

## RL78/G15

### ハンドシェイク対応 SPI 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・アレイ・ユニット (SAU) による簡易 SPI (CSI) のマスタ送受信の使用方法を説明します。チップセレクト ( $\overline{CS}$ ) 信号を使用し、1つのスレーブを選択してシングル送受信を行います。さらに、BUSY 信号を使用し、ハンドシェイクを行います。

#### 動作確認デバイス

RL78/G15

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 CSI 通信の概要 .....              | 3  |
| 1.2 通信の概要 .....                  | 4  |
| 1.3 通信フォーマット .....               | 6  |
| 1.4 ハンドシェイク .....                | 6  |
| 1.5 仕様詳細 .....                   | 7  |
| 2. 動作確認条件 .....                  | 10 |
| 3. 関連アプリケーションノート .....           | 10 |
| 4. ハードウェア説明 .....                | 11 |
| 4.1 ハードウェア構成例 .....              | 11 |
| 4.2 使用端子一覧 .....                 | 12 |
| 5. ソフトウェア説明 .....                | 13 |
| 5.1 オプション・バイトの設定一覧 .....         | 13 |
| 5.2 定数一覧 .....                   | 13 |
| 5.3 変数一覧 .....                   | 14 |
| 5.4 関数一覧 .....                   | 15 |
| 5.5 関数仕様一覧 .....                 | 15 |
| 5.6 フローチャート .....                | 17 |
| 5.6.1 メイン処理フローチャート .....         | 17 |
| 5.6.2 CSI ステータス確認処理フローチャート ..... | 19 |
| 5.6.3 CSI 送受信処理フローチャート .....     | 20 |
| 5.6.4 CSI 送信処理フローチャート .....      | 21 |
| 5.6.5 CSI 受信処理フローチャート .....      | 22 |
| 5.6.6 スレーブ応答待ち処理フローチャート .....    | 23 |
| 6. サンプルコード .....                 | 24 |
| 7. 参考ドキュメント .....                | 24 |
| 改訂記録 .....                       | 25 |

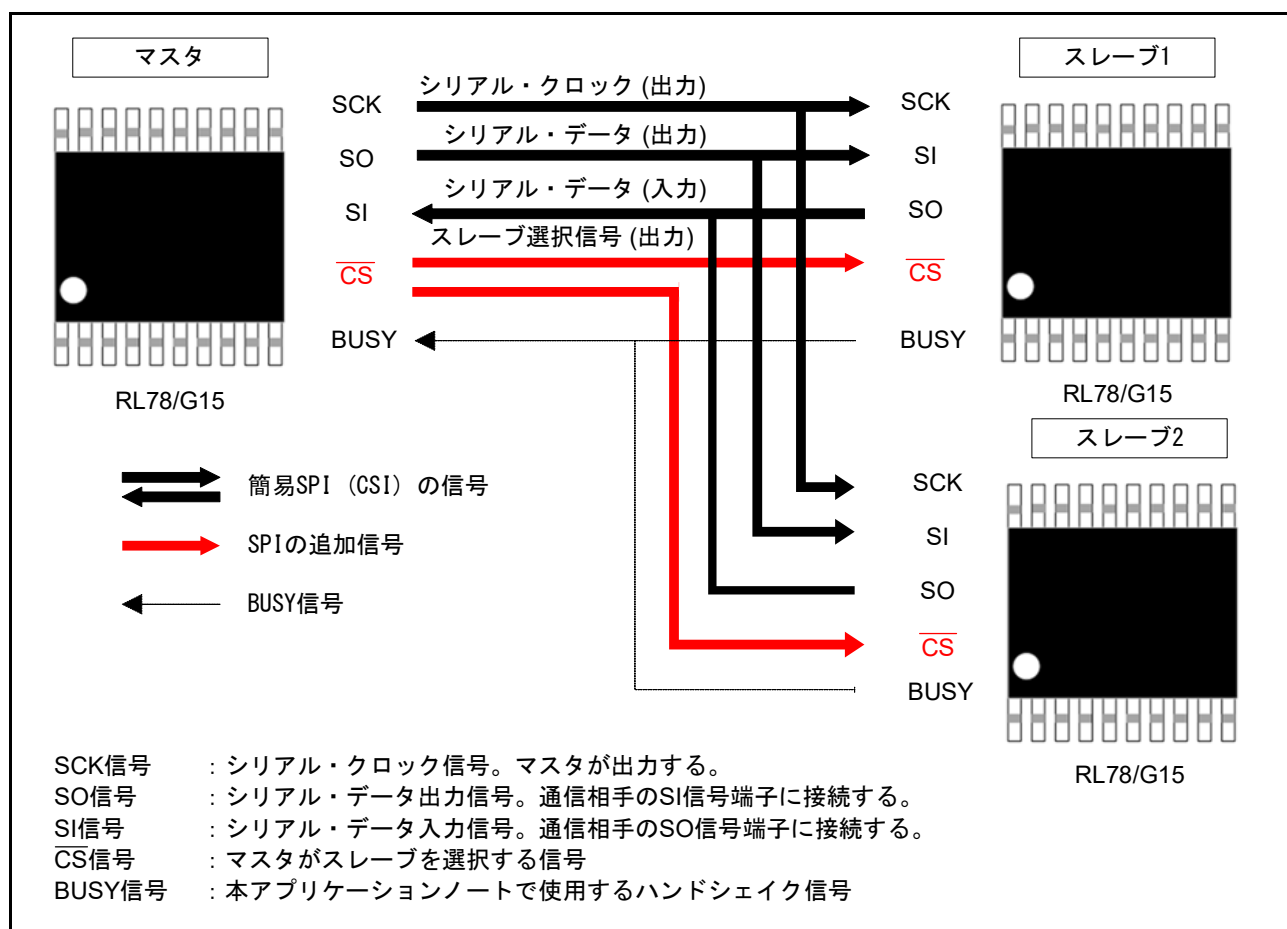
## 1. 仕様

本アプリケーションノートでは、シリアル・アレイ・ユニット (SAU) による CSI マスタ送受信を行います。チップセレクト ( $\overline{CS}$ ) 信号はポートを利用します。 $\overline{CS}$  信号で選択したスレーブに対して BUSY 信号によるハンドシェイクも行います。

### 1.1 CSI 通信の概要

CSI 通信はシリアル・クロック (SCK)、シリアル・データ入力 (SI)、シリアル・データ出力 (SO) の 3 本の信号を用いたクロック同期式シリアル通信です。Serial Peripheral Interface (SPI) には、スレーブを選択するチップセレクト ( $\overline{CS}$ ) 信号が追加されます。信号の関係を図 1-1 に示します。

図 1-1 CSI 通信の概要



最初に、マスタは通信したいスレーブを  $\overline{CS}$  信号で選択します。次に、マスタは SCK 信号および SCK 信号に同期した SO 信号にデータを出し、SI 信号のデータを入力します。

SPI / CSI 通信では、マスタが通信を開始 (SCK 出力) するまでにスレーブは通信準備ができている必要があります。本アプリケーションノートでは、BUSY 信号を使用してスレーブの通信準備ができたことを確認します。BUSY 信号のロウ・レベルを検出し、マスタは通信を開始します。

1.2 通信の概要

本アプリケーションノートでは、1ms 毎にコマンドとそのコマンドに対応した通信を行います。コマンドとそのコマンドに対応した 1 組の通信をスロットと定義します。スロットの概要を図 1-2 に示し、使用するコマンドを

表 1-1 に示します。

図 1-2 スロットの概要

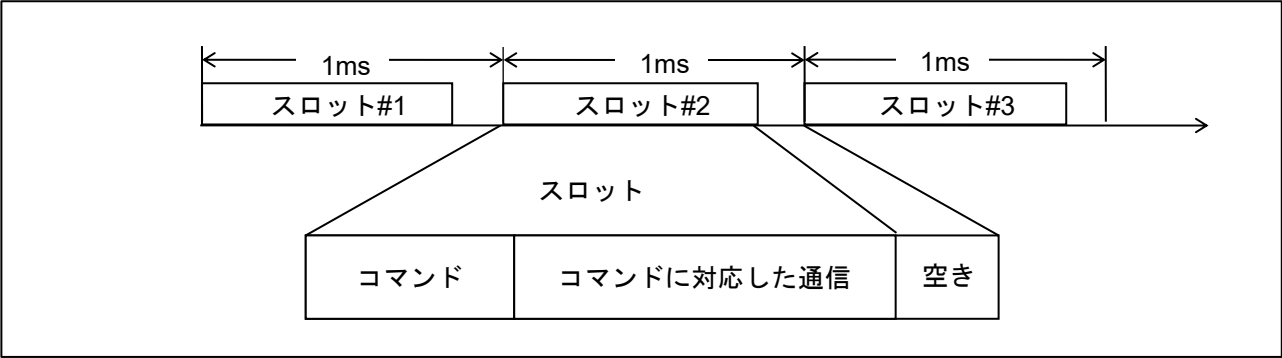


表 1-1 使用するコマンド

| コマンド    | コマンドの動作概要            |
|---------|----------------------|
| ステータス確認 | スレーブが送受信可能なデータ数を確認する |
| 受信      | スレーブからデータを受信する       |
| 送信      | スレーブへデータを送信する        |
| 送受信     | スレーブとデータを送受信する       |

表 1-2 に使用する周辺機能と用途を示し、図 1-3 エラー! 参照元が見つかりません。、図 1-4 に CSI の通信動作を示します。

表 1-2 使用する周辺機能と用途

| 周辺機能                      | 用途                                                                                      |
|---------------------------|-----------------------------------------------------------------------------------------|
| シリアル・アレイ・ユニット 0           | SCK00 信号 (クロック出力)、SI00 信号 (受信データ) と SO00 信号 (送信データ) を使用し、CSI マスタ通信を行う。                  |
| ポート                       | P02 を $\overline{\text{CS1}}$ 信号、P03 を $\overline{\text{CS2}}$ 信号、P07 を BUSY 信号として使用する。 |
| タイマ・アレイ・ユニット 0<br>チャンネル 1 | 上位 8 ビットは、1ms のインターバル・タイマとして使用する。<br>下位 8 ビットは、16 $\mu$ s のインターバル・タイマとして使用する。           |

図 1-3 ステータス確認コマンドのタイミング・チャート

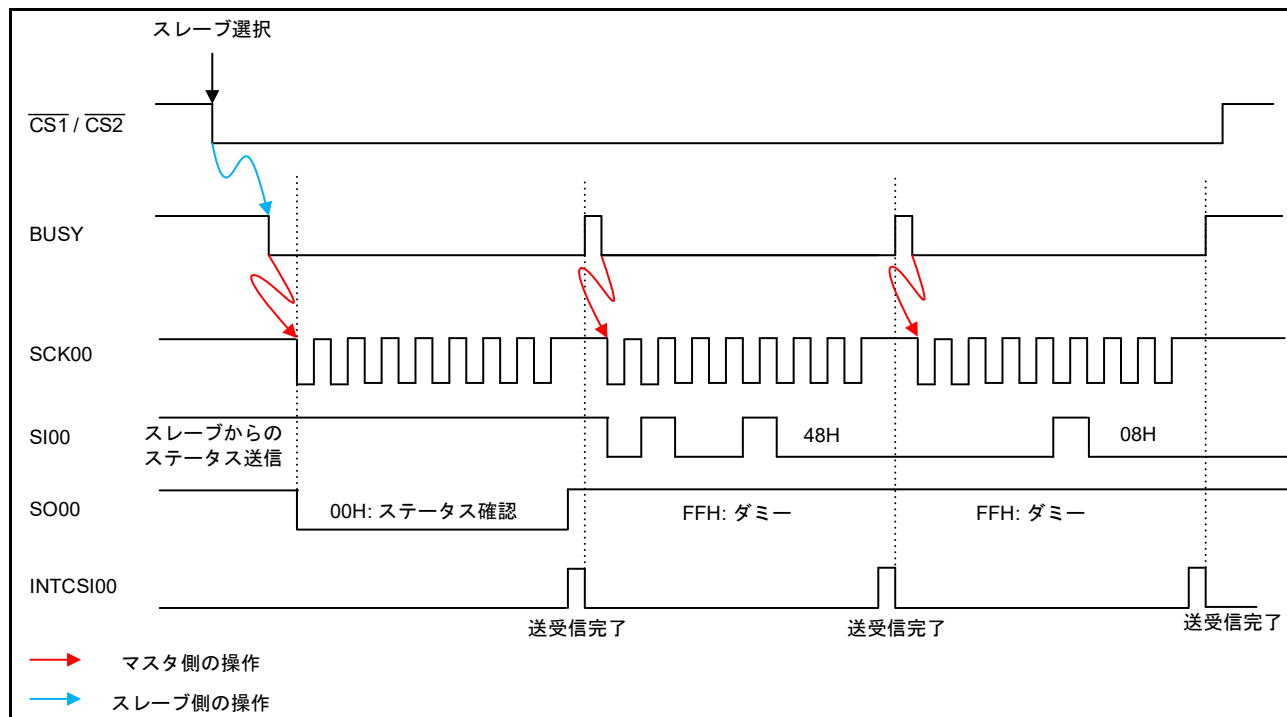
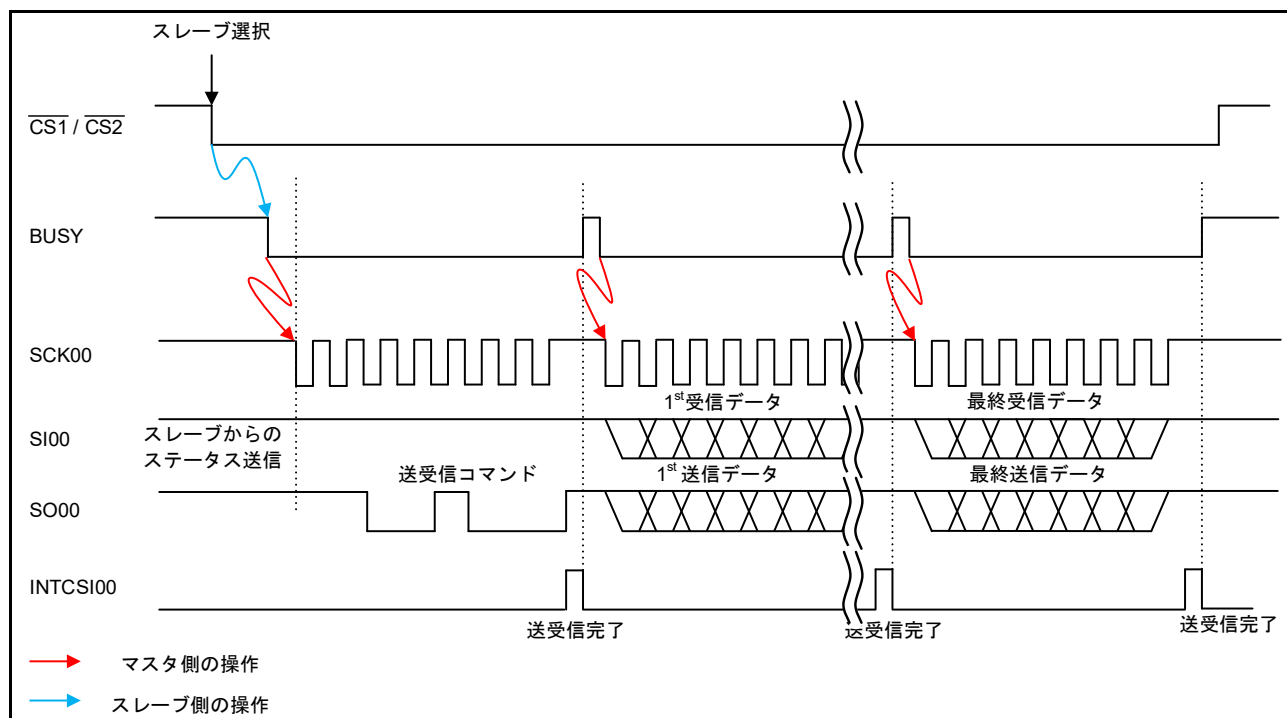


図 1-4 送受信コマンドのタイミング・チャート



### 1.3 通信フォーマット

サンプルコードで使用する CSI の通信フォーマットを表 1-3 に示します。

表 1-3 通信フォーマット

| 項目         | 規格           | 備考             |
|------------|--------------|----------------|
| 通信速度       | 1 Mbps       | 最低速度は約200 kbps |
| 通信データのビット長 | 8ビット／キャラクタ   |                |
| 転送順序       | MSBファースト     |                |
| 通信タイプ      | タイプ1         |                |
| 通信モード      | シングル転送       |                |
| 通信方向       | 受信／送信／送受信    |                |
| 最大転送データ数   | 63キャラクタ／スロット | デフォルトでは32キャラクタ |

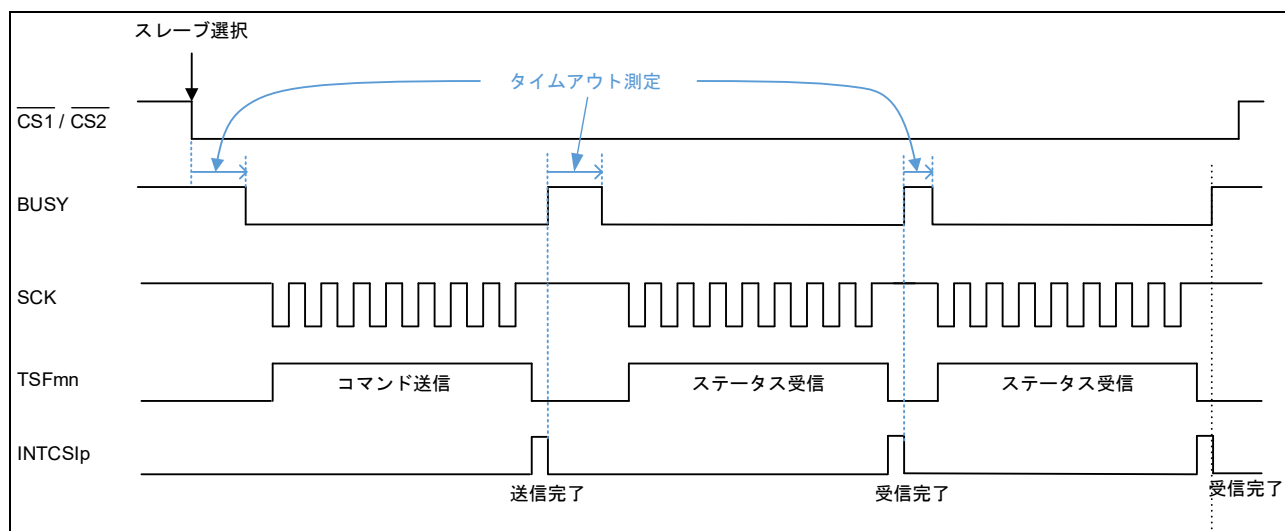
### 1.4 ハンドシェイク

本アプリケーションノートでは、スレーブの通信準備時間を確保するために、BUSY 信号を利用したハンドシェイクを行います。スレーブの BUSY 信号による応答がない場合に備え、16  $\mu$ s のタイムアウト時間を設定しています。タイムアウト時間内にスレーブからの応答がない場合は、スレーブが通信できない状態である、もしくはスレーブが存在しないと判断し、通信処理を中止します。

ステータス確認時のハンドシェイク例を図 1-5 に示します。 $\overline{CS}$  信号を立ち下げてから BUSY 信号がロウ・レベルになるまでの時間を計測します。タイムアウト時間内で、BUSY 信号がロウ・レベルであることを検出した場合、コマンドを送信します。転送完了割り込みでコマンド送信の完了を確認した後、再び、BUSY 信号がロウ・レベルになるまでの時間を計測します。このように、通信を開始する毎に BUSY 信号の確認することでスレーブと同期をとります。

ただし、スレーブが連続受信に対応している場合やスレーブが通信データ準備時間を十分に確保できる場合は、BUSY 信号による同期は必要ありません。

図 1-5 ハンドシェイク例



通常、SPI 通信専用スレーブデバイス (EEPROM 他) は、常に通信ができる状態ため、BUSY 信号は不要です。SPI 通信専用スレーブデバイスと接続する場合は、BUSY 信号入力端子をプルダウンしてください。さらに、プログラムの wait() 関数を取り除くことで、タイムアウト処理を削除できます。その場合、SPI 通信専用スレーブデバイスで規定されたコマンドと通信手順に従って通信を行ってください。

## 1.5 仕様詳細

本サンプルコードでは、初期設定完了後スレーブを選択し、選択されたスレーブに対してステータス確認を行います。ステータス確認後データの送受信を行い、スレーブを切り替え同様の動作を繰り返します。

(1) ポートの初期設定を行います。

＜ポート設定条件＞

- $\overline{\text{CS1}}$  信号を制御する P02 は出力で使用し、ハイ・レベルを出力します。
- $\overline{\text{CS2}}$  信号を制御する P03 は出力で使用し、ハイ・レベルを出力します。
- BUSY 信号を検出する P07 は入力で使用します。

(2) タイマの初期設定を行います。

＜タイマ設定条件＞

- チャネル 1 を 2 つの 8 ビット・インターバル・タイマとして使用します。
- 動作クロックは fCLK を 256 分周した 62.5 kHz を使用します。
- 上位の TM01H は 1ms のインターバル・タイマとして使用します。
- 下位の TM01 は 16 $\mu$ s のインターバル・タイマとして使用します。
- 割り込み (INTTM01H、INTTM01) はデフォルトの低優先 (レベル 3) を使用します。

(3) CSI の初期設定を行います。

＜CSI 設定条件＞

- SAU0 チャネル 0 を CSI00 として使用します。
- 転送クロックは CK00 使用します。
- クロック出力は P06/SCK00 端子、データ入力は P05/SI00 端子、データ出力は P04/SO00 端子を使用します。
- 転送モードはシングル転送モードを使用します。
- データ長は 8 ビットを使用します。
- データとクロックの位相はタイプ 1 を使用します。
- データ転送順設定は MSB ファーストを使用します。
- 転送レートは 1Mbps を使用します。
- 割り込み (INTCSI00) は送信完了、受信完了割り込みを使用します。
- 割り込み (INTCSI00) はデフォルトの低優先 (レベル 3) を使用します。

(4) 初期設定が完了したら、以下に示す手順でスレーブとの通信動作を行います。

- ① 1ms のインターバル・タイマ割り込み (INTTM01H) を HALT モードで待ちます。
- ② INTTM01 の発生で HALT モードが解除されたら、g\_slave\_select\_flag で指定されたスレーブを  $\overline{CS}$  信号で選択してスレーブの応答を待ちます。
- ③ BUSY 信号のロウ・レベルを検出した場合は④に移行します。タイムアウトを検出した場合はスレーブの選択を解除して⑨に移行します。
- ④ ステータス確認コマンドを送信し、スレーブのステータスを受信します。タイムアウトを検出した場合、またはスレーブのステータスを受信できない場合、スレーブの選択を解除して⑨に移行します。スレーブのステータス受信が完了すると、 $\overline{CS}$  信号の選択を解除します。
- ⑤ 再度、1ms のインターバル・タイマ割り込み (INTTM01H) を HALT モードで待ちます。
- ⑥ 再度、INTTM01 の発生で HALT モードを解除すると、g\_slave\_select\_flag で指定されたスレーブを  $\overline{CS}$  信号で選択してスレーブの応答を待ちます。
- ⑦ BUSY 信号のロウ・レベルを検出した場合、⑧に移行します。タイムアウトを検出した場合はスレーブの選択を解除して⑨に移行します。
- ⑧ ④で確認したステータスに合った数のデータを送受信します。タイムアウトを検出した場合はスレーブの選択を解除します。
- ⑨ g\_slave\_select\_flag を変更して、対象のスレーブを切り替えます。以降は①から繰り返し実行します。

#### (5) コマンド

各通信動作は 1 バイトのコマンドの送信から始まります。各コマンドのフォーマットを表 1-4 に示します。通信シーケンスの最初のスロットでステータス確認コマンドを送信して、スレーブからの応答を受信します。受信したスレーブが送信可能なデータ数、受信可能なデータ数が設定した送受信するデータ数以上であることを確認し、設定したデータ数分スレーブとの送受信を行います。

表 1-4 コマンドのフォーマット

| コマンド コード |           | コマンドの概要                                                                                                                      |
|----------|-----------|------------------------------------------------------------------------------------------------------------------------------|
| ステータス確認  | 00000000B | スレーブが送信可能なデータ数、受信可能なデータ数を確認する。スレーブからは以下の応答がある。<br>01xxxxxxB : スレーブが送信可能なデータ数は xxxxxxB<br>00xxxxxxB : スレーブが受信可能なデータ数は xxxxxxB |
| 受信       | 01xxxxxxB | マスタが xxxxxxB バイトのデータを受信する。                                                                                                   |
| 送信       | 10xxxxxxB | マスタが xxxxxxB バイトのデータを送信する。                                                                                                   |
| 送受信      | 11xxxxxxB | xxxxxxB バイトのデータを送受信する。                                                                                                       |



## (6) 通信コマンドの切り替え

main.c 内のコメントアウトを変更することにより、受信、送信、送受信のコマンドを切り替えることが可能です。デフォルト設定では送受信となっています。

```
/*-----  
Send Receive command  
-----*/  
  
CSI00_Send_Receive();           /*call Send Receive function*/  
/* CSI00_Send(); */  
/* CSI00_Receive(); */  
g_slave_select_flag ^= 1U;      /*change slave number*/  
g_status_confirmation_flag = 0; /* clear g_status_confirmation_flag */
```

## 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では、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/G15 (R5F12068ASP)                                                                                              |
| 使用ボード                          | RL78/G15 Fast Prototyping Board (RTK5RLG150C00000BJ)                                                                |
| 動作周波数                          | 高速オンチップ・オシレータ (HOCO) クロック : 16MHz<br>CPU/周辺ハードウェア・クロック : 16MHz                                                      |
| 動作電圧                           | 5.0V (2.4V~5.5V で動作可能)<br>SPOR 検出電圧<br>立ち上がり時 TYP. 2.57 V (2.44 V ~ 2.68 V)<br>立ち下がり時 TYP. 2.52 V (2.40 V ~ 2.62 V) |
| 統合開発環境 (CS+)                   | ルネサス エレクトロニクス製<br>CS+ for CC V8.11.00                                                                               |
| コンパイラ(CS+)                     | ルネサス エレクトロニクス製<br>CC-RL V1.13.00                                                                                    |
| 統合開発環境 (e <sup>2</sup> studio) | ルネサス エレクトロニクス製<br>e <sup>2</sup> studio V2024-01 (24.1.0)                                                           |
| コンパイラ(e <sup>2</sup> studio)   | ルネサス エレクトロニクス製<br>CC-RL V1.13.00                                                                                    |
| 統合開発環境 (IAR)                   | IAR Systems 製<br>IAR Embedded Workbench for Renesas RL78 V5.10.1                                                    |
| 統合開発環境 (IAR)                   | IAR Systems 製<br>IAR C/C++ Compiler for Renesas RL78 V5.10.1                                                        |
| スマート・コンフィグレータ                  | V.1.9.0                                                                                                             |
| ボードサポートパッケージ<br>(r_bsp)        | V.1.62                                                                                                              |

## 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。

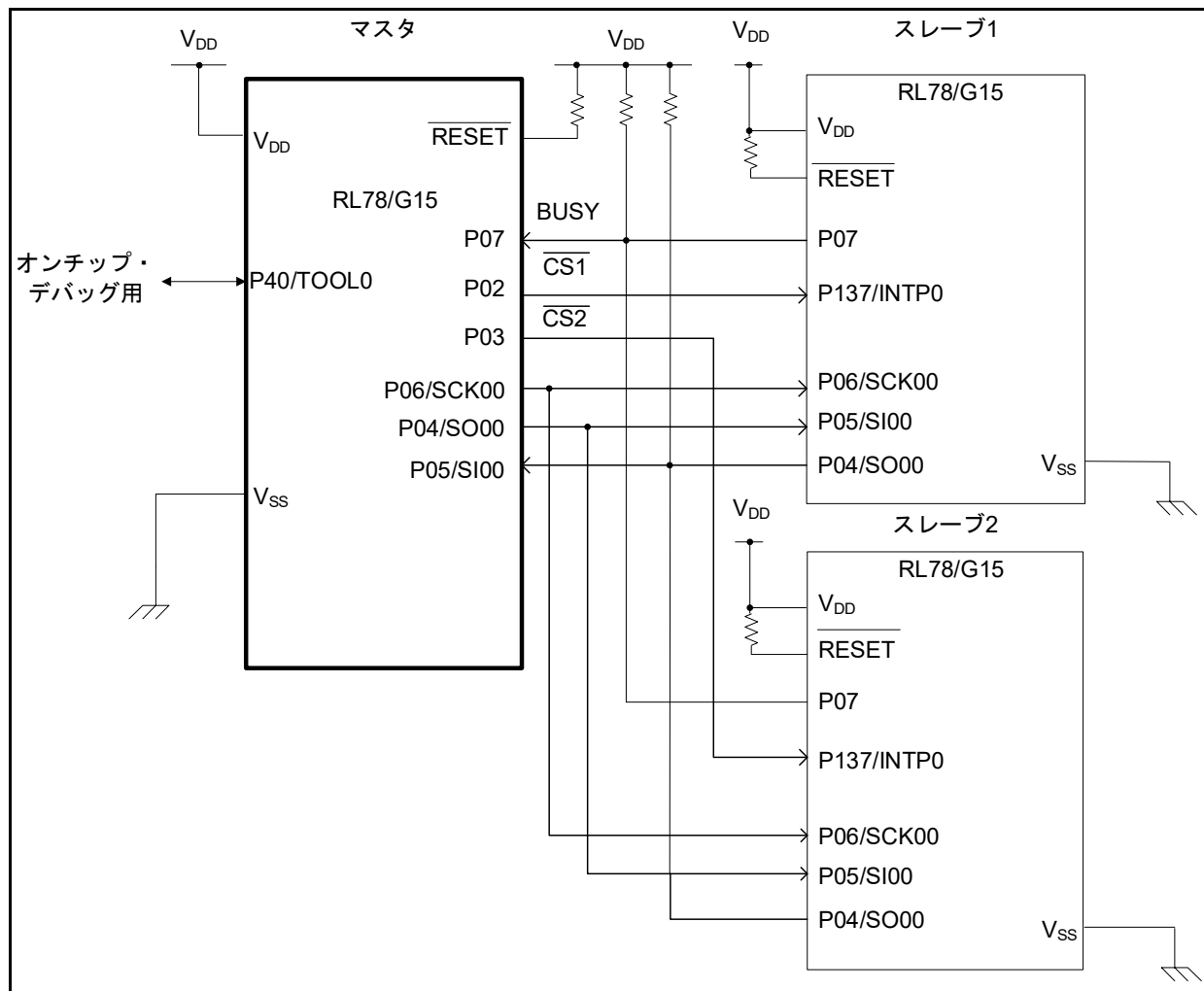
RL78/G15 ハンドシェイク対応 SPI スレーブ送受信 (R01AN7296J) アプリケーションノート

## 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 ハードウェア構成例



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 SPOR にて設定したリセット解除電圧 ( $V_{SPOR}$ ) 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                |
|-----------------------------------------------------------------|-----|-------------------|
| P06/ANI5/SI01/SDA01/SCLA0/(PCLBUZ0)/(INTP7)/(SCK00/SCL00)       | 出力  | シリアル・クロック出力用端子    |
| P05/ANI4/TI02/TO02/SO01/(INTP6)/(SCK00/SCL00)/(SI00/RxD0/SDA00) | 入力  | データ受信用端子          |
| P04/ANI3/IVREF0/INTP3/TI01/TO01/(SI00/RxD0/SDA00)/(SO00/TxD0)   | 出力  | データ送信用端子          |
| P07/ANI6/SCK01/SCL01/SDAA0/(INTP5)/(TO03)                       | 入力  | スレーブからの BUSY 信号入力 |
| P03/ANI2/IVCMP0/INTP4/TO00/(TI00)/(SO00/TxD0)                   | 出力  | スレーブ 2 選択信号       |
| P02/PCLBUZ0/ANI1/VCOUT0/INTP7/SCK00/SCL00/(TI01/TO01)/(SO01)    | 出力  | スレーブ 1 選択信号       |

注意 本アプリケーションノートは、使用端子のみを端子処理しています。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。

## 5. ソフトウェア説明

### 5.1 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 | 1110 1111B | ウォッチドッグ・タイマ 動作停止<br>(リセット解除後、カウント停止)                                                      |
| 000C1H | 1111 1011B | SPOR 検出電圧 :<br>立ち上がり時 TYP. 2.57V (2.44 V ~ 2.68 V)<br>立ち下がり時 TYP. 2.52V (2.40 V ~ 2.62 V) |
| 000C2H | 1111 1001B | 高速オンチップ・オシレータ・クロック : 16MHz                                                                |
| 000C3H | 1000 0101B | オンチップ・デバッグ許可                                                                              |

### 5.2 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         | 定義場所              | 設定値        | 内容                                      |
|-------------|-------------------|------------|-----------------------------------------|
| CS1_pin     | r_cg_userdefine.h | P0_bit.no2 | $\overline{\text{CS1}}$ 信号を制御するポート・レジスタ |
| CS2_pin     | r_cg_userdefine.h | P0_bit.no3 | $\overline{\text{CS2}}$ 信号を制御するポート・レジスタ |
| BUSYIN      | r_cg_userdefine.h | P0_bit.no7 | BUSY 信号を検出するポート・レジスタ                    |
| TX_NUM      | main.c            | 32         | 送信データ数                                  |
| RX_NUM      | main.c            | 32         | 受信データ数                                  |
| TX_RX_NUM   | main.c            | 32         | 送受信データ数                                 |
| data_length | main.c            | 1          | データ長                                    |
| MODE[]      | main.c            | 注 1        | 各コマンドのフォーマット                            |
| TX_DATA[]   | main.c            | 注 2        | 最大転送データ数の 63 キャラクタ分の送信データを格納            |

注 1 詳細は表 1-4 を参照してください。

2 本アプリケーションノートでは 0x20~0x5F までの ASCII コードを格納しています。

### 5.3 変数一覧

表 5-3 にグローバル変数を示します。

表 5-3 サンプルコードで使用するグローバル変数

| Type    | Variable Name              | Contents       | Function Used |
|---------|----------------------------|----------------|---------------|
| uint8_t | g_tx_data                  | 送信データバッファ      | main.c        |
| uint8_t | g_rx_data                  | 受信データバッファ      | main.c        |
| uint8_t | g_slave_select_flag        | スレーブセレクトフラグ    | main.c        |
| uint8_t | g_status_confirmation_flag | ステータス確認フラグ     | main.c        |
| uint8_t | g_timeout_flag             | タイムアウトフラグ      | main.c        |
| uint8_t | g_num                      | スレーブが送信可能なデータ数 | main.c        |
| uint8_t | g_rx_data_stored[]         | 受信データの格納       | main.c        |

## 5.4 関数一覧

表 5-4 に関数一覧を示します。

表 5-4 関数一覧

| 関数名                | 概要            | ソースファイル |
|--------------------|---------------|---------|
| main               | メイン処理         | main.c  |
| CSI00_Status_check | CSI ステータス確認処理 | main.c  |
| CSI00_Send_Receive | CSI 送受信処理     | main.c  |
| CSI00_Send         | CSI 送信処理      | main.c  |
| CSI00_Receive      | CSI 受信処理      | main.c  |
| wait               | スレーブ応答待ち処理    | main.c  |

## 5.5 関数仕様一覧

サンプルコードの関数仕様を示します。

[関数名] main

|       |                                                                                                                                                                                                                                                |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概要    | メイン処理                                                                                                                                                                                                                                          |
| ヘッダ   | r_smc_entry.h                                                                                                                                                                                                                                  |
| 宣言    | void main(void)                                                                                                                                                                                                                                |
| 説明    | CSI00、TAU01H の動作開始を行います。<br>スレーブの選択を行います。<br>スレーブ選択フラグ g_slave_select_flag を切り替えます。<br>スレーブ 1 選択時: g_slave_select_flag = 0<br>スレーブ 2 選択時: g_slave_select_flag = 1<br>タイムアウトを検出、または送受信完了後にステータス確認フラグ g_status_confirmation_flag を 0 (初期値) にします。 |
| 引数    | なし                                                                                                                                                                                                                                             |
| リターン値 | なし                                                                                                                                                                                                                                             |
| 備考    | なし                                                                                                                                                                                                                                             |

[関数名] CSI00\_Status\_check

|       |                                                                                             |
|-------|---------------------------------------------------------------------------------------------|
| 概要    | CSI ステータス確認処理                                                                               |
| ヘッダ   | r_smc_entry.h                                                                               |
| 宣言    | void CSI00_Status_check (void)                                                              |
| 説明    | スレーブとステータス確認処理を行います。<br>スレーブと正常にステータス確認が取れるとステータス確認フラグ g_status_confirmation_flag を 1 にします。 |
| 引数    | なし                                                                                          |
| リターン値 | なし                                                                                          |
| 備考    | なし                                                                                          |

---

[関数名] CSI00\_Send\_Receive

---

|       |                                |
|-------|--------------------------------|
| 概要    | CSI 送受信処理                      |
| ヘッダ   | r_smc_entry.h                  |
| 宣言    | void CSI00_Send_Receive (void) |
| 説明    | マスタ送受信処理を行います。                 |
| 引数    | なし                             |
| リターン値 | なし                             |
| 備考    | なし                             |

---

[関数名] CSI00\_Send

---

|       |                        |
|-------|------------------------|
| 概要    | CSI 送信処理               |
| ヘッダ   | r_smc_entry.h          |
| 宣言    | void CSI00_Send (void) |
| 説明    | マスタ送信処理を行います。          |
| 引数    | なし                     |
| リターン値 | なし                     |
| 備考    | なし                     |

---

[関数名] CSI00\_Receive

---

|       |                           |
|-------|---------------------------|
| 概要    | CSI 受信処理                  |
| ヘッダ   | r_smc_entry.h             |
| 宣言    | void CSI00_Receive (void) |
| 説明    | マスタ受信処理を行います。             |
| 引数    | なし                        |
| リターン値 | なし                        |
| 備考    | なし                        |

---

[関数名] wait

---

|       |                                        |
|-------|----------------------------------------|
| 概要    | スレーブ応答待ち処理                             |
| ヘッダ   | r_smc_entry.h                          |
| 宣言    | uint8_t wait(void)                     |
| 説明    | BUSY 信号がロウ・レベルになるのを待ちます                |
| 引数    | なし                                     |
| リターン値 | 0 : BUSY 信号のロウ・レベルを検出<br>1 : タイムアウトを検出 |
| 備考    | なし                                     |



## 5.6 フローチャート

## 5.6.1 メイン処理フローチャート

図 5-1~図 5-2 に本アプリケーションノートの全体フローを示します。

図 5-1 メイン処理 1/2

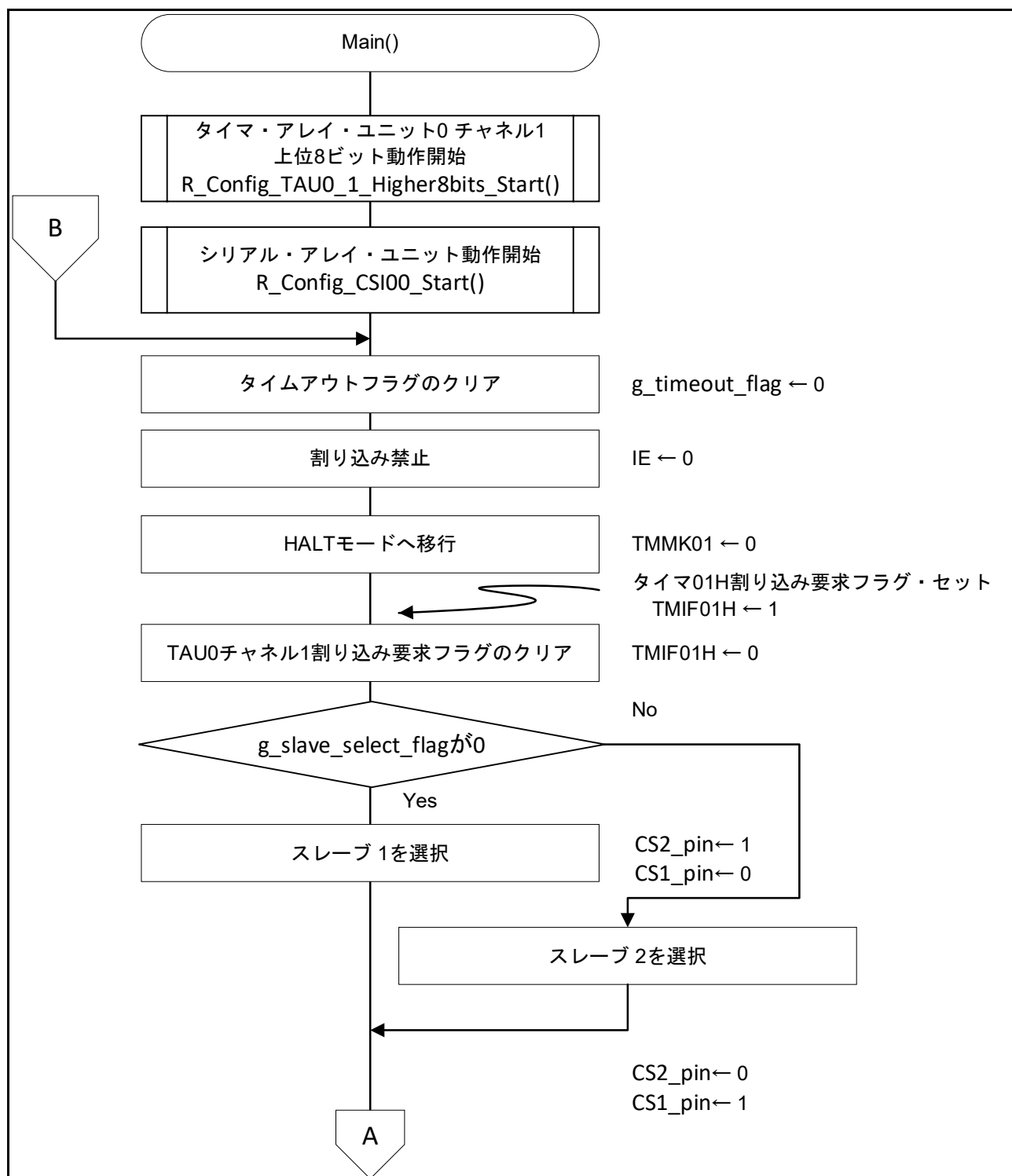
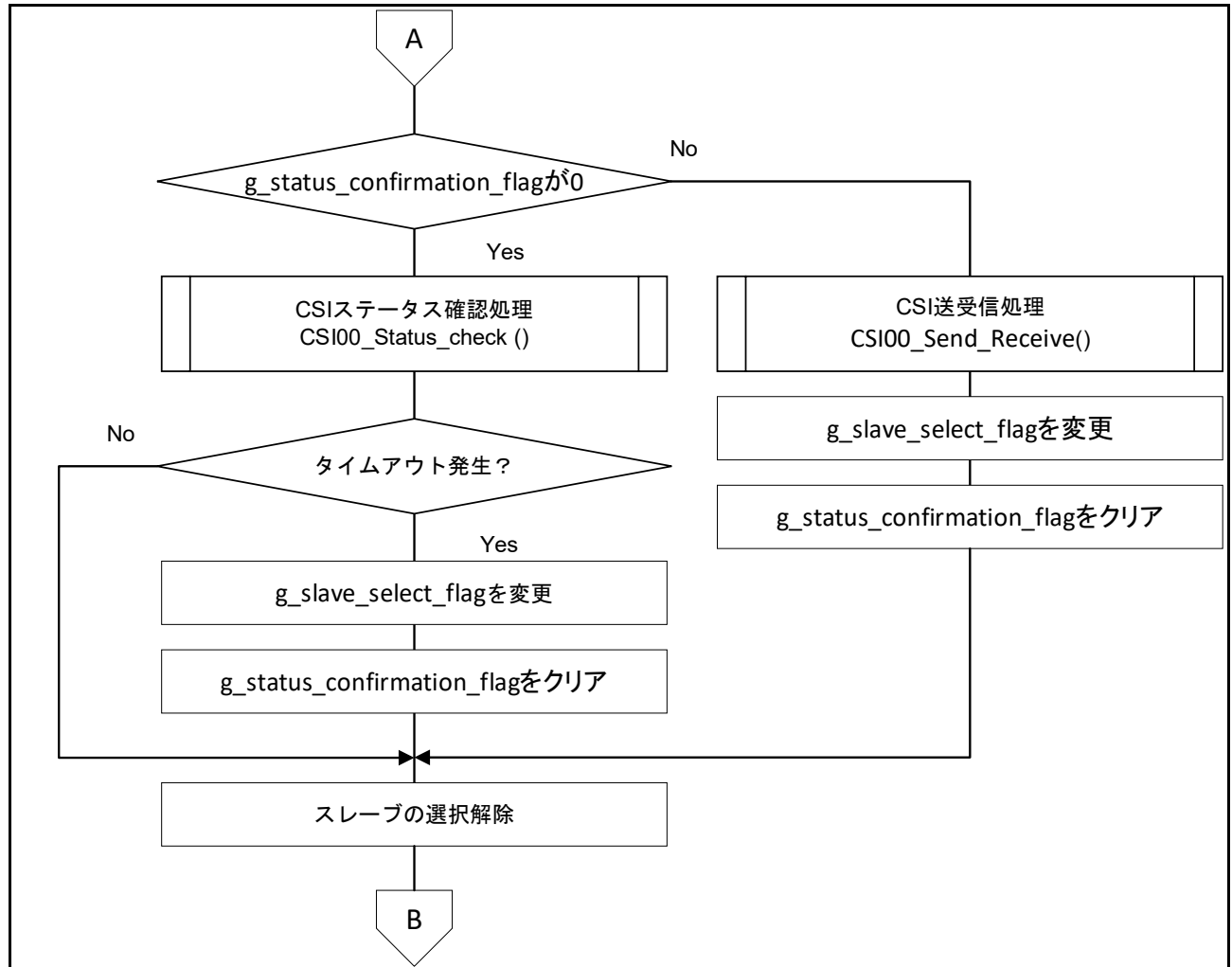


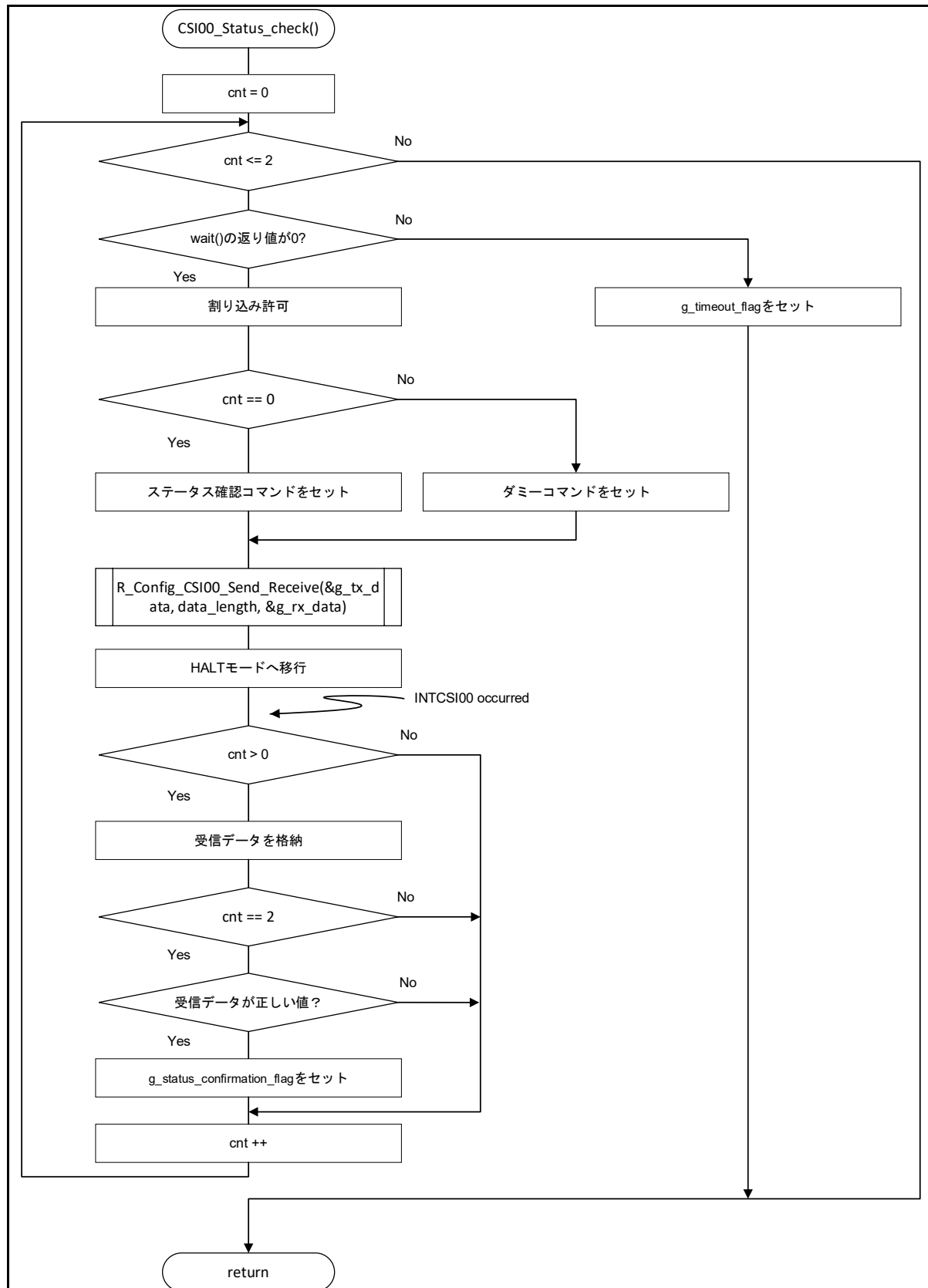
図 5-2 メイン処理 2/2



## 5.6.2 CSI ステータス確認処理フローチャート

図 5-3 にステータス確認のフローを示します。

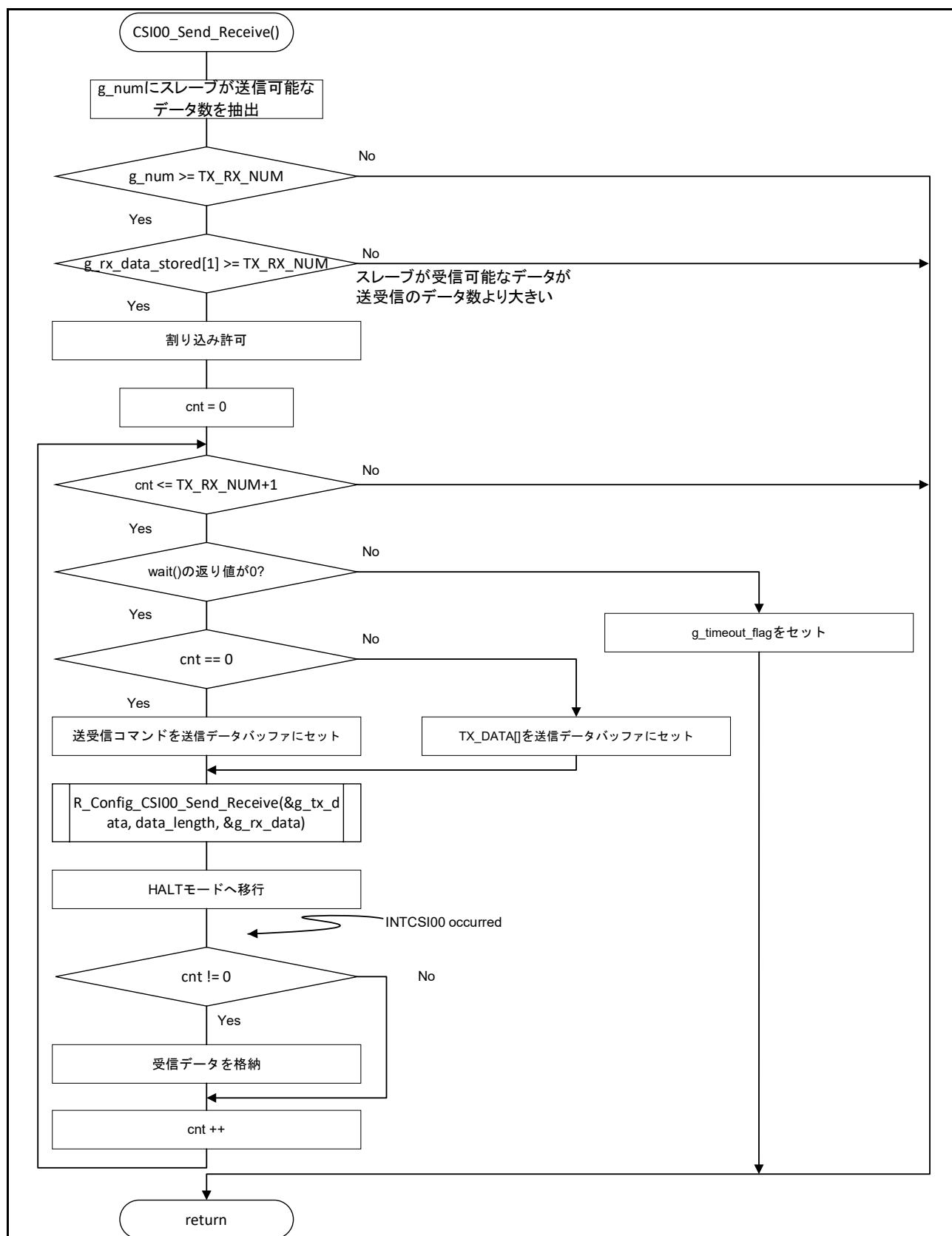
図 5-3 CSI ステータス確認処理



## 5.6.3 CSI 送受信処理フローチャート

図 5-4 に送受信処理のフローを示します。

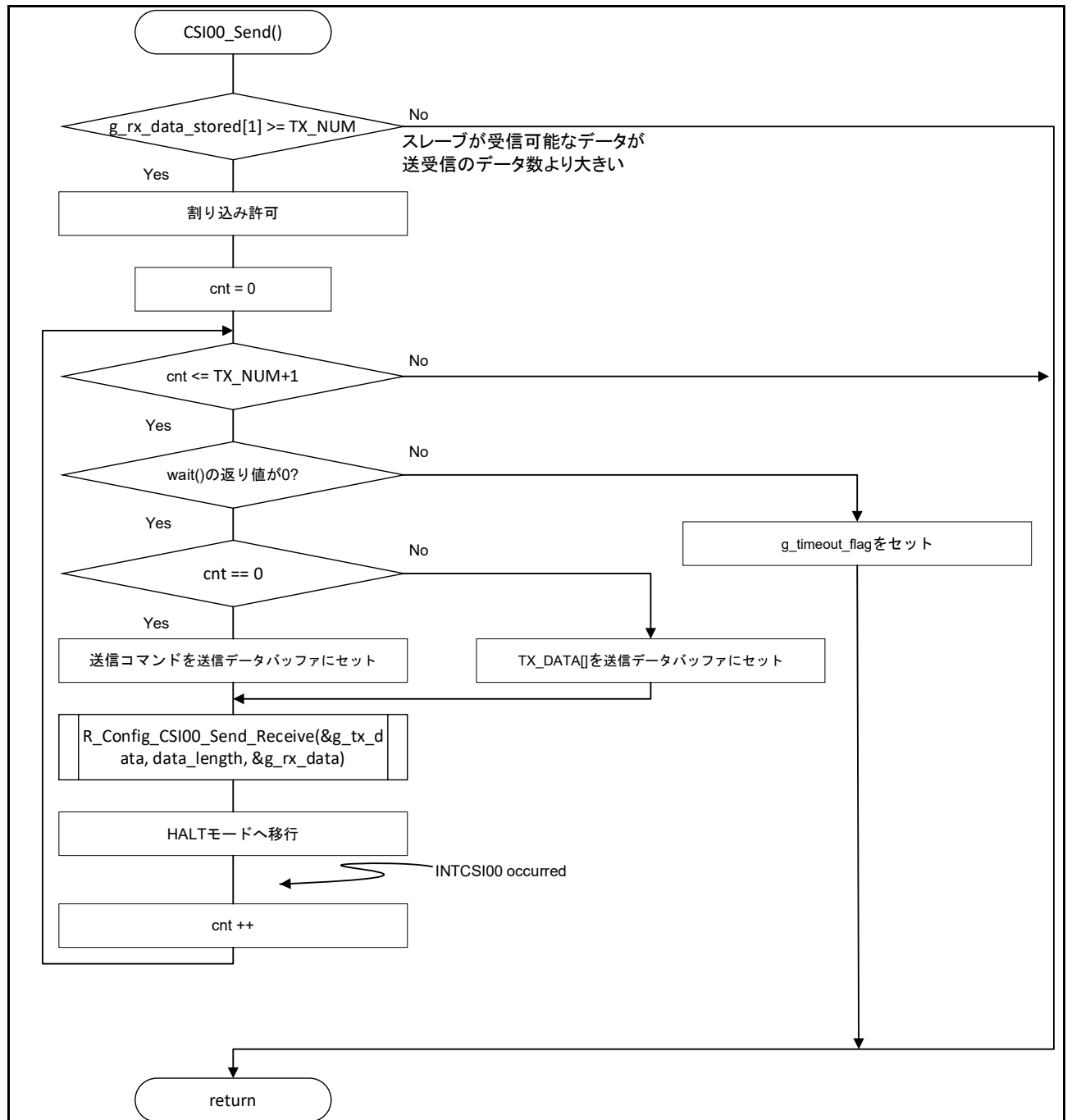
図 5-4 CSI 送受信処理



## 5.6.4 CSI 送信処理フローチャート

図 5-5 に送信処理のフローを示します。

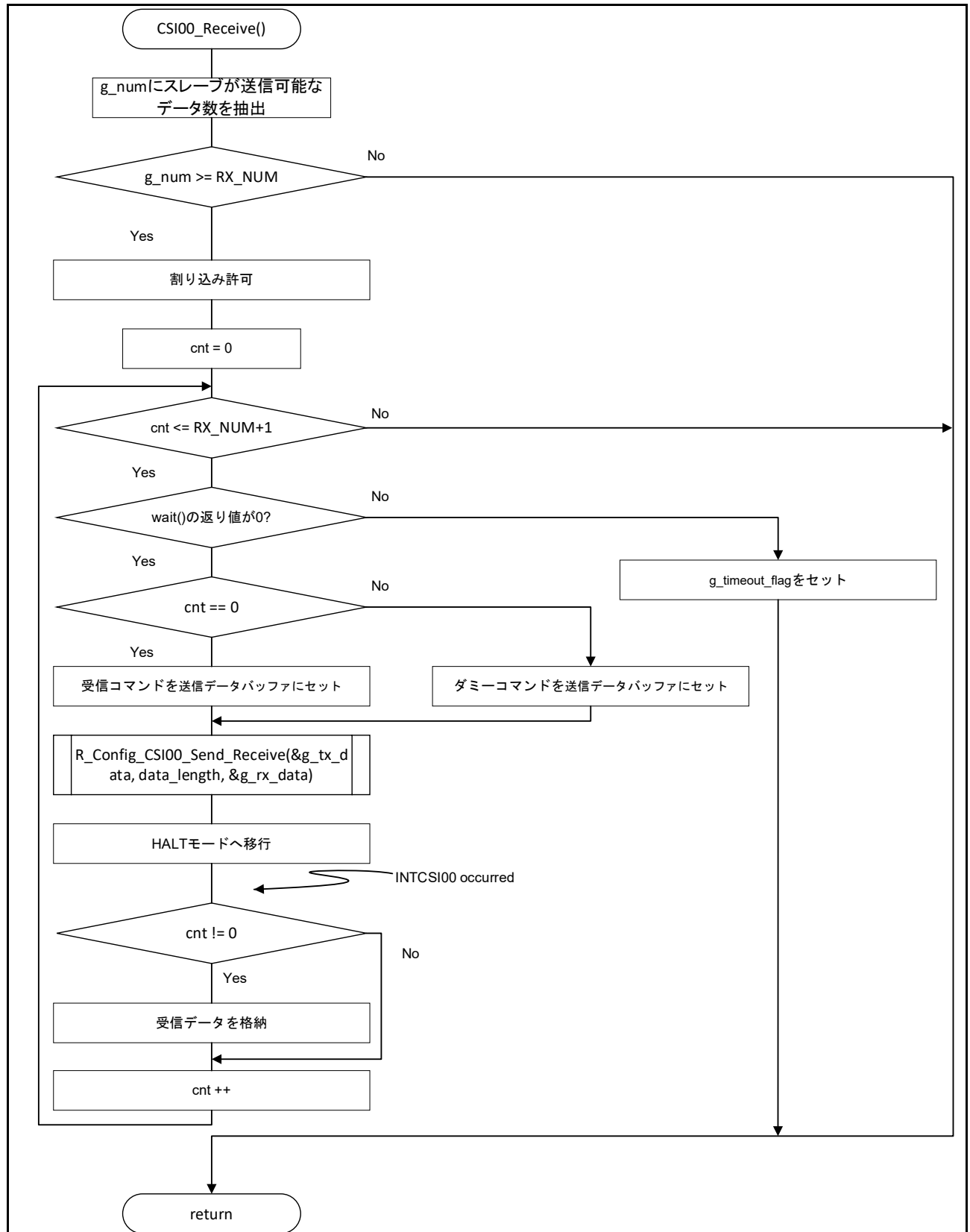
図 5-5 CSI 送信処理



## 5.6.5 CSI 受信処理フローチャート

図 5-6 に受信処理のフローを示します。

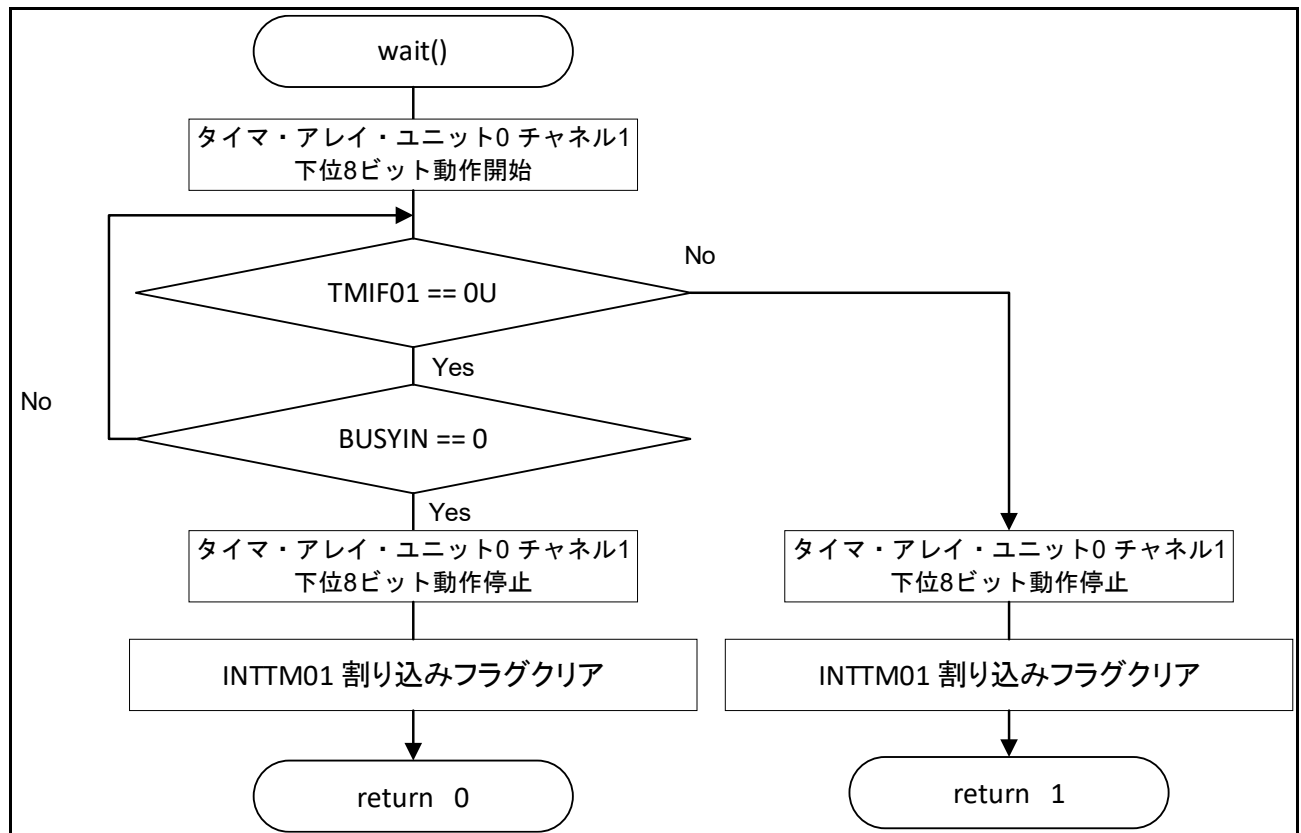
図 5-6 CSI 受信処理



## 5.6.6 スレーブ応答待ち処理フローチャート

図 5-7 にスレーブ応答待ち処理のフローを示します。

図 5-7 スレーブ応答待ち処理



## 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/G15 ユーザーズマニュアル ハードウェア編 (R01UH0959J)

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 | 2024.3.29 | -    | 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、 $V_{IL}(\text{Max.})$  から  $V_{IH}(\text{Min.})$  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、 $V_{IL}(\text{Max.})$  から  $V_{IH}(\text{Min.})$  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くと、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）、交通管制（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、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た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したします。
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](http://www.renesas.com)

## 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。

## 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/](http://www.renesas.com/contact/)