張機能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。
8. 製品間の相違について
   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、フラッシュメモリ、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、ノイズ幅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。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、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品または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またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 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. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング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。
6. 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）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
   当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、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（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因しまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他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。
10. 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。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. 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。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
13. 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。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

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直接的、間接的に支配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。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。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。

© 2023 Renesas Electronics Corporation. All rights reserved.