

---

# RX63Nグループ、RX631 グループ

## RSPI、DTCa および MTU2a を用いた I2S 通信

---

R01AN1339JJ0100  
Rev.1.00  
2013.04.05

### 要旨

本アプリケーションノートでは、RX63N グループ、RX631 グループのシリアルペリフェラルインタフェース (以下、RSPI)、データトランスファコントローラ (以下、DTCa)、およびマルチファンクションタイマパルスユニット 2 (以下、MTU2a)を使用して、I2S 通信によるオーディオデータの転送を実現する方法を説明します。

### 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     | 使用端子一覧                              | 5  |
| 4.2     | 周辺機能の接続                             | 6  |
| 5.      | ソフトウェア説明                            | 7  |
| 5.1     | 動作概要                                | 7  |
| 5.1.1   | RSPI、DTCa および MTU2a による I2S 通信の実現方法 | 7  |
| 5.1.1.1 | オーディオデータのフォーマット                     | 7  |
| 5.1.1.2 | オーディオデータの送受信                        | 9  |
| 5.1.1.3 | I2S 通信の同期回復と RSPI の初期化              | 12 |
| 5.1.1.4 | I2S 通信のクロックとチャンネル同期信号の生成            | 13 |
| 5.1.1.5 | ワードセレクト信号とスレーブセレクト信号の位相差            | 15 |
| 5.1.2   | 送信動作                                | 16 |
| 5.1.2.1 | 送信動作タイミング                           | 16 |
| 5.1.2.2 | 送信動作時の DTC 動作                       | 17 |
| 5.1.2.3 | DTC による送信データアドレスの転送                 | 19 |
| 5.1.2.4 | オーディオデータの送信終了                       | 20 |
| 5.1.3   | 受信動作                                | 22 |
| 5.1.3.1 | 受信動作タイミング                           | 22 |
| 5.1.3.2 | 受信動作時の DTC 動作                       | 23 |
| 5.1.3.3 | DTC による受信データアドレスの転送                 | 25 |
| 5.1.3.4 | オーディオデータの受信終了                       | 26 |
| 5.2     | ファイル構成                              | 28 |
| 5.3     | 定数一覧                                | 29 |
| 5.4     | 構造体/共用体一覧                           | 30 |
| 5.5     | 変数一覧                                | 31 |
| 5.6     | 関数一覧                                | 32 |
| 5.7     | 関数仕様                                | 33 |
| 6.      | フローチャート                             | 38 |
| 6.1     | main 関数                             | 38 |
| 6.2     | 関数                                  | 39 |
| 6.2.1   | i2s_au_data_init 関数                 | 39 |
| 6.2.2   | i2s_start 関数                        | 40 |
| 6.2.3   | i2s_dtc_init 関数                     | 41 |
| 6.2.4   | i2s_dtc_tx_l_init 関数                | 42 |
| 6.2.5   | i2s_dtc_tx_r_init 関数                | 46 |
| 6.2.6   | i2s_dtc_rx_l_init 関数                | 50 |
| 6.2.7   | i2s_dtc_rx_r_init 関数                | 51 |
| 6.2.8   | i2s_mtu2_init 関数                    | 52 |
| 6.2.9   | i2s_mtu2_ch2_init 関数                | 53 |
| 6.2.10  | i2s_mtu2_ch3_init 関数                | 54 |
| 6.2.11  | i2s_mtu2_ch4_init 関数                | 55 |
| 6.2.12  | i2s_rsipi_init 関数                   | 56 |
| 6.2.13  | i2s_rsipi0_init 関数                  | 57 |
| 6.2.14  | i2s_rsipi1_init 関数                  | 59 |
| 7.      | 付録                                  | 61 |
| 8.      | サンプルコード                             | 62 |
| 9.      | 参考ドキュメント                            | 62 |

## 1. 仕様

本アプリケーションノートでは、RSPI、DTCa および MTU2a を用いて、I2S 通信によるオーディオデータの送受信を行います。MTU2a が生成するクロック信号に従い、RSPI はオーディオデータのシリアル転送を行います。

表 1.1 に使用する周辺機能と用途を、図 1.1 にブロック図を示します。

表 1.1 使用する周辺機能と用途

| 周辺機能          | 用途                         |
|---------------|----------------------------|
| RSPI チャンネル 0  | オーディオデータ(Left Channel)入出力  |
| RSPI チャンネル 1  | オーディオデータ(Right Channel)入出力 |
| MTU2a チャンネル 2 | シリアル転送クロック(SCK)の生成         |
| MTU2a チャンネル 3 | ワードセレクト信号(WS)の生成           |
| MTU2a チャンネル 4 | RSPI へのスレーブセレクト信号(SSL)の生成  |
| DTCa          | 内蔵 RAM のオーディオデータ転送         |

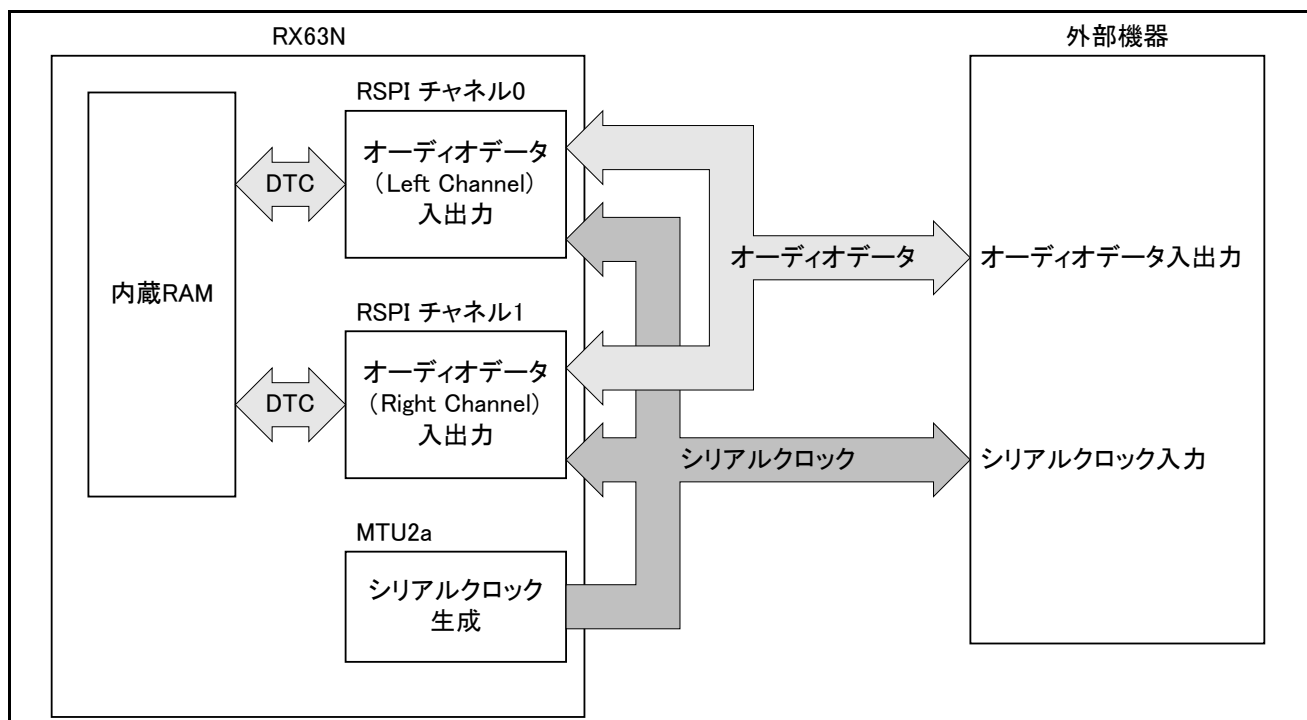


図 1.1 ブロック図

## 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"> <li>・メインクロック : 12MHz</li> <li>・サブクロック : 32.768kHz</li> <li>・PLL : 192MHz (メインクロック 1 分周 16 逡倍)</li> <li>・システムクロック (ICLK) : 96MHz (PLL2 分周)</li> <li>・周辺モジュールクロック A (PCLKA) : 96MHz (PLL2 分周)</li> <li>・周辺モジュールクロック B (PCLKB) : 48MHz (PLL4 分周)</li> <li>・外部バスクロック (BCLK) : 48MHz (PLL4 分周)</li> <li>・FlashIF クロック (FCLK) : 48MHz (PLL4 分周)</li> <li>・IEBUS クロック (IECLK) : 48MHz (PLL4 分周)</li> </ul> |
| 動作電圧              | 3.3V   |
| 総合開発環境            | ルネサスエレクトロニクス製<br>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<br>C/C++ compiler package for RX Family V.1.02 Release01<br>コンパイルオプション<br>-cpu=rx600 -output=obj="\$ (CONFIGDIR)¥\$(FILELEAF).obj" -debug -nologo<br>(統合開発環境のデフォルト設定を使用しています)  |
| iodifine.h のバージョン | Version 1.50   |
| 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          | スーパーバイザーモード  |
| 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 (R0K50563NC000BE)   |

## 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 Rev1.00 (R01AN1045JJ0100)

上記アプリケーションノートの初期設定関数を、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 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 使用端子と用途

| 端子名          | 入出力 | 用途                             |
|--------------|-----|--------------------------------|
| P1_4/MTCLKA  | 入力  | 外部クロック(12.288MHz)入力            |
| PB_5/MTIOC2A | 出力  | シリアル転送クロック(SCK)出力              |
| P1_7/MTIOC3A | 出力  | ワードセレクト信号(WS)出力                |
| PA_0/MTIOC4A | 出力  | スレーブセレクト信号(SSL)出力              |
| PA_4/SSLA0   | 入力  | RSPI0 スレーブセレクト信号入力             |
| PA_5/RSPCKA  | 入力  | RSPI0 シリアル転送クロック入力             |
| PA_6/MOSIA   | 入力  | オーディオデータ(Left channel)入力       |
| PA_7/MISOA   | 出力  | オーディオデータ(Left channel)出力(注 1)  |
| PE_4/SSLB0   | 入力  | RSPI1 スレーブセレクト信号入力             |
| PE_5/RSPCKB  | 入力  | RSPI1 シリアル転送クロック入力             |
| PE_2/MOSIB   | 入力  | オーディオデータ(Right channel)入力      |
| PE_3/MISOB   | 出力  | オーディオデータ(Right channel)出力(注 1) |

注 1：受信モードの場合は、データを出さないために出力端子をハイインピーダンスに設定します。

## 4.2 周辺機能の接続

図 4.1 に接続図を示します。また、表 4.2 に MTU2a が生成するクロック信号を示します。

MTU2a は外部クロック(12.288MHz)をカウントクロックとし、表 4.2 に示すクロック信号を生成します。

RSPI チャンネル 0(以下、RSPI0)はオーディオデータの Left Channel(以下、L-ch)成分、RSPI チャンネル 1(以下、RSPI1)は Right Channel(以下、R-ch)成分の送受信を行います。RSPI はスレーブモードで動作し、SSL によりアクティブとなるチャンネルを切り替えます。

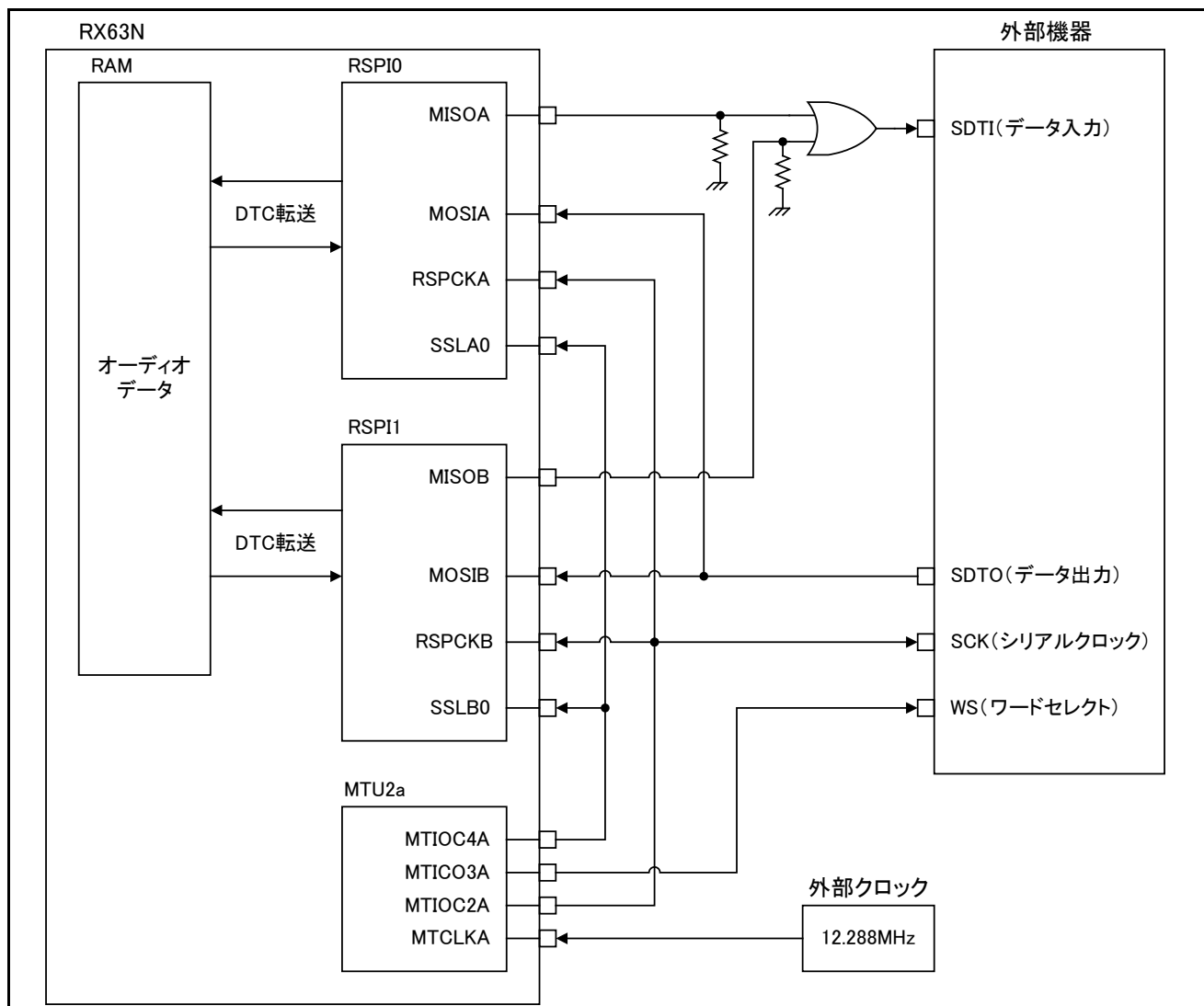


図 4.1 接続図

表 4.2 MTU2a が生成するクロック信号

| チャンネル   | クロック信号名         | シンボル | 出力端子    | 周波数      |
|---------|-----------------|------|---------|----------|
| チャンネル 2 | シリアル転送クロック      | SCK  | MTIOC2A | 3.072MHz |
| チャンネル 3 | ワードセレクト信号       | WS   | MTIOC3A | 48kHz    |
| チャンネル 4 | RSPI スレーブセレクト信号 | SSL  | MTIOC4A | 48kHz    |

## 5. ソフトウェア説明

### 5.1 動作概要

#### 5.1.1 RSPI、DTCa および MTU2a による I2S 通信の実現方法

RSPI、DTCa および MTU2a を用いて I2S 通信を実現する方法を示します。

##### 5.1.1.1 オーディオデータのフォーマット

本アプリケーションノートでは、オーディオデータは 24 ビット(データ)+8 ビット(パディング)の合計 32 ビットとして扱います。

##### (a) 内蔵 RAM 上のデータフォーマット

図 5.1 に内蔵 RAM のオーディオデータを示します。

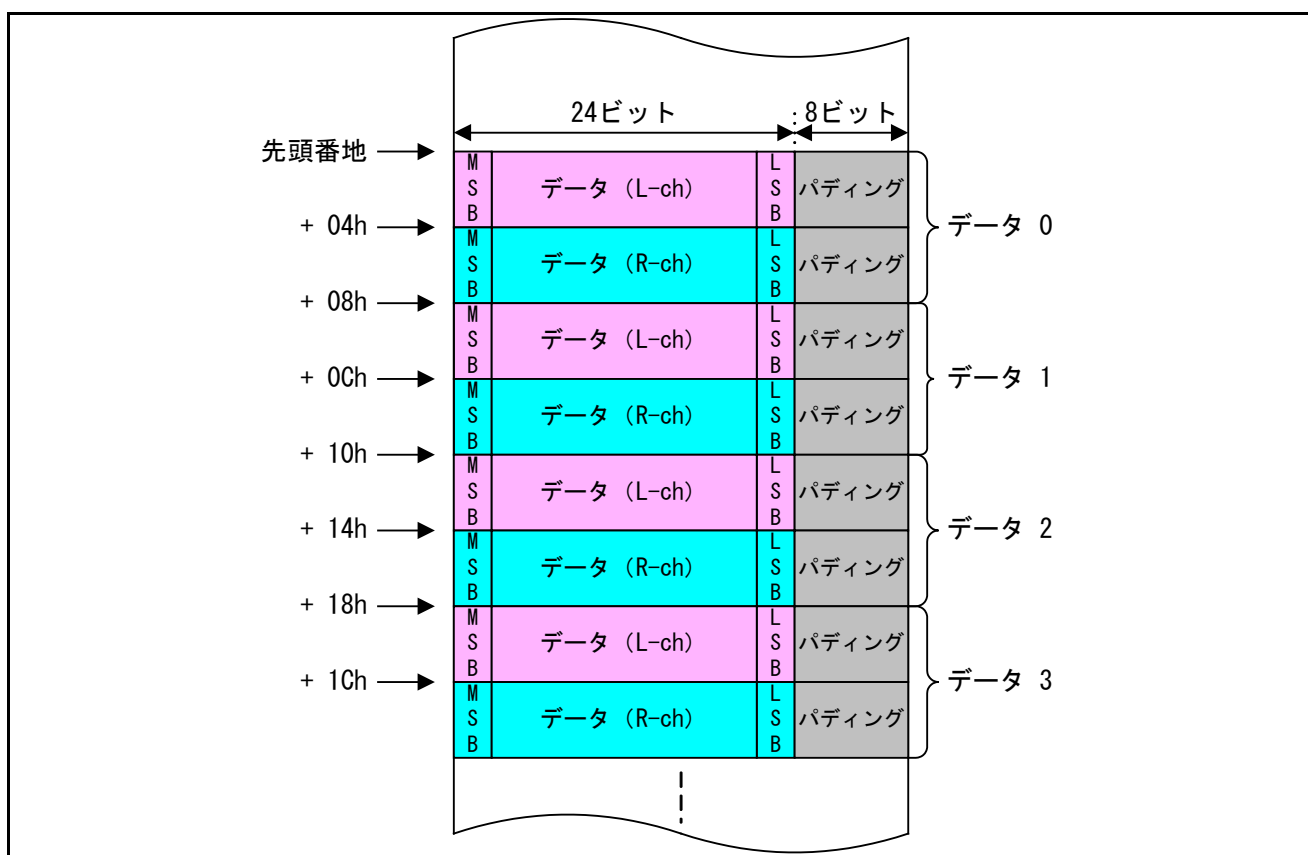


図 5.1 内蔵 RAM 上のオーディオデータ

(b) 送受信データフォーマット

本アプリケーションノートでは、データ長 32 ビット、MSB ファーストのシリアル転送を行います。SCK はシリアル転送クロック、WS はチャンネルセレクト信号として使用し、WS が Low のときオーディオデータの L-ch 成分、WS が High のときオーディオデータの R-ch 成分の転送を行います。

送受信データは、外部機器のオーディオインターフェースフォーマットに従い、パディングを付加する位置を選択します。本アプリケーションノートでは、以下の 3 つのフォーマットを選択することができます。

- 標準フォーマット
- 後方パディングフォーマット
- 前方パディングフォーマット

図 5.2 に送受信データフォーマットを示す。

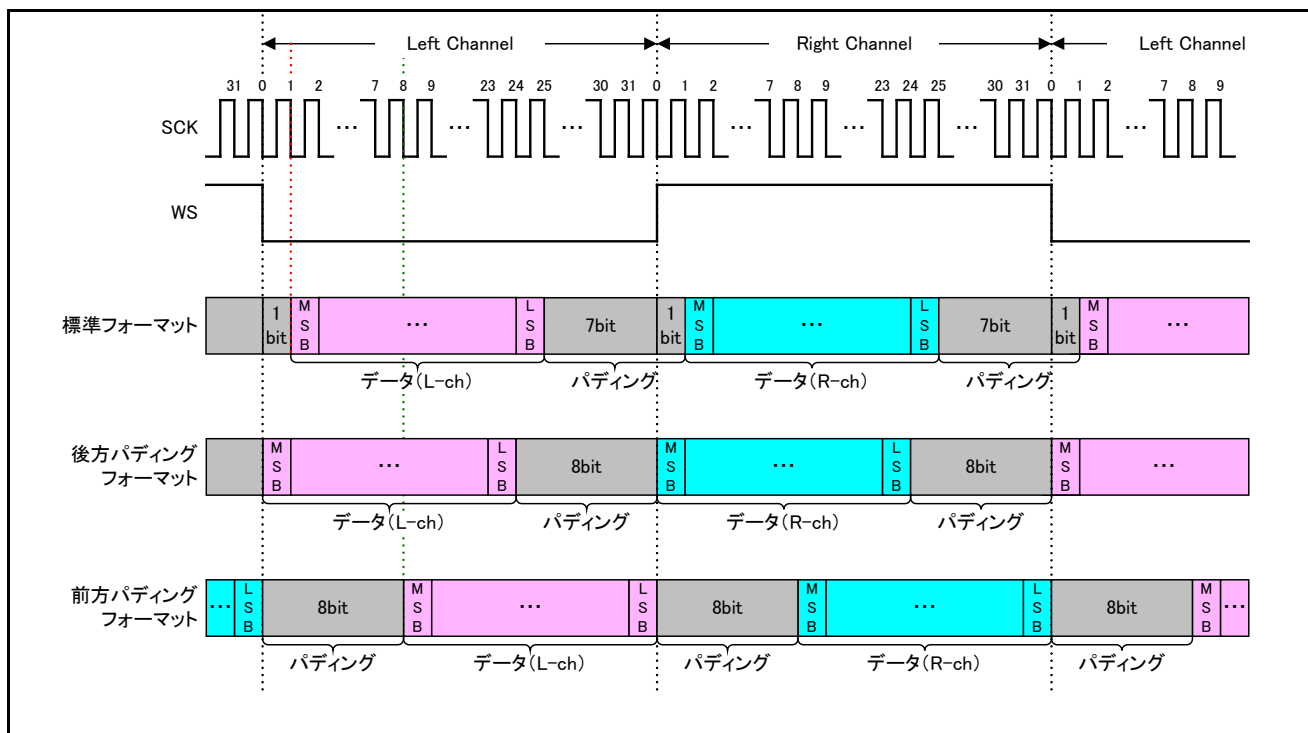


図 5.2 送受信データフォーマット



## 5.1.1.2 オーディオデータの送受信

I2S 通信の送信は、内蔵 RAM に配置したオーディオデータを DTC により RSPI に転送し、RSPI から出力することで行います。I2S 通信の受信は、RSPI から受信したオーディオデータを DTC により内蔵 RAM に転送することで行います。

RSPI はスレーブモード(SPI 動作)、データ長 32 ビット、MSB ファーストです。

図 5.3 にオーディオデータの流れを示します。

オーディオデータは L-ch 成分および R-ch 成分に分割し、RSPI0 は L-ch 成分、RSPI1 は R-ch 成分の送受信を行います。RSPI0 のスレーブセレクト極性を Low アクティブ、RSPI1 のスレーブセレクト極性を High アクティブに設定し、MTU2a チャンネル 4 で出力する SSL により RSPI0 と RSPI1 のチャンネルを選択します。

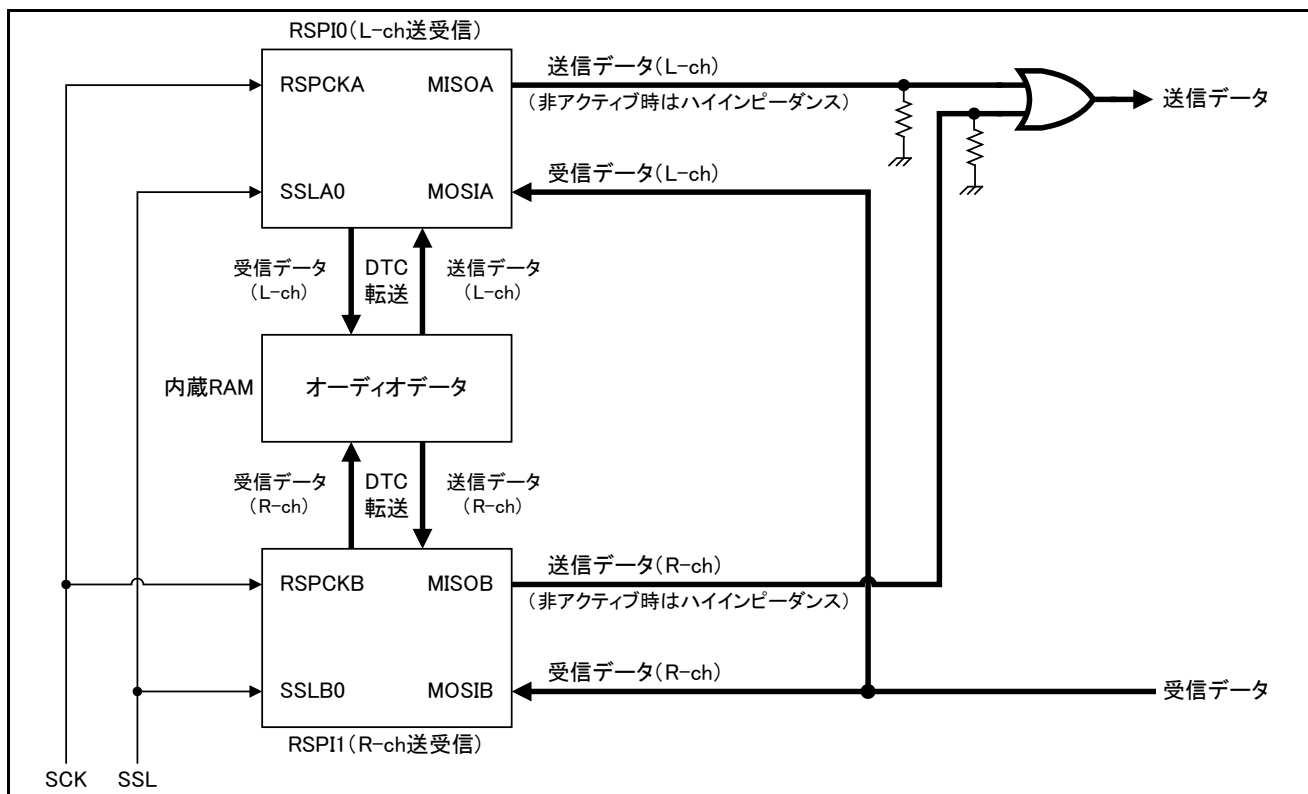


図 5.3 オーディオデータの流れ

## (a) 送信動作

図 5.4 に RSPI によるオーディオデータの送信タイミングを示します。

外部機器へ送信するオーディオデータは、RSPI0 から出力する送信データ(L-ch)と RSPI1 から出力する送信データ(R-ch)を、外付けの論理和回路で合成して生成します。このとき、送信データを出力していない RSPI の出力はハイインピーダンスであるため、論理和した合成データに影響が無いようにプルダウンしてください。

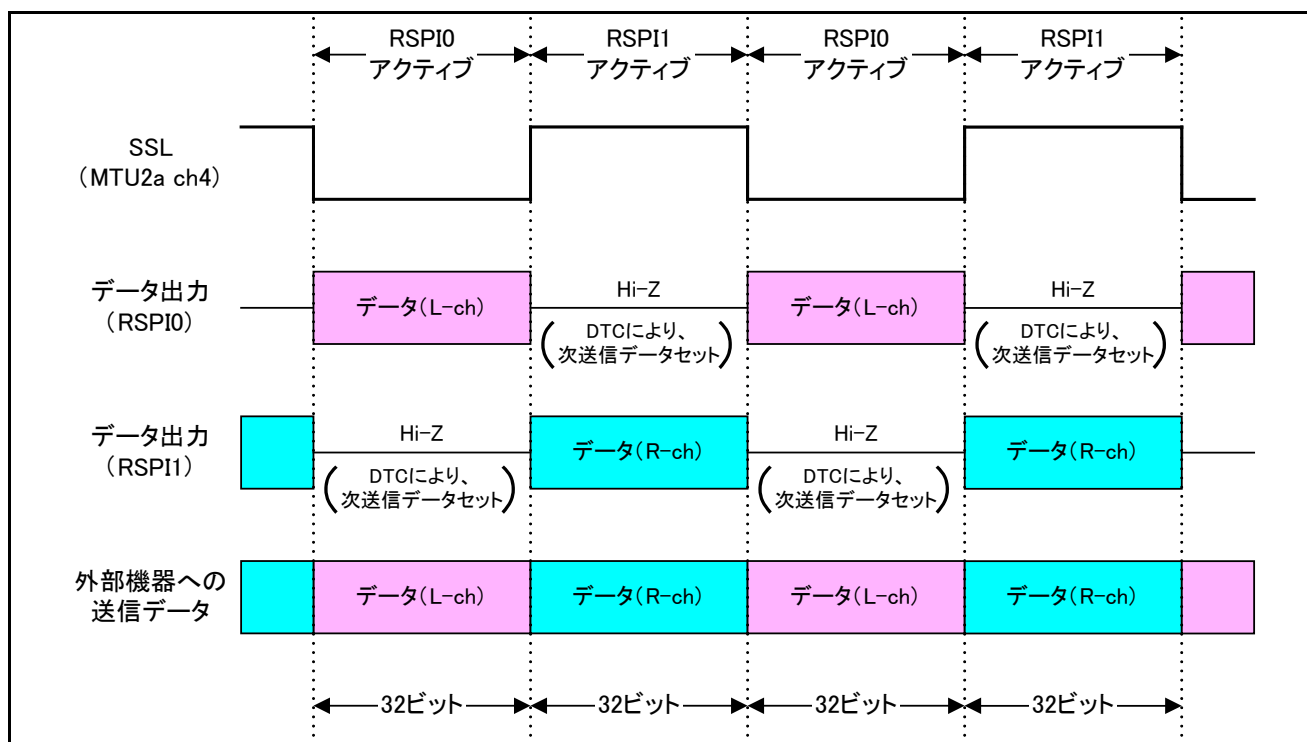


図 5.4 RSPI によるオーディオデータの送信タイミング

(b) 受信動作

図 5.5 に RSPI によるオーディオデータの受信タイミングを示します。  
 外部機器からのオーディオデータ受信は RSPI0 および RSPI1 の両チャンネルに入力します。このとき、SSL によってアクティブとなる RSPI のチャンネルが選択され、入力されたオーディオデータを L-ch 成分と R-ch 成分に分割して受信します。

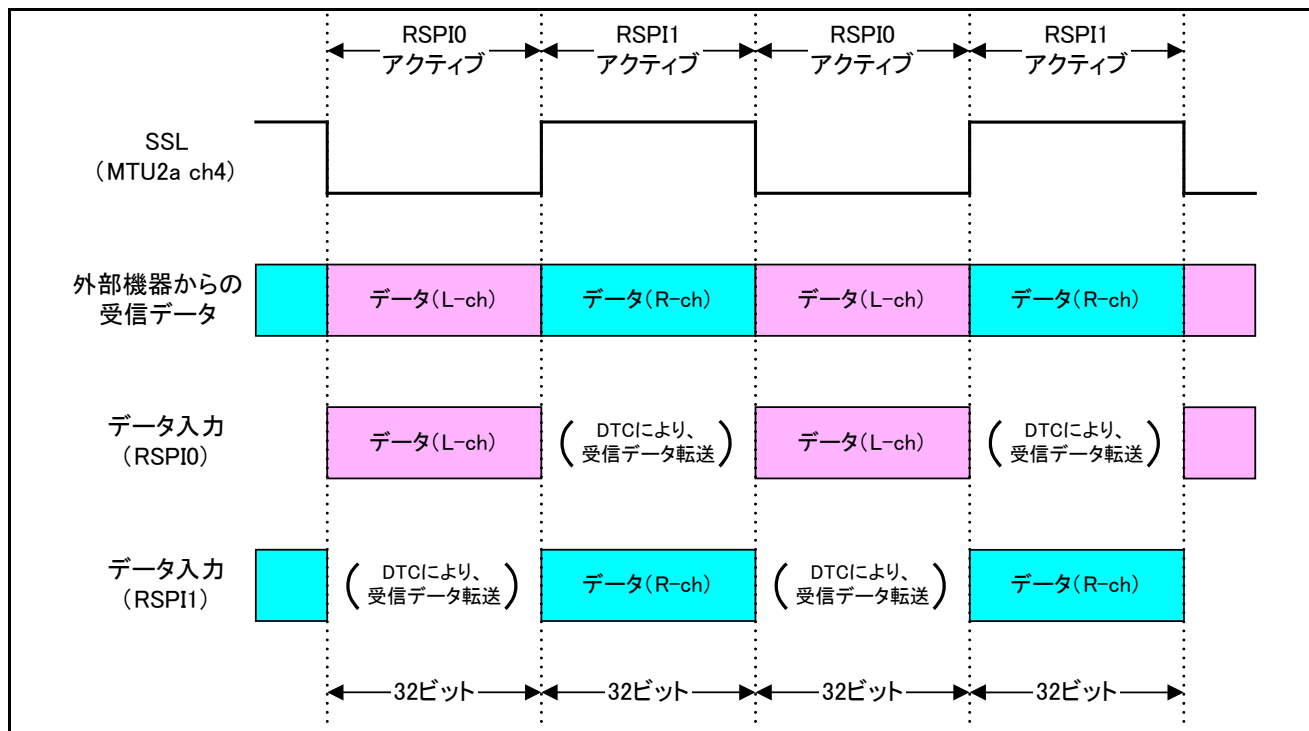


図 5.5 RSPI によるオーディオデータの受信タイミング

5.1.1.3 I2S 通信の同期回復と RSPI の初期化

ノイズが混入し、ビットずれが起こった場合にI2S通信の同期回復を行うため、L-ch、R-chそれぞれでデータ転送時にRSPIの初期化を行っています。RSPIの初期化はDTCによって実行します。

図5.6にRSPIの初期化タイミングを示します。

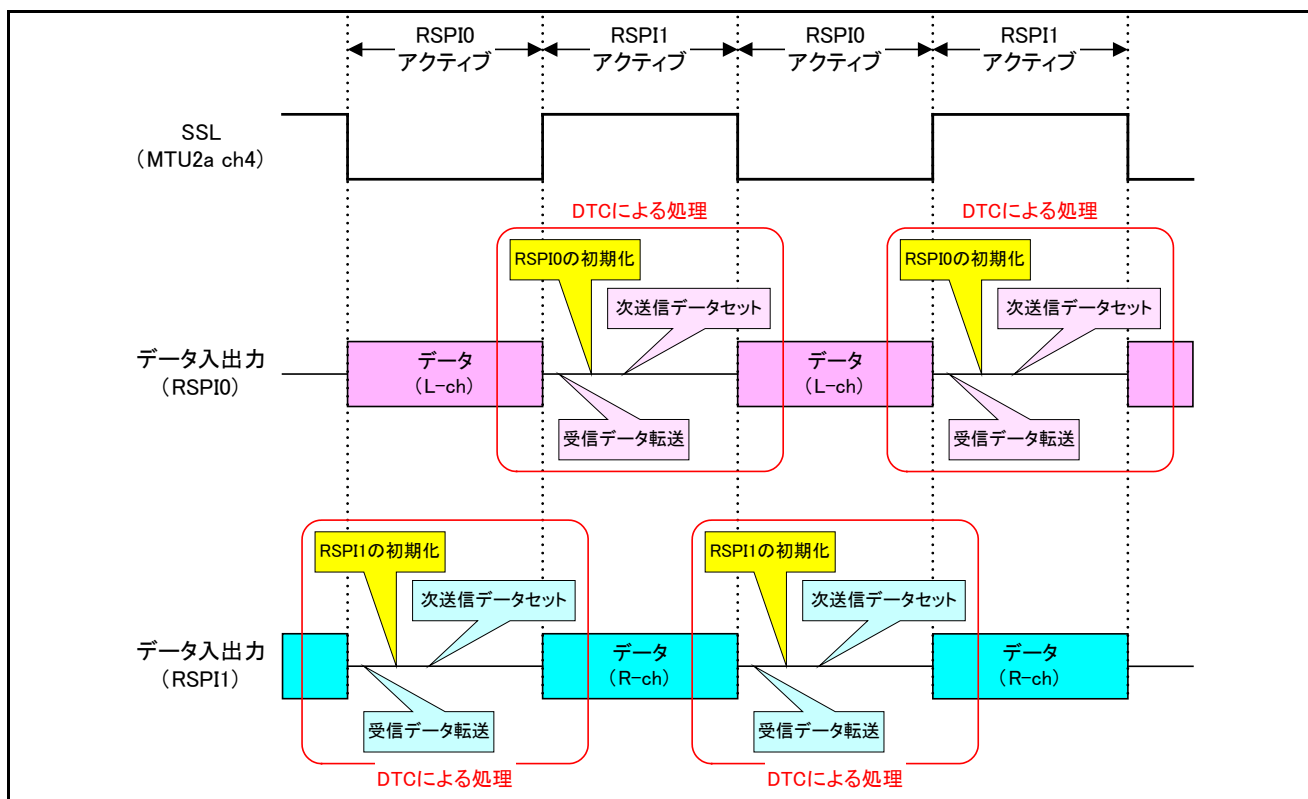


図 5.6 RSPI の初期化タイミング

## 5.1.1.4 I2S 通信のクロックとチャンネル同期信号の生成

外部クロック(12.288MHz)をカウントクロックとして、SCK と WS を MTU2a のコンペアマッチにより生成します。同様に、RSPI のチャンネルセレクト信号(SSL)を MTU2a のコンペアマッチにより生成します。

## (a) SCK の生成

MTU2a チャンネル 2 を PWM モード 1 に設定し、周波数 3.072MHz(12.288MHz/4)、デューティ比 50%、初期出力が High 出力の SCK を出力します。

図 5.7 に MTU2a チャンネル 2 による SCK の生成を示します。

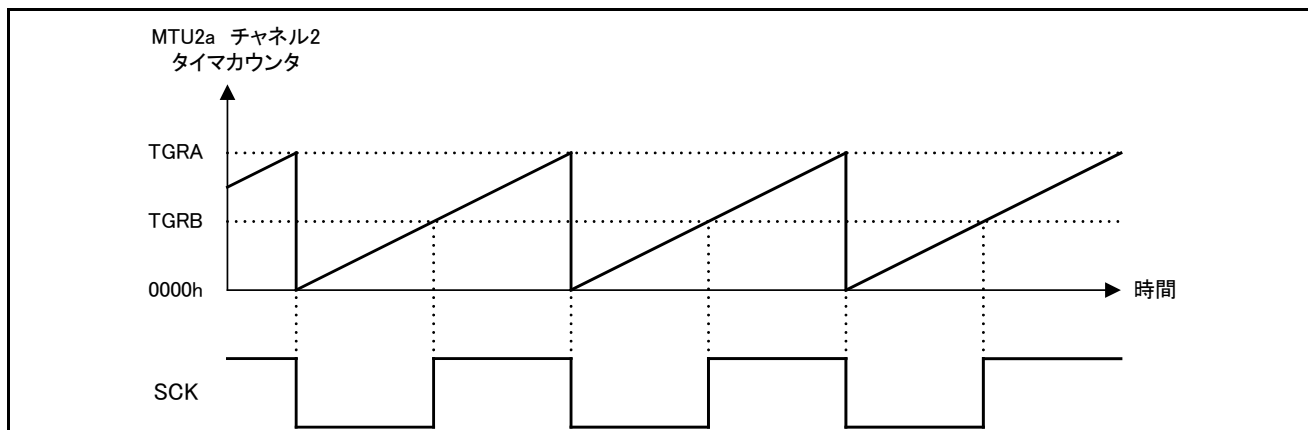


図 5.7 MTU2a チャンネル 2 による SCK の生成

## (b) WS の生成

MTU2a チャンネル 3 を PWM モード 1 に設定し、周波数 48kHz(12.288MHz/256)、デューティ比 50%、初期出力が High 出力の WS を出力します。

図 5.8 に MTU2a チャンネル 3 による WS の生成を示します。

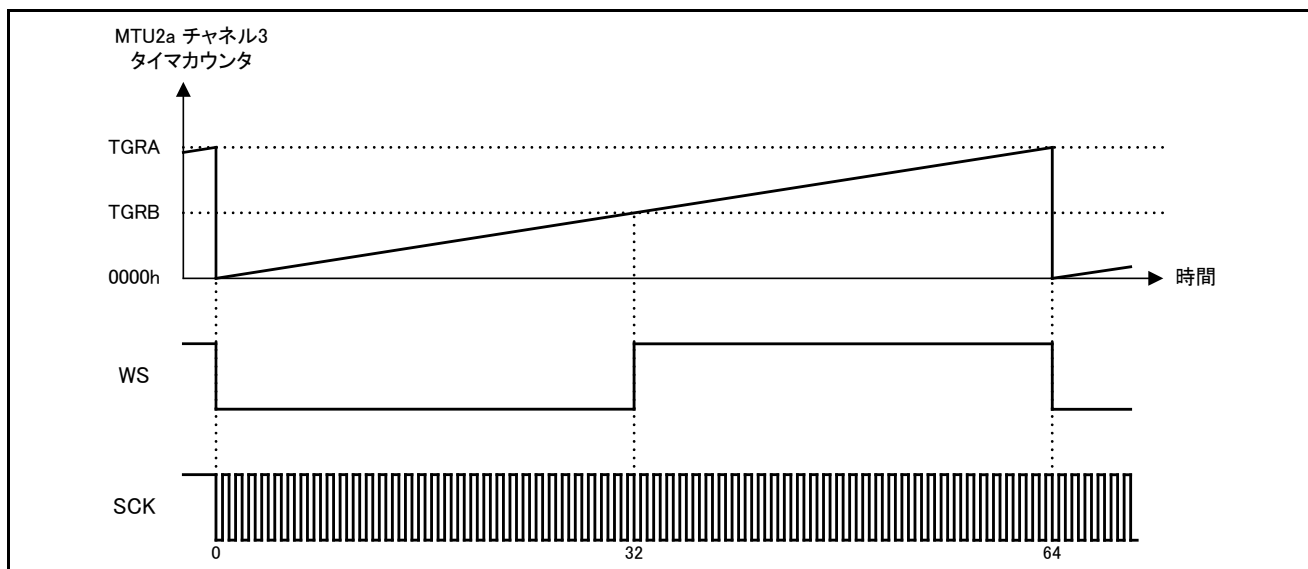


図 5.8 MTU2a チャンネル 3 による WS の生成

## (c) SSL の生成

MTU2a チャンネル 4 を PWM モード 1 に設定し、周波数 48kHz(12.288MHz/256)、デューティ比 50%、初期出力が High 出力の SSL を出力します。

MTU2a チャンネル 4 のタイマカウンタの初期値を、WS を生成している MTU2a チャンネル 3 のタイマカウンタの初期値と異なる値に設定することにより、WS と SSL の位相差を生成します。WS と SSL の位相差によって、「5.1.1.1(b) 送受信データフォーマット」に示した各送受信データフォーマットに対応します。WS と SSL の位相差の詳細は「5.1.1.5 ワードセレクト信号とスレーブセレクト信号の位相差」を参照してください。

図 5.9 に MTU2a チャンネル 4 による SSL の生成を示します。

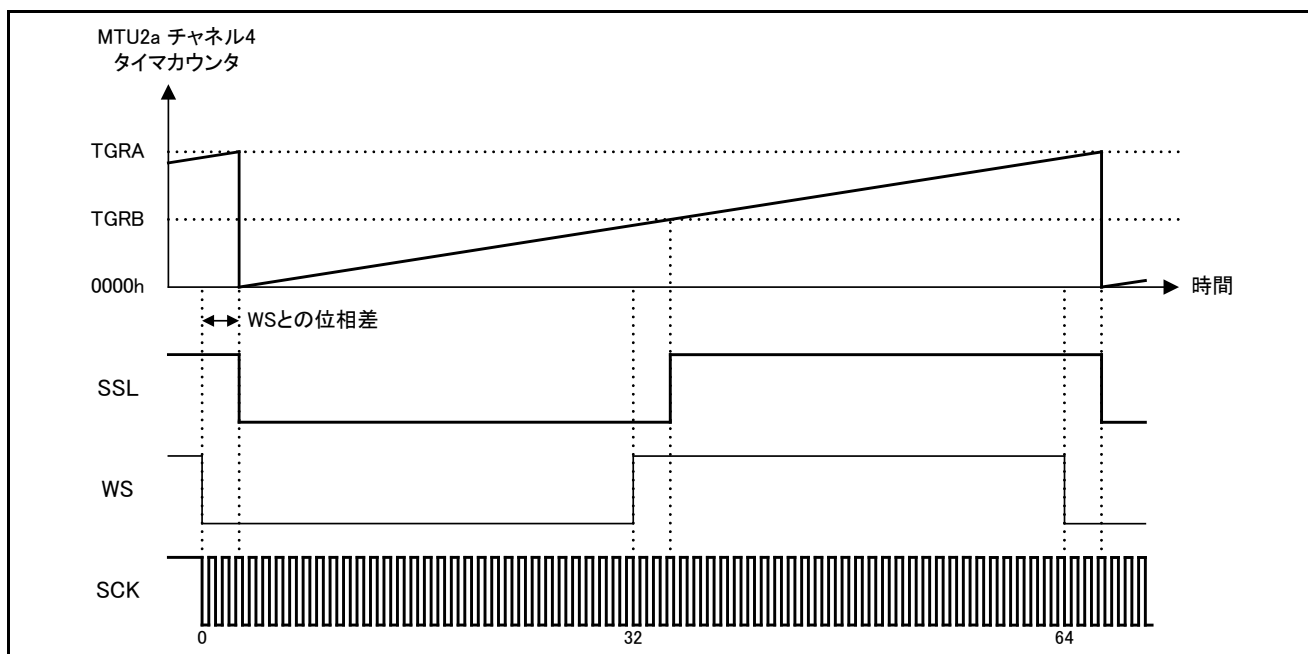


図 5.9 MTU2a チャンネル 4 による SSL の生成

5.1.1.5 ワードセレクト信号とスレーブセレクト信号の位相差

本アプリケーションノートでは、「5.1.1.4 (c) SSL の生成」にて示した MTU2a チャンネル 4 の初期値を変更することにより、「5.1.1.1 (b) 送受信データフォーマット」に示した各送受信データフォーマットに対応します。

図 5.10 に各送受信データフォーマットの WS と SSL の位相差を示します。

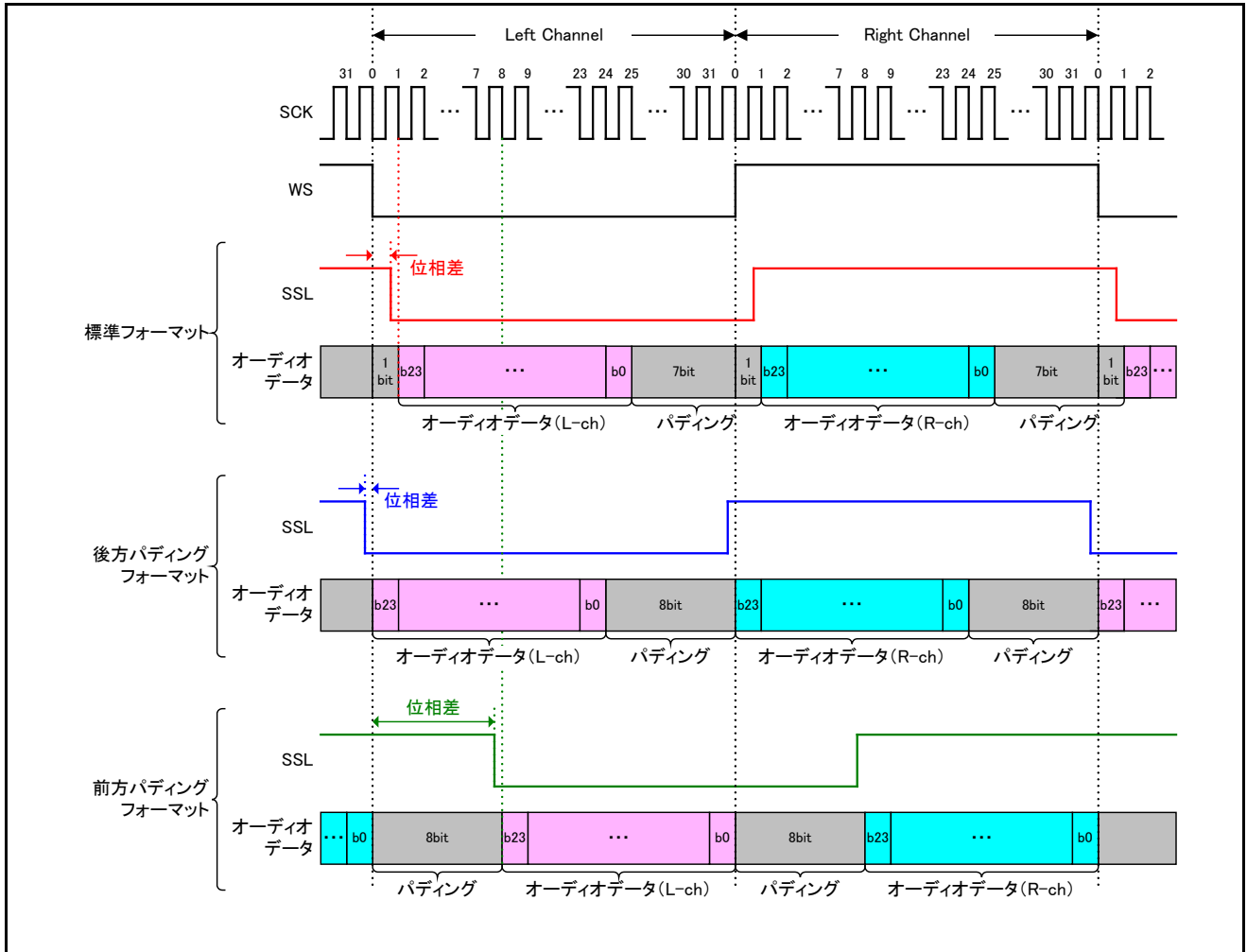


図 5.10 各送受信データフォーマットの WS と SSL の位相差





## 5.1.2.2 送信動作時の DTC 動作

図 5.12 に送信動作時の DTC 動作を示します。

DTC は内蔵 RAM から RSPI に送信データを転送します。DTC 転送は L-ch 送信および R-ch 送金の 2 つがあります。DTC は、SSL のエッジ発生時のコンペアマッチ割り込みで起動します。SSL の立ち上がりエッジでは L-ch 送金の DTC、立ち下がりエッジでは R-ch 送金の DTC が起動します。

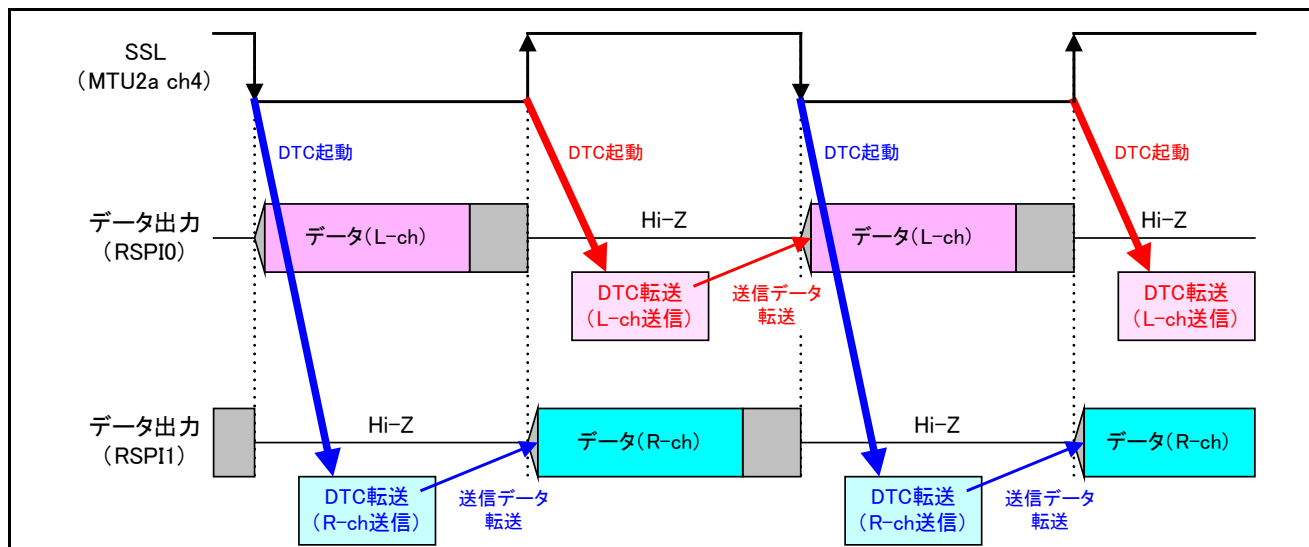


図 5.12 送信動作時の DTC 動作

図 5.13 に L-ch 送信の DTC 動作を、図 5.14 に R-ch 送信の DTC 動作を示します。  
 DTC はチェーン転送により、RSPI の初期化、送信データの転送、送信データアドレスの転送を行います。  
 なお、RSPI の初期化は、RSPI 制御レジスタ (SPCR) の RSPI 機能許可ビット (SPE) に "0" を書いて RSPI を無効化し、続いて "1" を書いて RSPI を再び有効化することによって行います。  
 注) RSPI 機能許可ビットによる初期化についてはハードウェアマニュアルを参照してください。

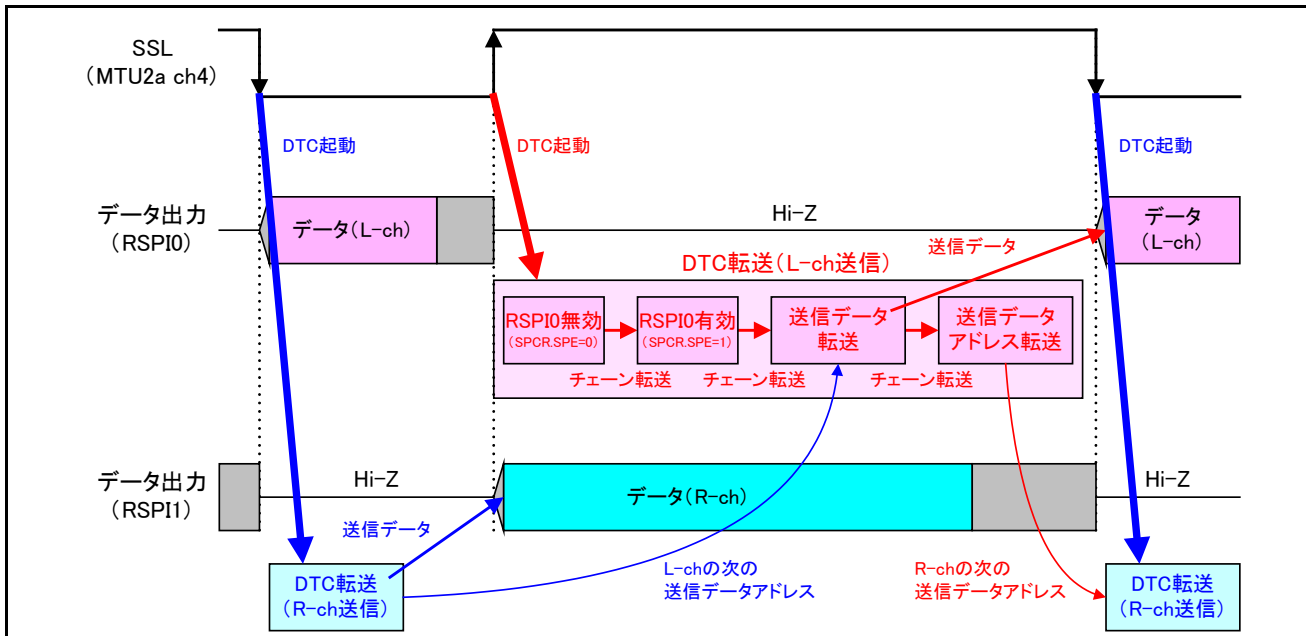


図5.13 L-ch送信のDTC動作

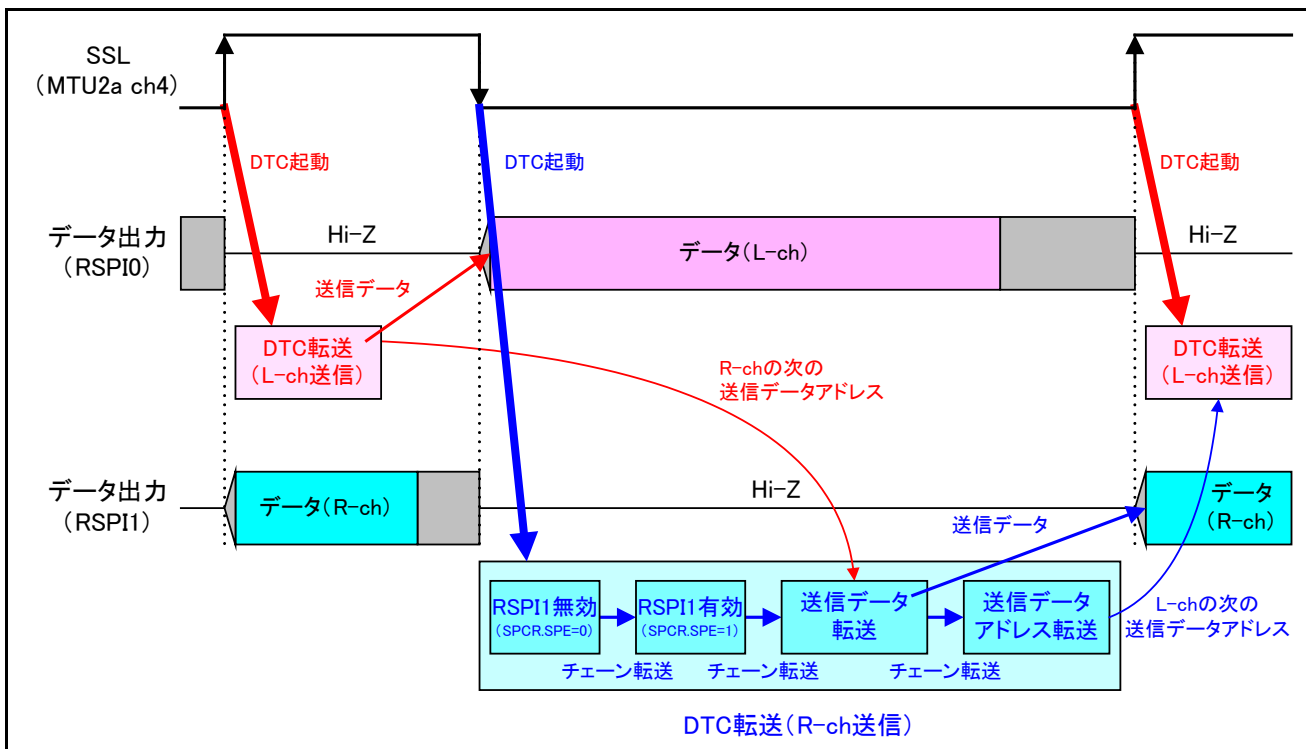


図5.14 R-ch送信のDTC動作

5.1.2.3 DTC による送信データアドレスの転送

図 5.15 に DTC による送信データアドレスの転送を示します。

本アプリケーションノートでは、送信するオーディオデータを「図 5.1 内蔵 RAM 上のオーディオデータ」に示した通り、L-ch 成分と R-ch 成分を 4 バイトずつ交互に内蔵 RAM 上に配置しています。本アプリケーションノートでは、L-ch 送信の DTC 転送と R-ch 送信の DTC 転送の 2 つの DTC 転送を使用して、送信データの転送を交互に行っています。

図 5.15 に示した通り、L-ch 送信の DTC 転送は送信データのデータ 0(L-ch)を転送後、チェーン転送を用いて L-ch 送信の DTC 転送後の転送元アドレスを R-ch 送信の DTC 転送の転送元アドレスに転送します。R-ch 送信の DTC 転送は、L-ch 送信の DTC 転送後の転送元アドレスを開始アドレスとしてデータ 0(R-ch)を転送し、チェーン転送を用いて R-ch 送信の DTC 転送後の転送元アドレスを L-ch 送信の DTC 転送の転送元アドレスに転送します。

L-ch 送信の DTC 転送は、R-ch 送信のチェーン転送で転送された R-ch 送信の DTC 転送後の転送元アドレスを開始アドレスとしてデータ 1(L-ch)を転送し、チェーン転送を用いて L-ch 送信の DTC 転送後の転送元アドレスを次の R-ch 送信の DTC 転送に必要な開始アドレスとして R-ch 送信の DTC 転送の転送元アドレスに転送します。

このように L-ch 送信の DTC 転送は R-ch 送信の DTC 転送に必要な送信データアドレスを、R-ch 送信の DTC 転送は L-ch 送信の DTC 転送に必要な送信データアドレスを転送します。

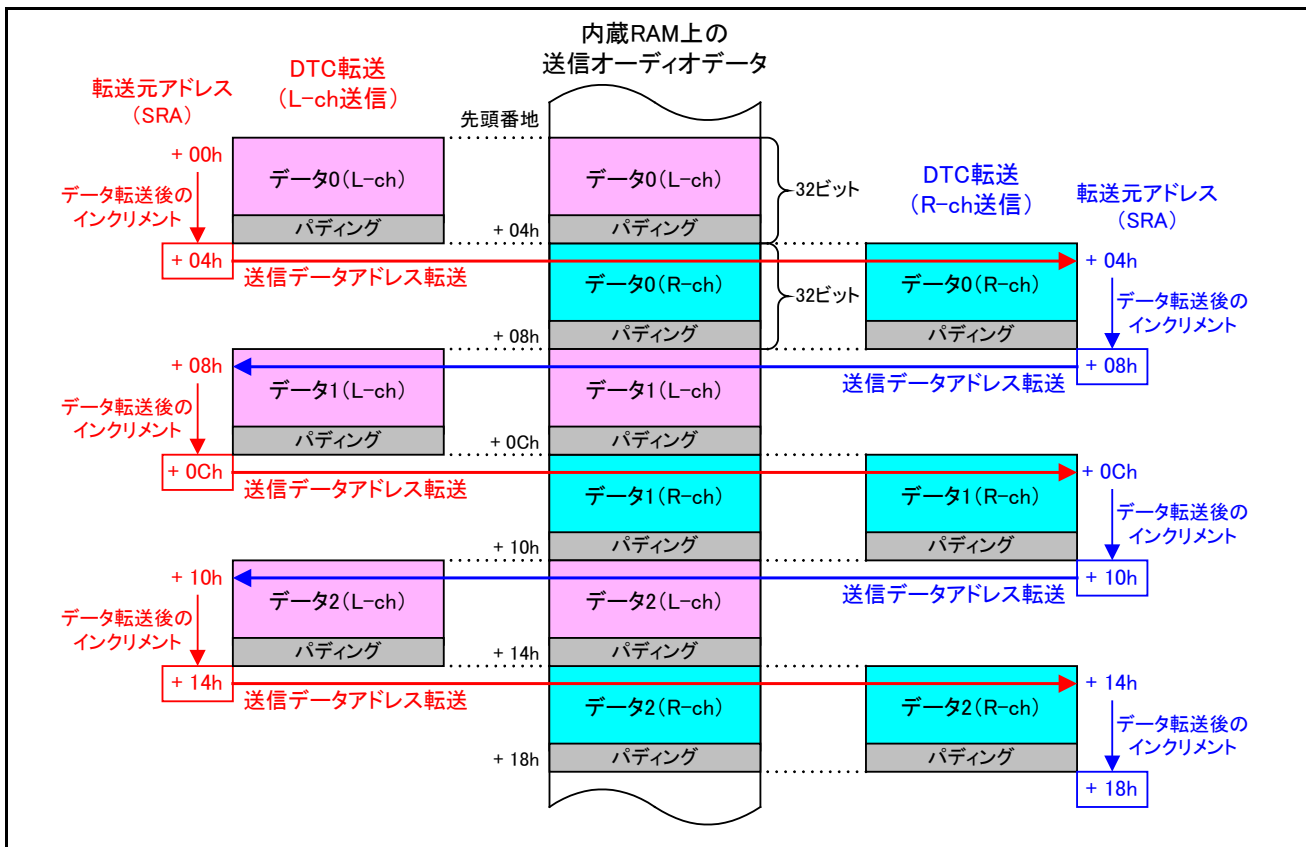


図 5.15 DTC による送信データアドレスの転送

5.1.2.4 オーディオデータの送信終了

本アプリケーションノートでは、RSPI を無効化して動作を停止させることによりオーディオデータの送信終了とします。RSPI の無効化は L-ch 送信および R-ch 送信の DTC 転送が行います。

(a) L-ch 送信の送信終了処理

図 5.16 に L-ch 送信の送信終了動作の概略を示します。

L-ch 送信の DTC 転送は n-1 回目の起動時、チェーン転送を用いて送信終了処理を行います。送信終了処理により、n 回目の起動時は RSPI0 の無効化のみ行われ、RSPI0 は動作を停止します。

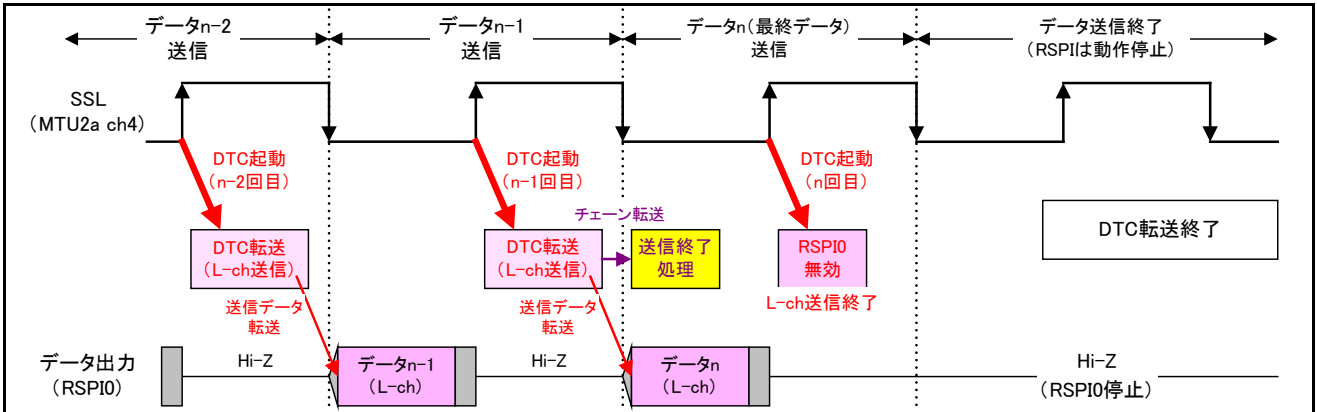


図5.16 L-ch送信の送信終了動作

L-ch送信の送信終了処理を行っているチェーン転送の詳細動作を図5.17に示します。

L-ch送信のDTC転送は、「図5.13 L-ch送信のDTC動作」に示した通り、チェーン転送を用いて、RSPI0無効化、RSPI0有効化、送信データ転送、送信データアドレス転送を行います。

この送信データアドレス転送の転送情報にある転送カウンタをn-1回、DTCチェーン転送選択ビットを“転送カウンタ=0のときのみチェーン転送を行う”に設定することで、n-1回目の起動時の送信データアドレス転送後のチェーン転送により送信終了処理を行います。

送信終了処理は、RSPI0無効化の転送情報にあるDTCチェーン転送イネーブルビットの値を書き換えることで、n回目の起動時におけるRSPI0無効化後のチェーン転送を禁止にします。これにより、n回目の起動時はRSPI0の無効化のみが実行され、RSPI0の動作が停止し送信動作を終了します。

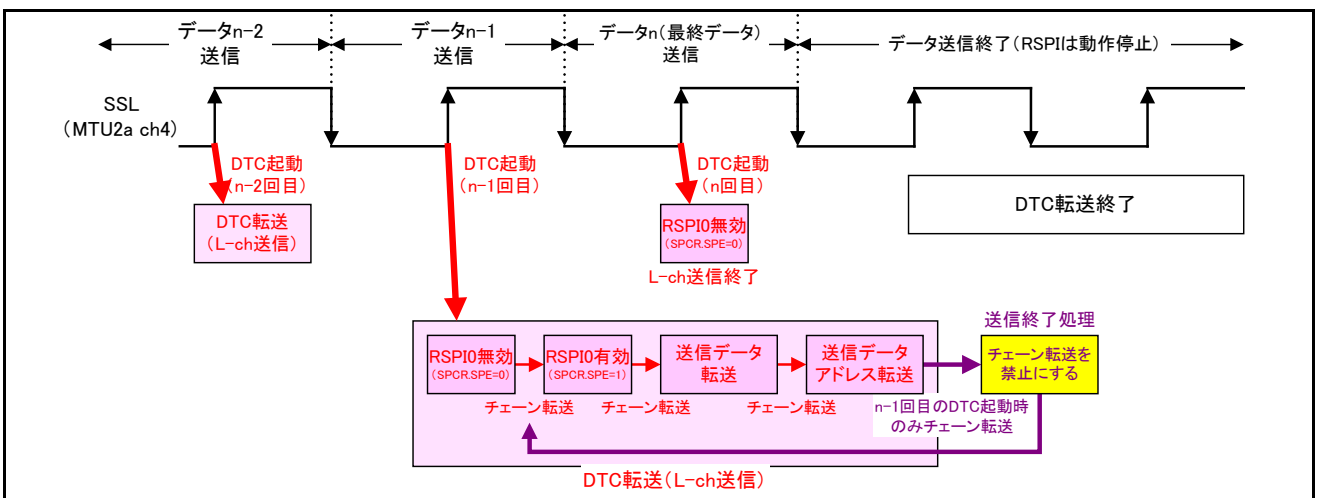


図5.17 L-ch送信のDTC転送による送信終了処理

(b) R-ch送信の送信終了処理

図5.18にR-ch送信の送信終了動作の概略を示します。  
 R-ch送信のDTC転送はn回目の起動時、チェーン転送を用いて送信終了処理を行います。送信終了処理により、n+1回目の起動時はRSPI1の無効化のみ行われ、RSPI1は動作を停止します。

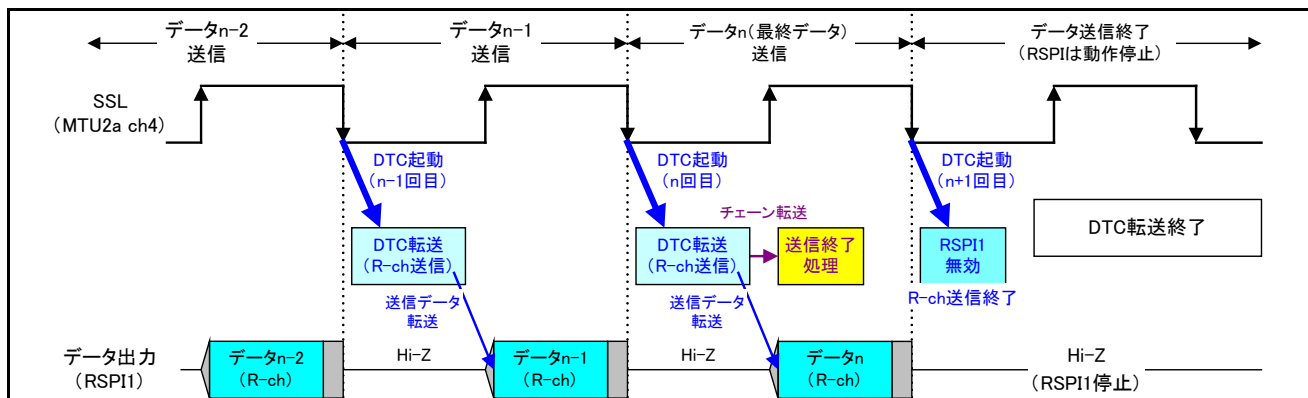


図5.18 R-ch送信の送信終了動作

R-ch送信の送信終了処理を行っているチェーン転送の詳細動作を図5.19に示します。

R-ch送信のDTC転送は、「図5.14 R-ch送信のDTC動作」に示した通り、チェーン転送を用いて、RSPI1無効化、RSPI1有効化、送信データ転送、送信データアドレス転送を行います。

この送信データアドレス転送の転送情報にある転送カウンタをn回、DTCチェーン転送選択ビットを“転送カウンタ=0のときのみチェーン転送を行う”に設定することで、n回目の起動時の送信データアドレス転送後のチェーン転送により送信終了処理を行います。

送信終了処理は、RSPI1無効化の転送情報にあるDTCチェーン転送イネーブルビットの値を書き換えることで、n+1回目の起動時におけるRSPI1無効化後のチェーン転送を禁止にします。これにより、n+1回目の起動時はRSPI1の無効化のみが実行され、RSPI1の動作が停止し送信動作を終了します。

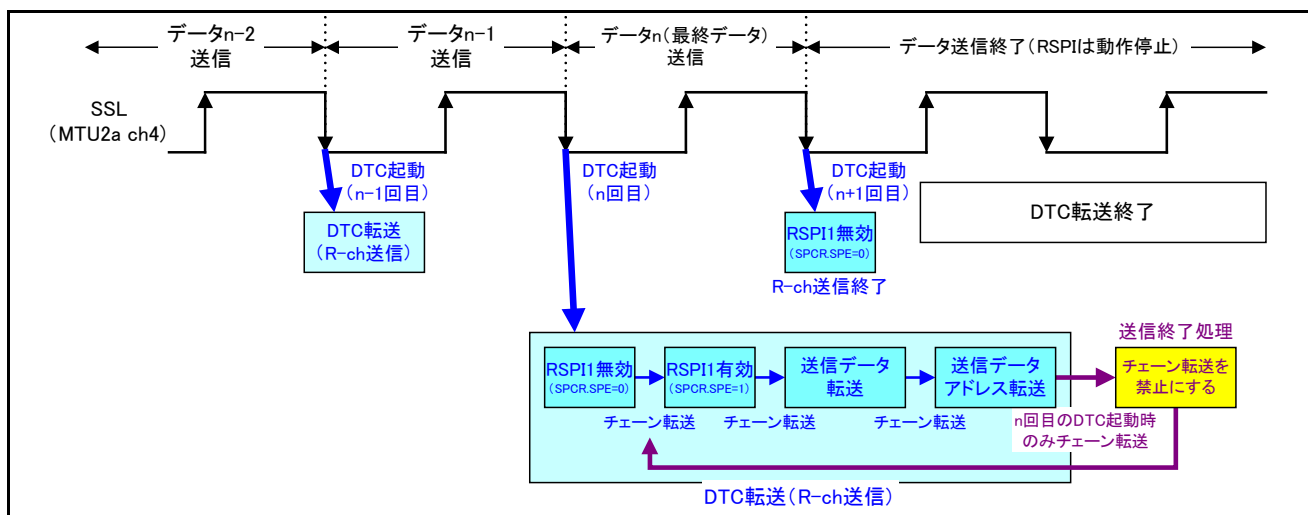


図5.19 R-ch送信のDTC転送による送信終了処理

## 5.1.3 受信動作

## 5.1.3.1 受信動作タイミング

図5.20に受信動作タイミング(標準フォーマット)を示します。

RSPIは外部機器からオーディオデータを受信します。RSPIはスレーブモード(SPI動作)、データ長32ビット、MSBファーストです。

外部機器からのオーディオデータはRSPI0およびRSPI1の両チャンネルに入力し、L-ch成分とR-ch成分に分解して受信します。RSPI0のスレーブセレクト極性をLowアクティブ、RSPI1のスレーブセレクト極性をHighアクティブに設定し、SSLによりRSPI0とRSPI1のチャンネルを選択します。アクティブとなったチャンネルのRSPIは、チャンネル切り替え後に発生する最初のSCKのアップエッジでデータを取り込み、以後SCKに同期して入力データを取り込みます。

RSPIによって取り込まれた入力データはDTCにより内蔵RAMに転送され、図5.1のように格納されます。

MTU2aは、WSに対して、SCKの約1ビット分位相が遅れたSSLを出力します。WSとSSLの位相差の詳細については、「5.1.1.5 ワードセレクト信号とスレーブセレクト信号の位相差」を参照してください。

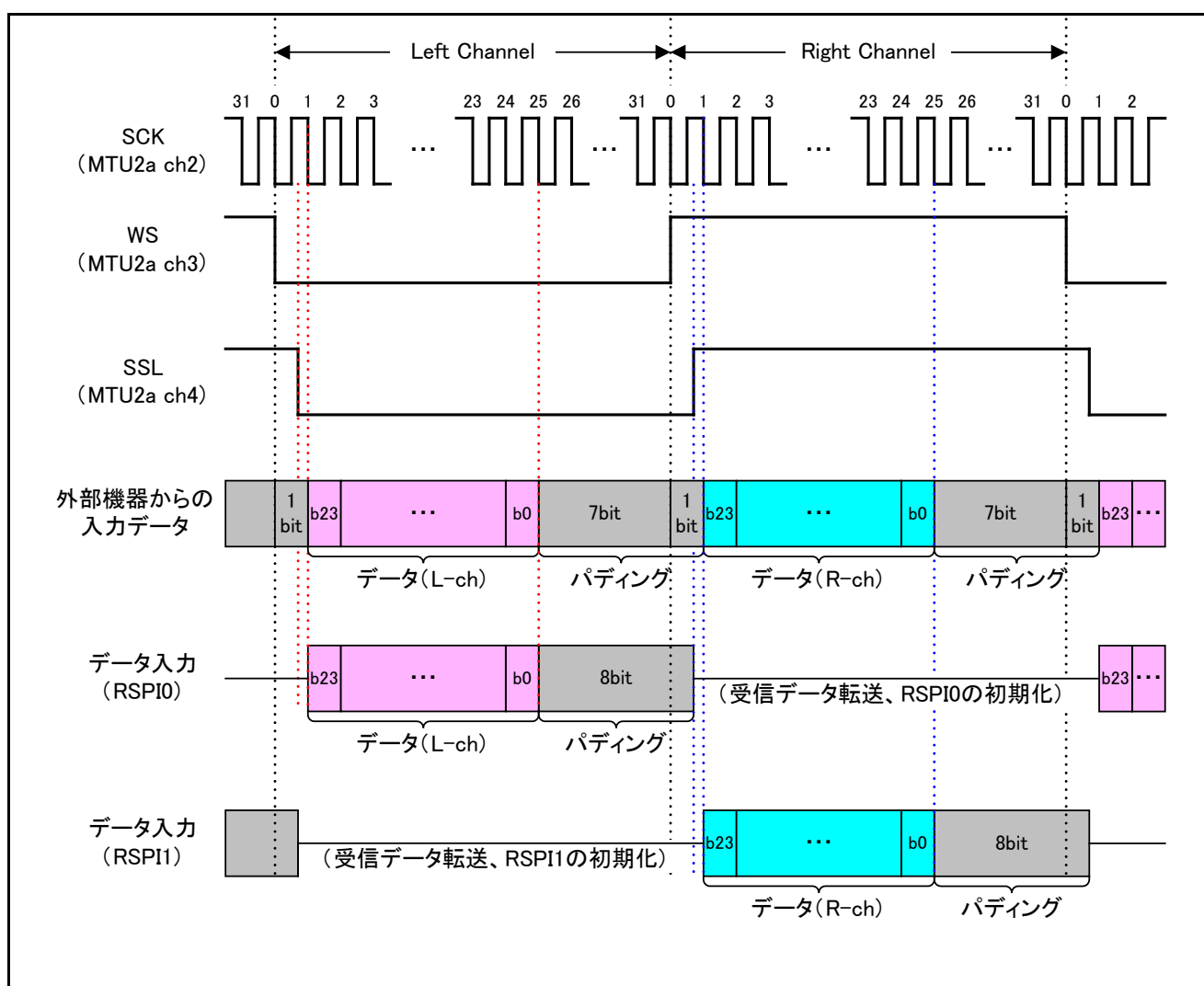


図5.20 受信動作タイミング(標準フォーマット)

## 5.1.3.2 受信動作時の DTC 動作

図 5.21 に受信動作時の DTC 動作を示します。

DTC は RSPI から内蔵 RAM に受信データを転送します。DTC 転送は L-ch 受信と R-ch 受信、RSPI0 初期化と RSPI1 初期化の 4 つがあります。データ受信の DTC は、RSPI の受信バッファフル割り込みで起動します。RSPI0 の受信バッファフル割り込みでは L-ch 受信の DTC、RSPI1 の受信バッファフル割り込みでは R-ch 受信の DTC が起動します。また、RSPI 初期化の DTC は、SSL のエッジ発生時のコンペアマッチ割り込みで起動します。SSL の立ち上がりエッジでは RSPI0 初期化の DTC、立ち下がりエッジでは RSPI1 初期化の DTC が起動します。

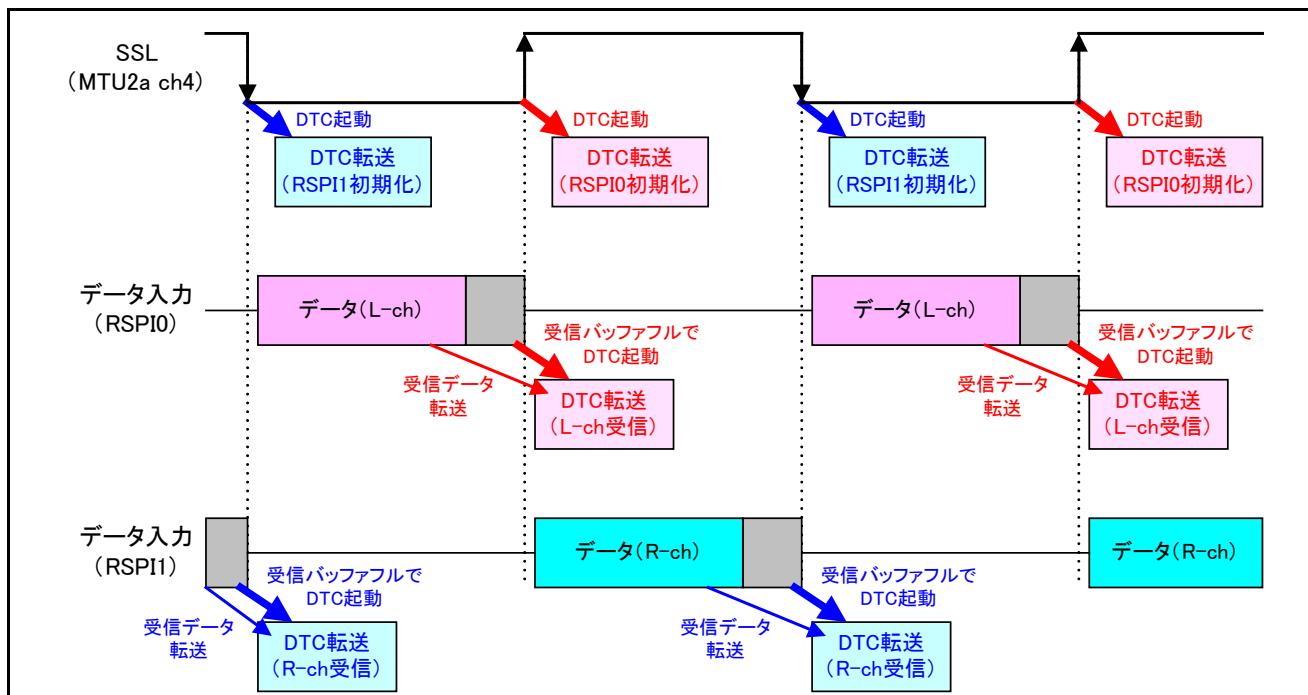


図5.21 受信動作時のDTC動作

図 5.22 に L-ch 受信の DTC 動作を、図 5.23 に R-ch 受信の DTC 動作を示します。

データ受信の DTC はチェーン転送により、受信データの転送、受信データアドレスの転送を行います。また、RSPI 初期化の DTC はチェーン転送により、RSPI の初期化を行います。なお、RSPI の初期化は、RSPI 制御レジスタ(SPCR)の RSPI 機能許可ビット(SPE)に“0”を書いて RSPI を無効化し、続いて“1”を書いて RSPI を再び有効化することによって行います。

注)RSPI 機能許可ビットによる初期化についてはハードウェアマニュアルを参照してください。

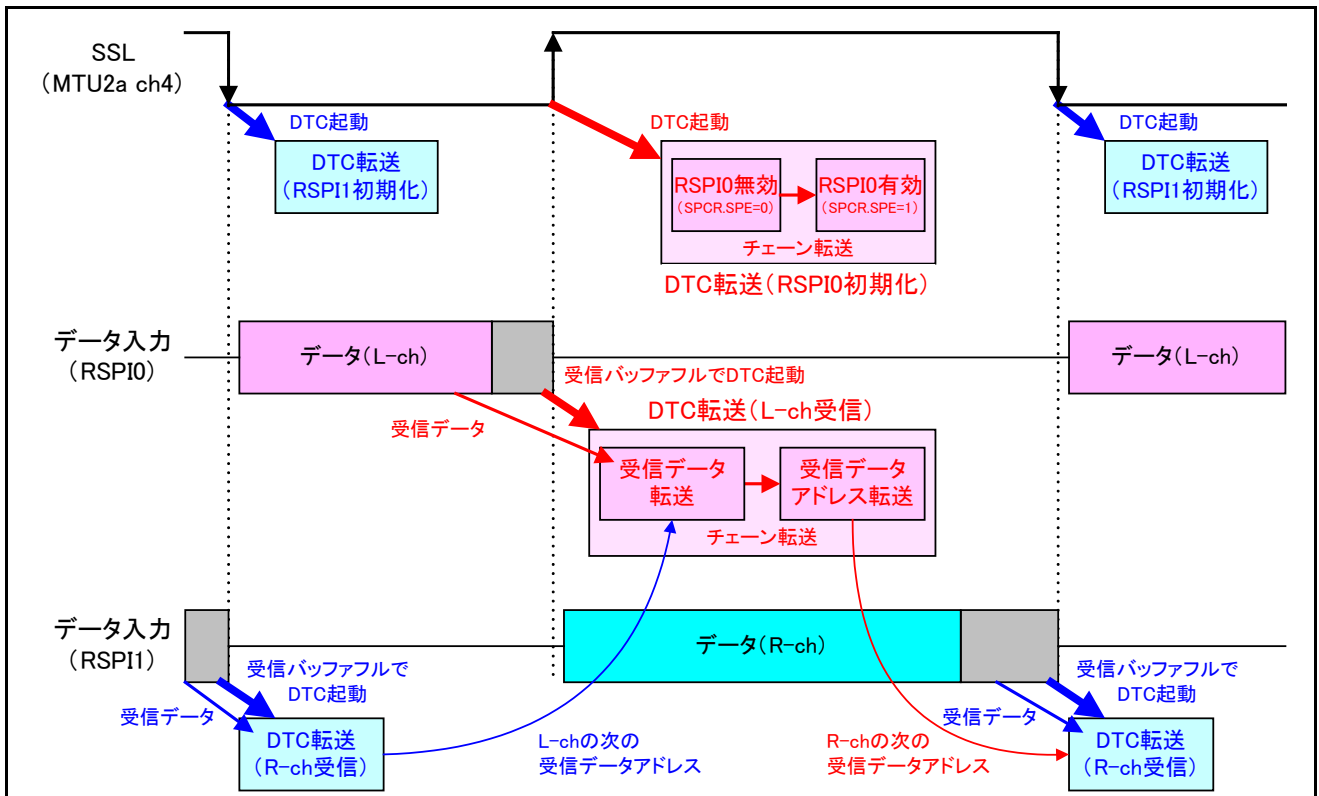


図5.22 L-ch受信のDTC動作

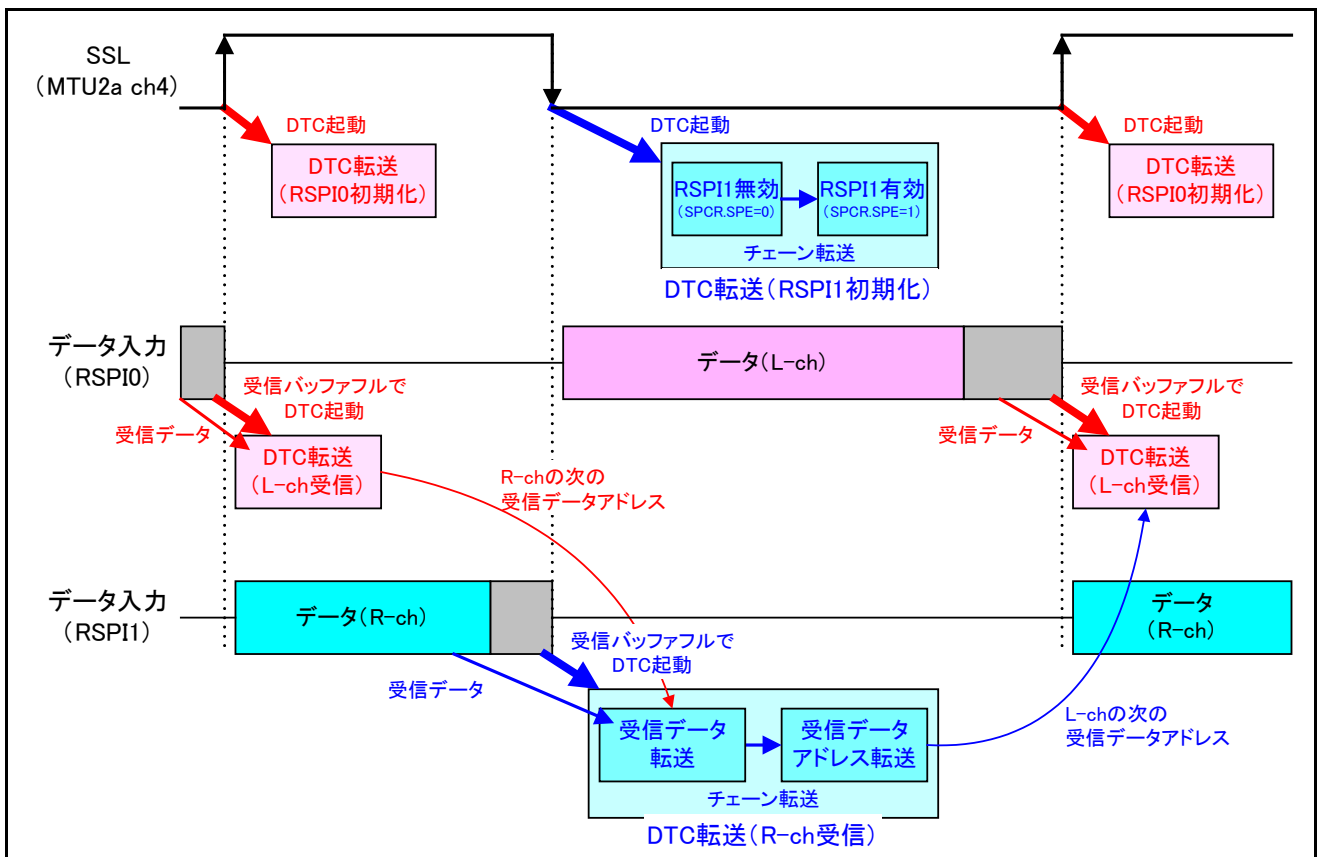


図5.23 R-ch受信のDTC動作



5.1.3.3 DTC による受信データアドレスの転送

図 5.24 に DTC による受信データアドレスの転送を示します。

本アプリケーションノートでは、受信したオーディオデータを「図 5.1 内蔵 RAM 上のオーディオデータ」に示した通り、L-ch 成分と R-ch 成分を 4 バイトずつ交互に内蔵 RAM 上に配置します。本アプリケーションノートでは、L-ch 受信の DTC 転送と R-ch 受信の DTC 転送の 2 つの DTC 転送を使用して、受信データの転送を交互に行っています。

図 5.24 に示した通り、L-ch 受信の DTC 転送は受信データのデータ 0(L-ch)を転送後、チェーン転送を用いて L-ch 受信の DTC 転送後の転送先アドレスを R-ch 受信の DTC 転送の転送先アドレスに転送します。R-ch 受信の DTC 転送は、L-ch 受信の DTC 転送後の転送先アドレスを開始アドレスとしてデータ 0(R-ch)を転送し、チェーン転送を用いて R-ch 受信の DTC 転送後の転送先アドレスを L-ch 受信の DTC 転送の転送先アドレスに転送します。

L-ch 受信の DTC 転送は、R-ch 受信のチェーン転送で転送された R-ch 受信の DTC 転送後の転送先アドレスを開始アドレスとしてデータ 1(L-ch)を転送し、チェーン転送を用いて L-ch 受信の DTC 転送後の転送先アドレスを次の R-ch 受信の DTC 転送に必要な開始アドレスとして R-ch 受信の DTC 転送の転送先アドレスに転送します。

このように L-ch 受信の DTC 転送は R-ch 受信の DTC 転送に必要な受信データアドレスを、R-ch 受信の DTC 転送は L-ch 受信の DTC 転送に必要な受信データアドレスを転送します。

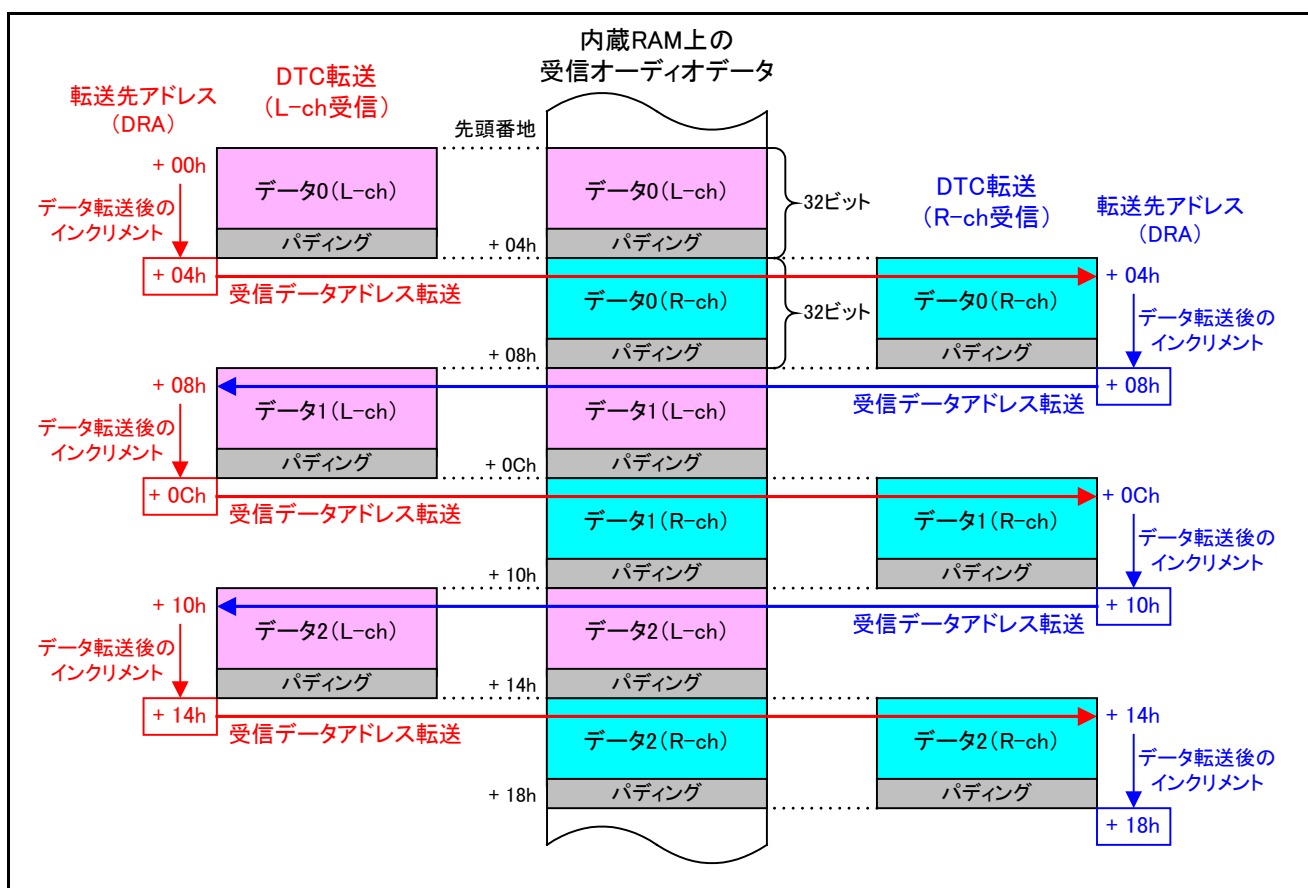


図5.24 DTCによる受信データアドレス転送

5.1.3.4 オーディオデータの受信終了

本アプリケーションノートでは、RSPI を無効化して動作を停止させることによりオーディオデータの受信終了とします。RSPI の無効化は RSPI0 初期化および RSPI1 初期化の DTC 転送が行います。

(a) L-ch 受信の受信終了処理

図 5.25 に L-ch 受信の受信終了動作の概略を示します。

RSPI0 初期化の DTC 転送は n-1 回目の起動時、チェーン転送を用いて受信終了処理を行います。受信終了処理により、n 回目の起動時は RSPI0 の無効化が行われ、RSPI0 は動作を停止します。

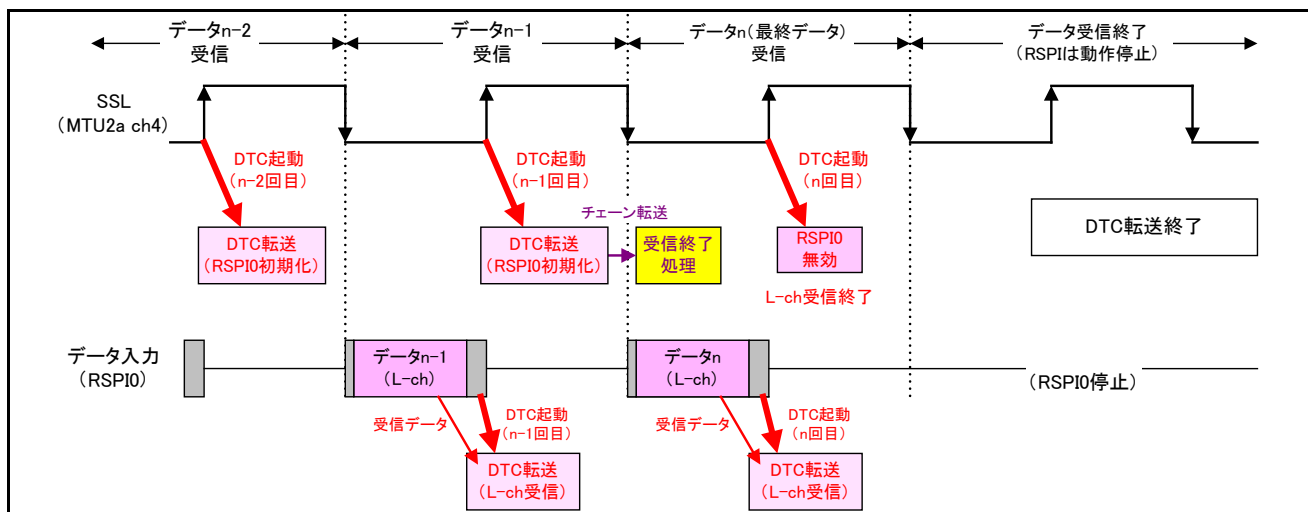


図5.25 L-ch受信の受信終了動作

L-ch受信の受信終了処理を行っているチェーン転送の詳細動作を図 5.26に示します。

RSPI0初期化のDTC転送は、「図5.22 L-ch受信のDTC動作」に示した通り、チェーン転送を用いて、RSPI0無効化、RSPI0有効化を行います。

このRSPI0有効化の転送情報にある転送カウンタをn-1回、DTCチェーン転送選択ビットを“転送カウンタ=0のときのみチェーン転送を行う”に設定することで、n-1回目の起動時のRSPI0有効化後のチェーン転送により受信終了処理を行います。

受信終了処理は、RSPI0無効化の転送情報にあるDTCチェーン転送イネーブルビットの値を書き換えることで、n回目の起動時におけるRSPI0無効化後のチェーン転送を禁止にします。これにより、n回目の起動時はRSPI0の無効化のみが実行され、RSPI0の動作が停止し受信動作を終了します。

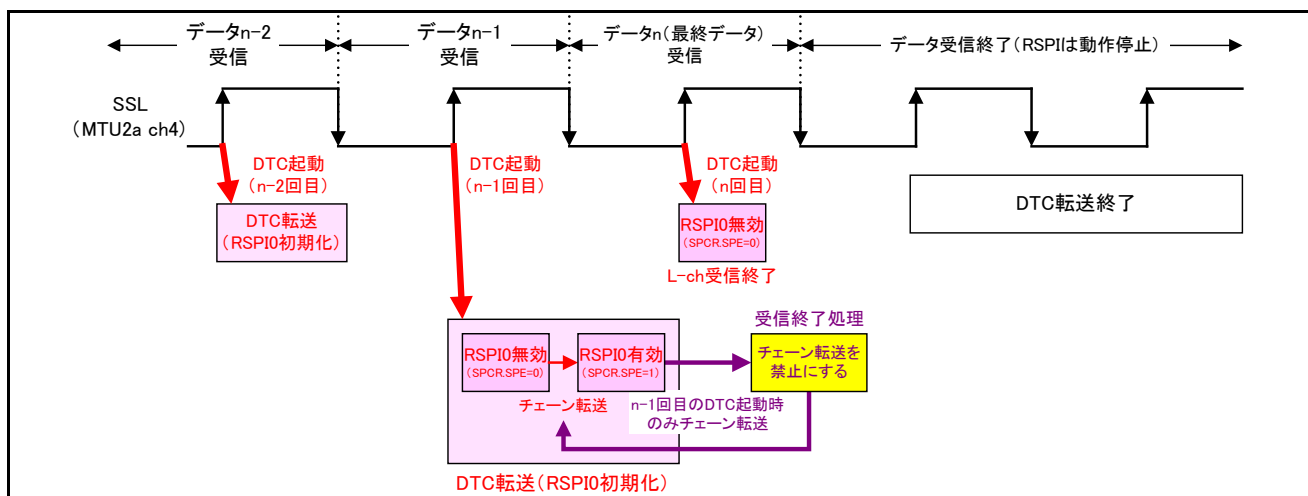


図5.26 RSPI0初期化のDTC転送による受信終了処理

(b) R-ch 受信の受信終了処理

図 5.27 に R-ch 受信の受信終了動作の概略を示します。

RSPI1 初期化の DTC 転送は n 回目の起動時、チェーン転送を用いて受信終了処理を行います。受信終了処理により、n+1 回目の起動時は RSPI1 の無効化が行われ、RSPI1 は動作を停止します。

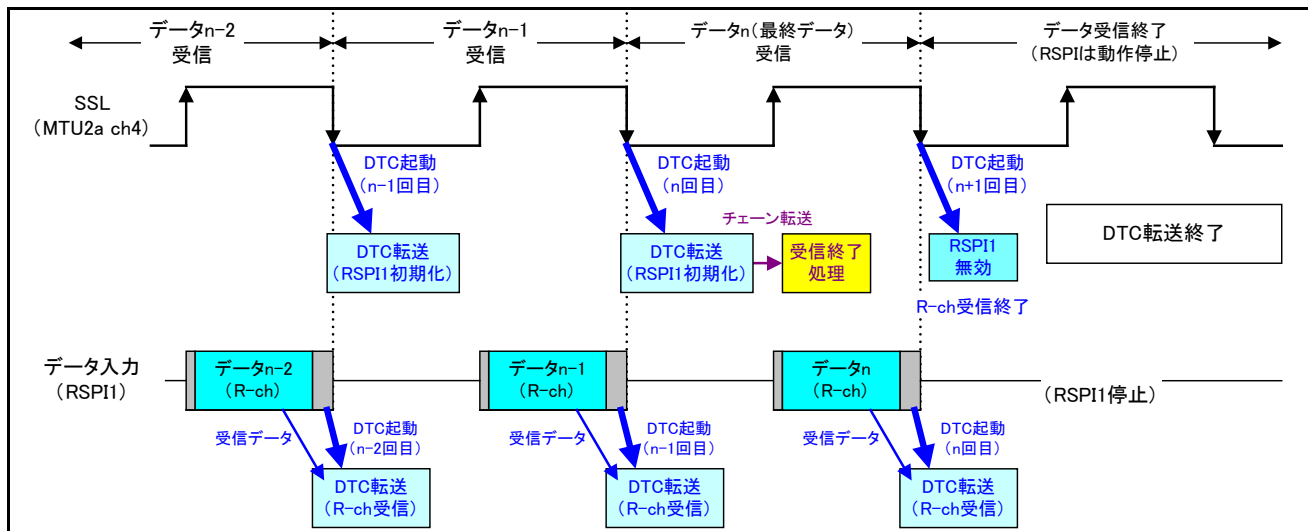


図5.27 R-ch受信の受信終了動作

R-ch受信の受信終了処理を行っているチェーン転送の詳細動作を図5.28に示します。

RSPI1初期化のDTC転送は、「図5.23 R-ch受信のDTC動作」に示した通り、チェーン転送を用いて、RSPI1無効化、RSPI1有効化を行います。

このRSPI1有効化の転送情報にある転送カウンタをn回、DTCチェーン転送選択ビットを“転送カウンタ=0のときのみチェーン転送を行う”に設定することで、n回目の起動時のRSPI1有効化後のチェーン転送により受信終了処理を行います。

受信終了処理は、RSPI1無効化の転送情報にあるDTCチェーン転送イネーブルビットの値を書き換えることで、n+1回目の起動時におけるRSPI1無効化後のチェーン転送を禁止にします。これにより、n+1回目の起動時はRSPI1の無効化のみが実行され、RSPI1の動作が停止し受信動作を終了します。

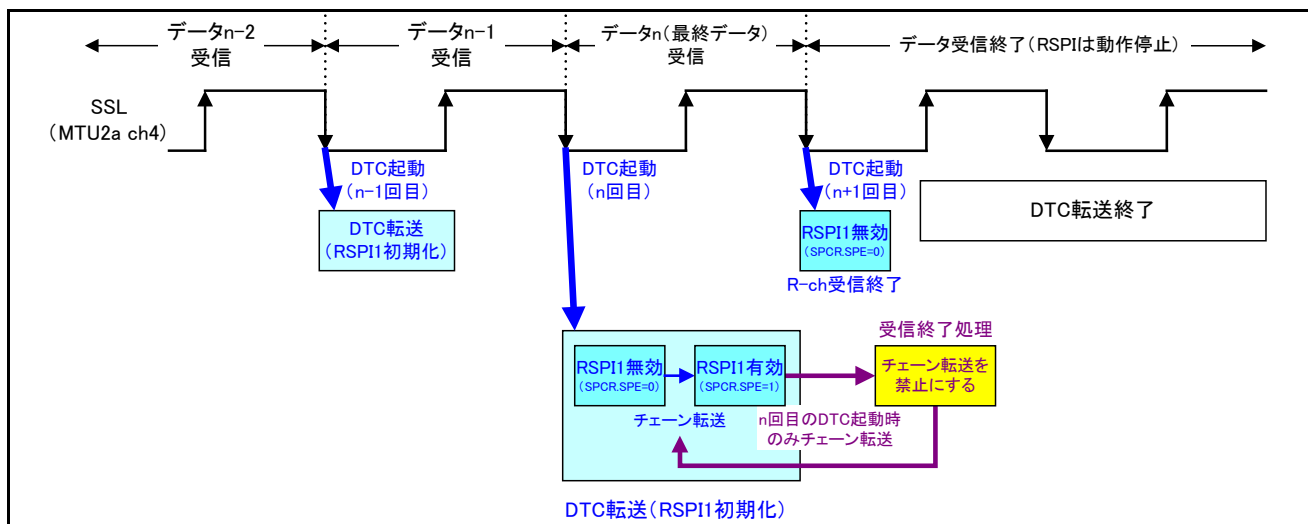


図 5.28 RSPI1 初期化の DTC 転送による受信終了処理

## 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                              | 備考              |
|----------------------------|---------------------------------|-----------------|
| main.c                     | メイン処理                           |                 |
| r_init_stop_module.c       | リセット後に動作している周辺機能の停止             |                 |
| r_init_stop_module.h       | r_init_stop_module.c のヘッダファイル   |                 |
| r_init_non_existent_port.c | 存在しないポートの初期設定                   |                 |
| r_init_non_existent_port.h | r_init_non_existent_port.c のヘッダ |                 |
| r_init_clock.c             | クロック初期設定                        |                 |
| r_init_clock.h             | r_init_clock.c のヘッダ             |                 |
| i2s_dtc.c                  | DTCa 動作設定                       |                 |
| i2s_mtu2.c                 | MTU2a 動作設定                      |                 |
| i2s_rsipi.c                | RSPI動作設定                        | 送受信データフォーマットの設定 |
| i2s.h                      | I2S 通信のマクロ定義                    |                 |

## 5.3 オプション設定メモリ

表 5.2 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 5.2 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<br>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<br>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 5.3 にサンプルコードで使用する定数を示します。

表 5.3 サンプルコードで使用する定数

| 定数名                | 設定値   | 内容  |
|--------------------|---|---|
| L_CH               | 0   | 内蔵 RAM のオーディオデータ配列の L-ch Index No.          |
| R_CH               | 1   | 内蔵 RAM のオーディオデータ配列の R-ch Index No.          |
| NULL_DATA          | 0x00000000  | 受信オーディオデータ領域の初期化データ                         |
| TRANSMIT_FORMAT    | STANDARD_PADDING<br>or<br>BACKWARD_PADDING<br>or<br>FORWARD_PADDING | 送受信データフォーマットの選択<br>(標準、後方パディング、前方パディングから選択) |
| STANDARD_PADDING   | 0   | 標準フォーマット                                    |
| BACKWARD_PADDING   | 1   | 後方パディングフォーマット                               |
| FORWARD_PADDING    | 2   | 前方パディングフォーマット                               |
| TRANSMIT_MODE      | 0   | 送信動作モード                                     |
| RECEIVE_MODE       | 1   | 受信動作モード                                     |
| TRANSCIEVE_MODE    | 2   | 送受信動作モード                                    |
| AUD_SIZE           | 0x100   | オーディオデータの総バイト数                              |
| AUD_NUM            | $(AUD\_SIZE / 4) / 2$   | オーディオデータ(L-ch/R-ch)のデータ数                    |
| DTC_TX_CHAIN_SIZE  | 5   | 送信動作の DTCa チェーン転送の転送情報数                     |
| DTC_RX_CHAIN_SIZE  | 2   | 受信動作の DTCa チェーン転送の転送情報数                     |
| DTC_TX_L_COUNT     | AUD_NUM - 1   | DTC 転送(L-ch 送信、RSPI0 初期化)の転送回数              |
| DTC_TX_R_COUNT     | AUD_NUM   | DTC 転送(R-ch 送信、RSPI1 初期化)の転送回数              |
| DTC_RX_L_COUNT     | AUD_NUM   | DTC 転送(L-ch 受信)の転送回数                        |
| DTC_RX_R_COUNT     | AUD_NUM   | DTC 転送(R-ch 受信)の転送回数                        |
| SCK_CYCLE_VALUE    | 0x0003  | SCK の周期                                     |
| WS_SSL_CYCLE_VALUE | 0x00FF  | WS、SSL の周期                                  |
| SSL_DELAY_VALUE_S  | 0x0003  | WS と SSL の位相値(標準フォーマット)                     |
| SSL_DELAY_VALUE_B  | 0x0001  | WS と SSL の位相値<br>(後方パディングフォーマット)            |
| SSL_DELAY_VALUE_F  | 0x001F  | WS と SSL の位相値<br>(前方パディングフォーマット)            |

## 5.5 構造体/共用体一覧

図 5.29 にサンプルコードで仕様する構造体/共用体を示します。

```

#ifdef _LIT
struct st_dtc_data{
    unsigned char    wk[2]; /* リトルエンディアン */
    unsigned char    MRB; /* DTCモードレジスタB */
    unsigned char    MRA; /* DTCモードレジスタA */
    unsigned long    SAR; /* DTC転送元アドレスレジスタ */
    unsigned long    DAR; /* DTC転送先アドレスレジスタ */
    unsigned short   CRB; /* DTC転送カウントレジスタB */
    unsigned short   CRA; /* DTC転送カウントレジスタA */
};
#endif
#ifdef _BIG
struct st_dtc_data{
    unsigned char    MRA; /* ビッグエンディアン */
    unsigned char    MRB; /* DTCモードレジスタB */
    unsigned char    wk[2]; /* 予約領域 */
    unsigned long    SAR; /* DTC転送元アドレスレジスタ */
    unsigned long    DAR; /* DTC転送先アドレスレジスタ */
    unsigned short   CRA; /* DTC転送カウントレジスタA */
    unsigned short   CRB; /* DTC転送カウントレジスタB */
};
#endif

struct st_dtc_data DTC_TX_L[DTC_TX_CHAIN_SIZE]; /* DTC転送情報(L-ch送信) */
struct st_dtc_data DTC_TX_R[DTC_TX_CHAIN_SIZE]; /* DTC転送情報(R-ch送信) */
struct st_dtc_data DTC_RX_L[DTC_RX_CHAIN_SIZE]; /* DTC転送情報(L-ch受信) */
struct st_dtc_data DTC_RX_R[DTC_RX_CHAIN_SIZE]; /* DTC転送情報(R-ch受信) */

```

図 5.29 サンプルコードで仕様する構造体/共用体

## 5.6 変数一覧

表 5.4 にグローバル変数、表 5.5 に const 型変数を示します。

表 5.4 グローバル変数

| 型             | 変数名                    | 内容          | 使用関数  |
|---------------|------------------------|-------------|---|
| unsigned long | tx_au_data[AUD_NUM][2] | 送信オーディオデータ  | i2s_au_data_init<br>i2s_start<br>i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| unsigned long | rx_au_data[AUD_NUM]    | 受信オーディオデータ  | i2s_au_data_init<br>i2s_dtc_rx_l_init<br>i2s_dtc_rx_r_init              |
| unsigned long | DTC_VECT_TABLE[256]    | DTC ベクタテーブル | i2s_dtc_init  |

表 5.5 const 型変数

| 型             | 変数名               | 内容                            | 使用関数                                   |
|---------------|-------------------|-------------------------------|--|
| const uint8_t | RSPI_TX_DISABLE   | RSPI 初期化用レジスタ設定値<br>(送信動作)    | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | RSPI_TX_ENABLE    | RSPI 初期化解除用レジスタ設定値<br>(送信動作)  | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | RSPI_RX_DISABLE   | RSPI 初期化用レジスタ設定値<br>(受信動作)    | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | RSPI_RX_ENABLE    | RSPI 初期化解除用レジスタ設定値<br>(受信動作)  | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | RSPI_TRX_DISABLE  | RSPI 初期化用レジスタ設定値<br>(送受信動作)   | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | RSPI_TRX_ENABLE   | RSPI 初期化解除用レジスタ設定値<br>(送受信動作) | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |
| const uint8_t | DTC_CHAIN_DISABLE | チェーン転送禁止用レジスタ設定値              | i2s_dtc_tx_l_init<br>i2s_dtc_tx_r_init |

## 5.7 関数一覧

表 5.6 にサンプルコードで使用する関数を示します。

表 5.6 サンプルコードで使用する関数

| 関数名                    | 説明                            |
|------------------------|-------------------------------|
| main                   | メイン処理                         |
| 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                      |
| i2s_au_data_init       | 内蔵 RAM のオーディオデータ作成            |
| i2s_start              | I2S 通信開始処理                    |
| i2s_dtc_init           | DTC 初期設定                      |
| i2s_dtc_tx_l_init      | DTC 転送情報作成(L-ch 送信、RSPI0 初期化) |
| i2s_dtc_tx_r_init      | DTC 転送情報作成(R-ch 送信、RSPI1 初期化) |
| i2s_dtc_rx_l_init      | DTC 転送情報作成(L-ch 受信)           |
| i2s_dtc_rx_r_init      | DTC 転送情報作成(R-ch 受信)           |
| i2s_mtu2_init          | MTU2a 初期設定                    |
| i2s_mtu2_ch2_init      | MTU2a チャンネル 2 設定(SCK 生成)      |
| i2s_mtu2_ch3_init      | MTU2a チャンネル 3 設定(WS 生成)       |
| i2s_mtu2_ch4_init      | MTU2a チャンネル 4 設定(SSL 生成)      |
| i2s_rspi_init          | RSPI 初期設定                     |
| i2s_rspi0_init         | RSPI0 初期設定(L-ch 送受信)          |
| i2s_rspi1_init         | RSPI1 初期設定(R-ch 送受信)          |



## 5.8 関数仕様

サンプルコードの関数仕様を示します。

| main                   |   |
|------------------------|---|
| 概要                     | I2S 通信動作メイン関数   |
| ヘッダ                    | i2s.h   |
| 宣言                     | void main (void)  |
| 説明                     | 各関数(ポートの初期設定、クロックの初期設定、周辺機能(MTU2a、RSPI、DTC)、の初期設定をコールします。   |
| 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。<br>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.1.00」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。<br>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。<br>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.1.00」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。<br>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.1.00」を参照してください。 |
| i2s_au_data_init |  |
| 概要               | 内蔵 RAM のオーディオデータを作成します。  |
| ヘッダ              | i2s.h  |
| 宣言               | void i2s_au_data_init (void)   |
| 説明               | 送信オーディオデータを作成します。(tx_au_data)<br>受信オーディオデータ領域を 0 クリアします。(rx_au_data)   |
| 引数               | なし   |
| リターン値            | なし   |
| i2s_start        |  |
| 概要               | I2S 通信の開始処理を行います。  |
| ヘッダ              | i2s.h, iodef.h   |
| 宣言               | void i2s_start (char i2s_mode)   |
| 説明               | RSPI 機能を有効にします。<br>DTC モジュールを起動します。<br>MTU2a のカウント動作を開始します。  |
| 引数               | ●第一引数 : i2s_mode : 動作モードの選択  |
| リターン値            | なし   |
| i2s_dtc_init     |  |
| 概要               | DTC の初期設定を行います。  |
| ヘッダ              | iodefine.h   |
| 宣言               | void i2s_dtc_init (char i2s_mode)  |
| 説明               | DTC ベクタテーブルを作成します。<br>割り込みによる DTC 起動を許可します。  |
| 引数               | ●第一引数 : i2s_mode : 動作モードの選択  |
| リターン値            | なし   |

| i2s_dtc_tx_l_init |  |
|-------------------|--|
| 概要                | DTC 転送情報を作成します。(L-ch 送信、RSPi0 初期化)     |
| ヘッダ               | i2s.h, iodef.h                         |
| 宣言                | void i2s_dtc_tx_l_init (char i2s_mode) |
| 説明                | 送信データを内蔵 RAM から RSPi0 に転送します。          |
| 引数                | ●第一引数 : i2s_mode : 動作モードの選択            |
| リターン値             | なし                                     |
| i2s_dtc_tx_r_init |  |
| 概要                | DTC 転送情報を作成します。(R-ch 送信、RSPi1 初期化)     |
| ヘッダ               | i2s.h, iodef.h                         |
| 宣言                | void i2s_dtc_tx_r_init (char i2s_mode) |
| 説明                | 送信データを内蔵 RAM から RSPi1 に転送します。          |
| 引数                | ●第一引数 : i2s_mode : 動作モードの選択            |
| リターン値             | なし                                     |
| i2s_dtc_rx_l_init |  |
| 概要                | DTC 転送情報を作成します。(L-ch 受信)               |
| ヘッダ               | i2s.h, iodef.h                         |
| 宣言                | void i2s_dtc_rx_l_init (void)          |
| 説明                | 受信データを RSPi0 から内蔵 RAM に転送します。          |
| 引数                | なし                                     |
| リターン値             | なし                                     |
| i2s_dtc_rx_r_init |  |
| 概要                | DTC 転送情報を作成します。(R-ch 受信)               |
| ヘッダ               | i2s.h, iodef.h                         |
| 宣言                | void i2s_dtc_rx_r_init (void)          |
| 説明                | 受信データを RSPi1 から内蔵 RAM に転送します。          |
| 引数                | なし                                     |
| リターン値             | なし                                     |
| i2s_mtu2_init     |  |
| 概要                | MTU2a の初期設定を行います。                      |
| ヘッダ               | ioodef.h                               |
| 宣言                | void i2s_mtu2_init (void)              |
| 説明                | MTU2a のポート設定を行います。                     |
| 引数                | なし                                     |
| リターン値             | なし                                     |

---

**i2s\_mtu2\_ch2\_init**

---

|       |                                   |
|-------|-----------------------------------|
| 概要    | MTU2a チャンネル 2 の初期設定を行います。(SCK 生成) |
| ヘッダ   | i2s.h, iodef.h                    |
| 宣言    | void i2s_mtu2_ch2_init (void)     |
| 説明    | 外部クロックから SCK を生成し、出力します。          |
| 引数    | なし                                |
| リターン値 | なし                                |

---

**i2s\_mtu2\_ch3\_init**

---

|       |                                  |
|-------|----------------------------------|
| 概要    | MTU2a チャンネル 3 の初期設定を行います。(WS 生成) |
| ヘッダ   | i2s.h, iodef.h                   |
| 宣言    | void i2s_mtu2_ch3_init (void)    |
| 説明    | 外部クロックから WS を生成し、出力します。          |
| 引数    | なし                               |
| リターン値 | なし                               |

---

**i2s\_mtu2\_ch4\_init**

---

|       |   |
|-------|---|
| 概要    | MTU2a チャンネル 4 の初期設定を行います。(SSL 生成)               |
| ヘッダ   | i2s.h, iodef.h                                  |
| 宣言    | void i2s_mtu2_ch4_init (void)                   |
| 説明    | 外部クロックから SSL を生成し、出力します。<br>コンペアマッチ割り込みを有効にします。 |
| 引数    | なし  |
| リターン値 | なし  |

---

**i2s\_rspi\_init**

---

|       |                                    |
|-------|------------------------------------|
| 概要    | RSPI の初期設定を行います。                   |
| ヘッダ   | なし                                 |
| 宣言    | void i2s_rspi_init (char i2s_mode) |
| 説明    | RSPI の設定を行います。                     |
| 引数    | ●第一引数 : i2s_mode : 動作モードの選択        |
| リターン値 | なし                                 |

---

**i2s\_rspi0\_init**

---

|       |                                     |
|-------|-------------------------------------|
| 概要    | RSPI0 の初期設定を行います。(L-ch 送受信)         |
| ヘッダ   | ioodef.h                            |
| 宣言    | void i2s_rspi0_init (char i2s_mode) |
| 説明    | RSPI0 の設定を行います。                     |
| 引数    | ●第一引数 : i2s_mode : 動作モードの選択         |
| リターン値 | なし                                  |

---

**i2s\_rspi1\_init**

---

|       |                                     |            |
|-------|-------------------------------------|------------|
| 概要    | RSPI1 の初期設定を行います。(R-ch 送受信)         |            |
| ヘッダ   | iodef.h                             |            |
| 宣言    | void i2s_rspi1_init (char i2s_mode) |            |
| 説明    | RSPI1 の設定を行います。                     |            |
| 引数    | ●第一引数 : i2s_mode                    | : 動作モードの選択 |
| リターン値 | なし                                  |            |

## 6. フローチャート

## 6.1 main 関数

図 6.1 に main 関数のフローチャートを示します。

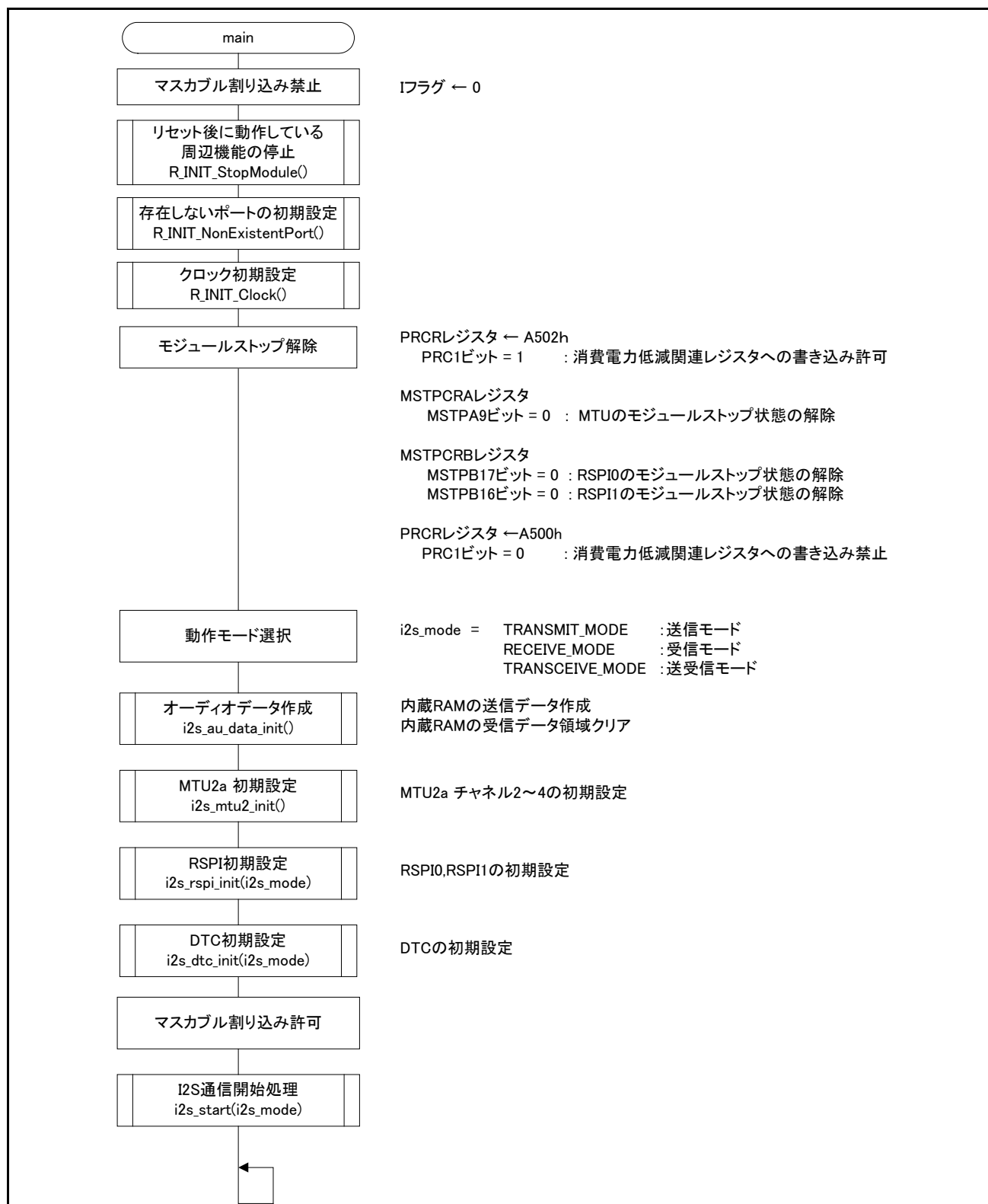


図 6.1 メイン処理

## 6.2 関数

### 6.2.1 i2s\_au\_data\_init 関数

図 6.2 に i2s\_au\_data\_init 関数のフローチャートを示します。

i2s\_au\_data\_init 関数は内蔵 RAM のオーディオデータを作成します。

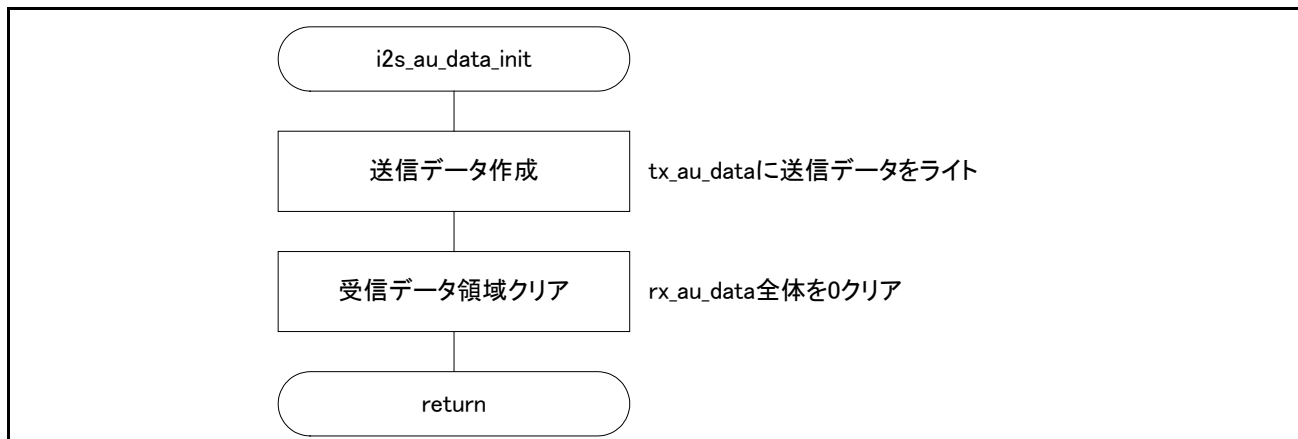


図 6.2 オーディオデータ作成

6.2.2 i2s\_start 関数

図 6.3 に i2s\_start 関数のフローチャートを示します。  
 i2s\_start 関数は周辺機能を有効に設定し I2S 通信を開始します。

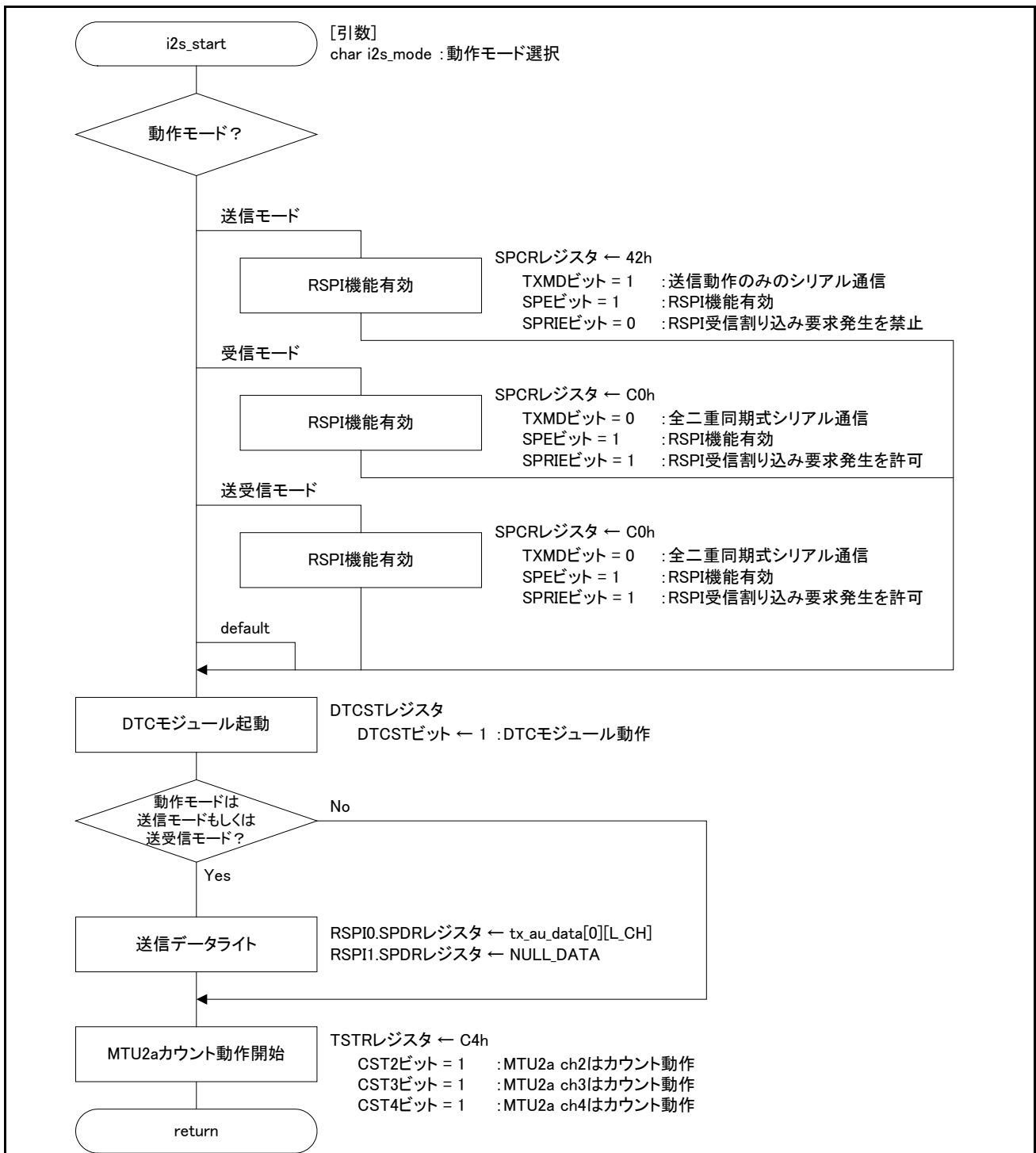


図 6.3 I2S 通信開始処理



6.2.3 i2s\_dtc\_init 関数

図 6.4 に i2s\_dtc\_init 関数のフローチャートを示します。  
i2s\_dtc\_init 関数は DTCa の初期設定を行います。

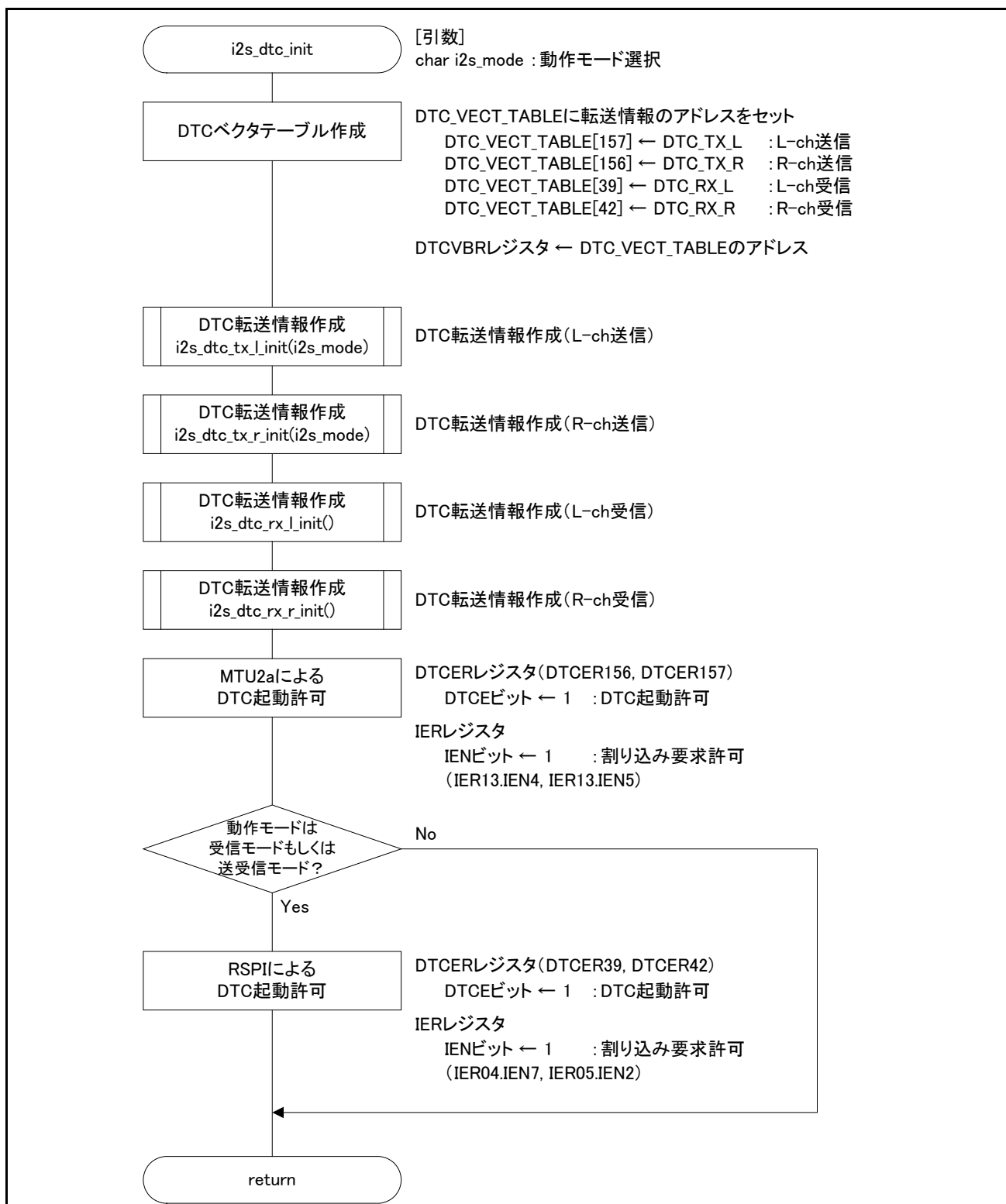


図 6.4 DTC の初期設定

6.2.4 i2s\_dtc\_tx\_l\_init 関数

図 6.5～図 6.8 に i2s\_dtc\_tx\_l\_init 関数のフローチャートを示します。

i2s\_dtc\_tx\_l\_init 関数は L-ch 送信(送信モード、送受信モード)または RSPI0 初期化(受信モード)の DTC 転送情報を作成します。

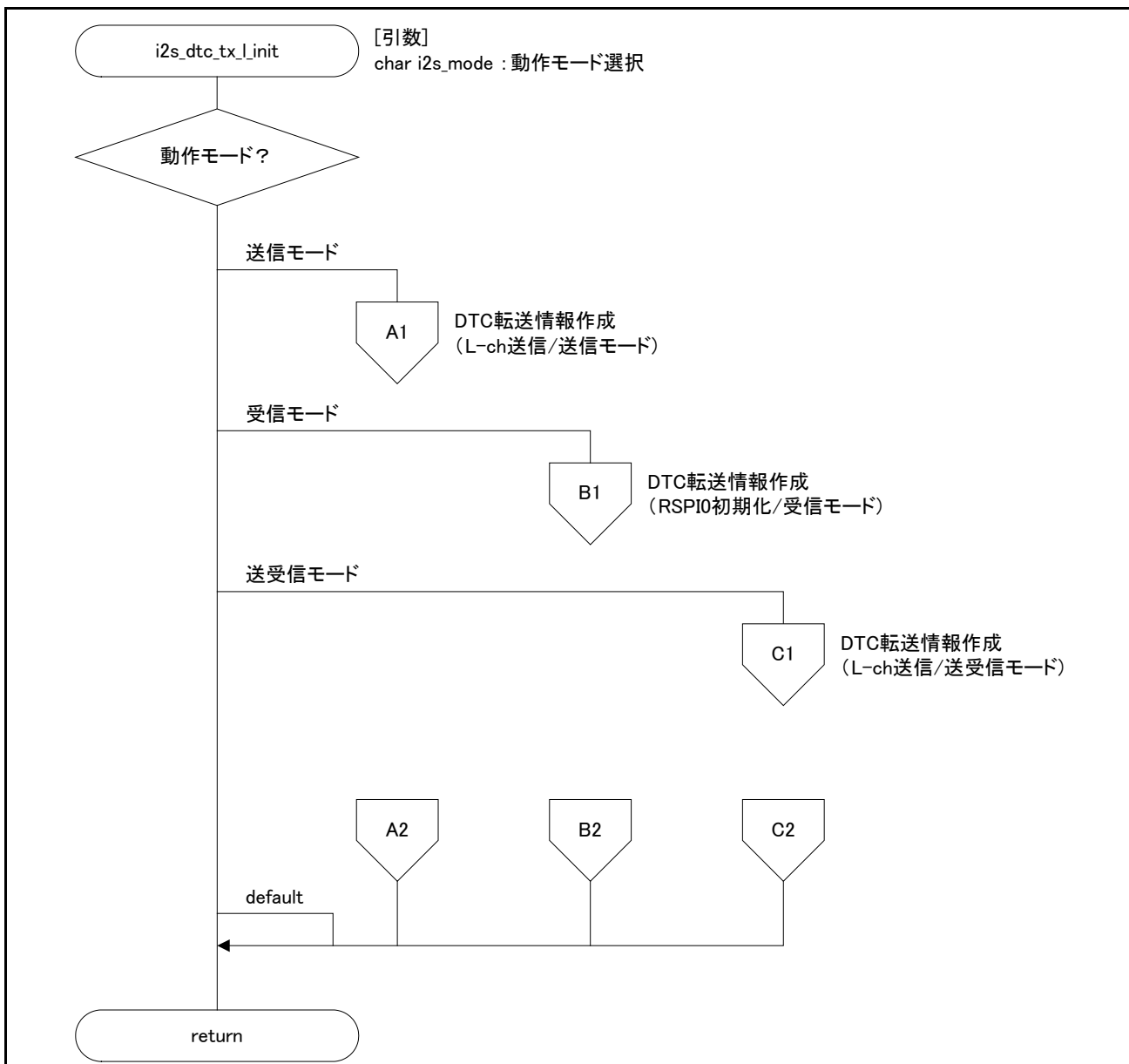


図 6.5 DTC 転送情報作成(L-ch 送信、RSPI0 初期化)(1/4)

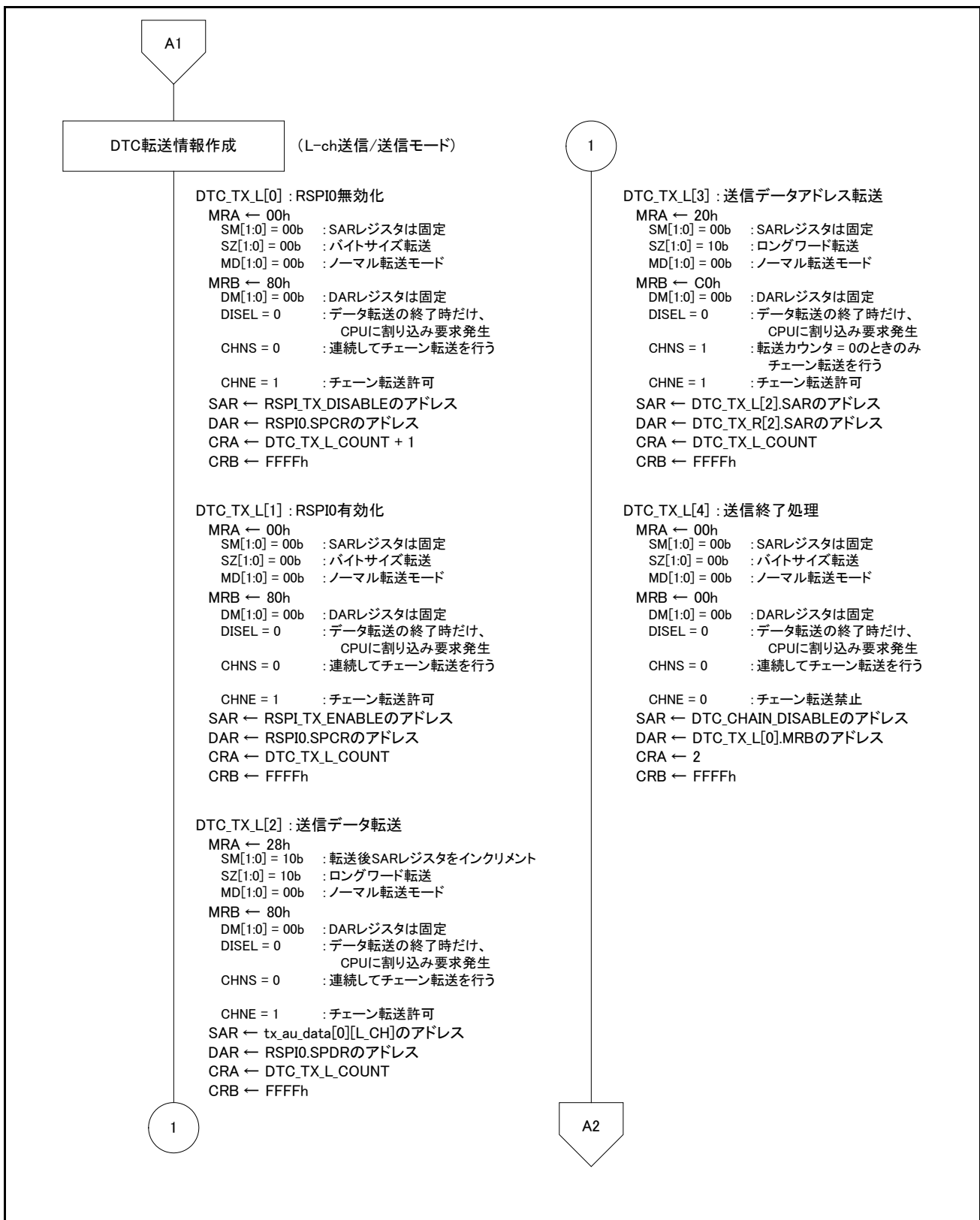


図 6.6 DTC 転送情報作成(L-ch 送信、RSPI0 初期化)(2/4)

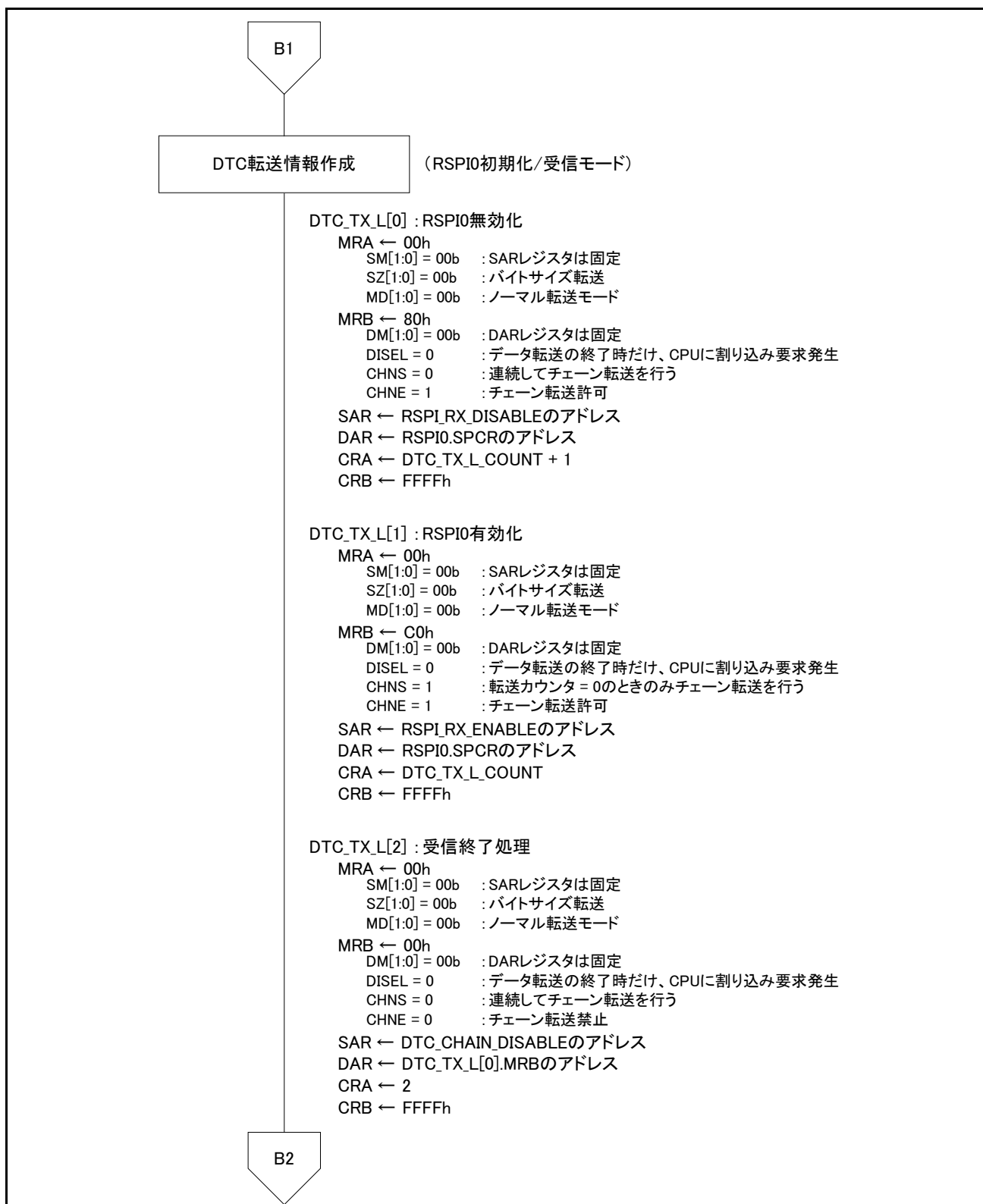


図 6.7 DTC 転送情報作成(L-ch 送信、RSPI0 初期化)(3/4)

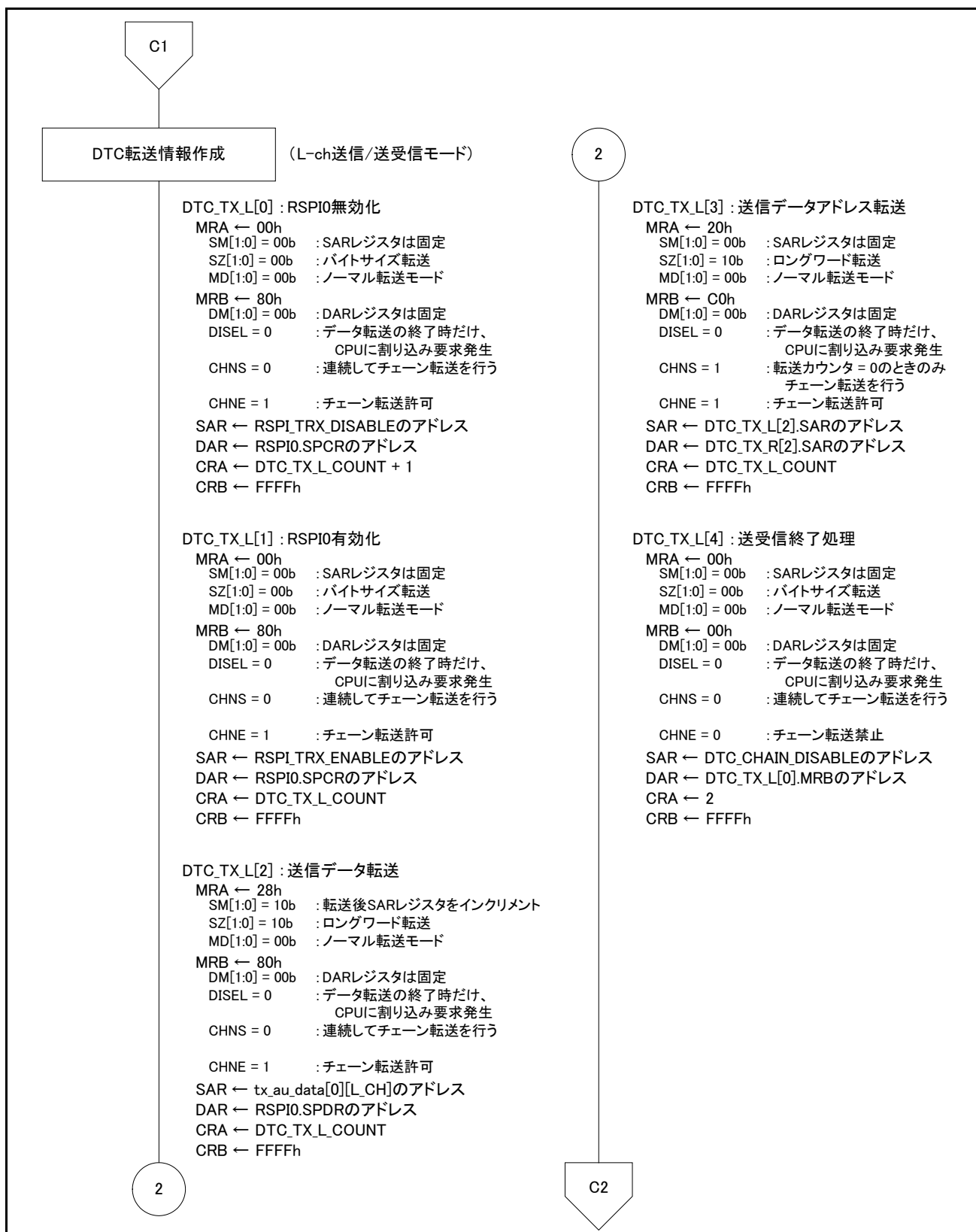


図 6.8 DTC 転送情報作成(L-ch 送信、RSPI0 初期化)(4/4)

6.2.5 i2s\_dtc\_tx\_r\_init 関数

図 6.9～図 6.12 に i2s\_dtc\_tx\_r\_init 関数のフローチャートを示します。

i2s\_dtc\_tx\_r\_init 関数は R-ch 送信(送信モード、送受信モード)または RSPI1 初期化(受信モード)の DTC 転送情報を作成します。

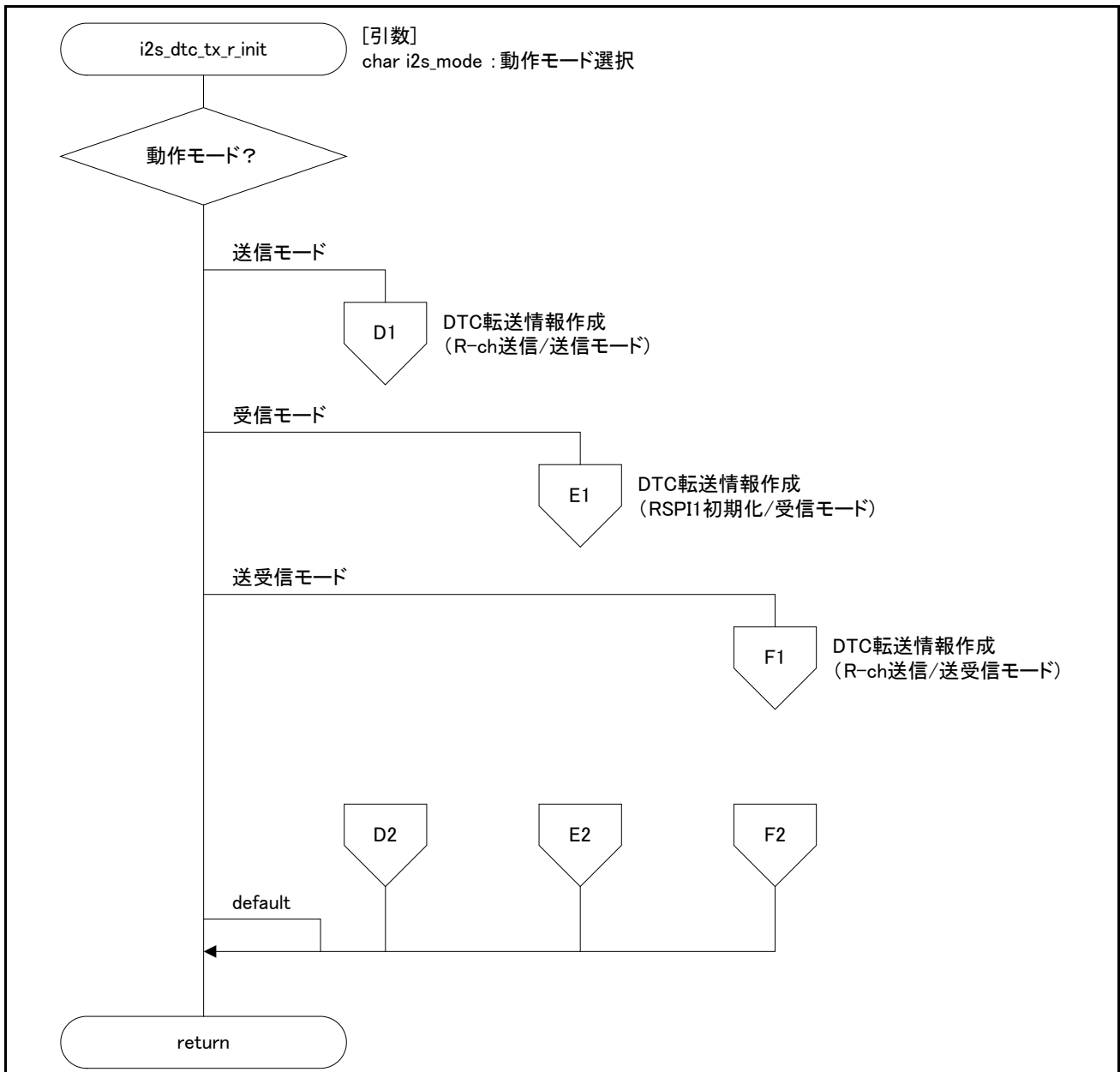


図 6.9 DTC 転送情報作成(R-ch 送信、RSPI1 初期化)(1/4)



図 6.10 DTC 転送情報作成(R-ch 送信、RSPI1 初期化)(2/4)



図 6.11 DTC 転送情報作成(R-ch 送信、RSPI1 初期化)(3/4)





図 6.12 DTC 転送情報作成(R-ch 送信、RSPI1 初期化)(4/4)

## 6.2.6 i2s\_dtc\_rx\_l\_init 関数

図 6.13 に i2s\_dtc\_rx\_l\_init 関数のフローチャートを示します。

i2s\_dtc\_rx\_l\_init 関数は L-ch 受信の DTC 転送情報を作成します。

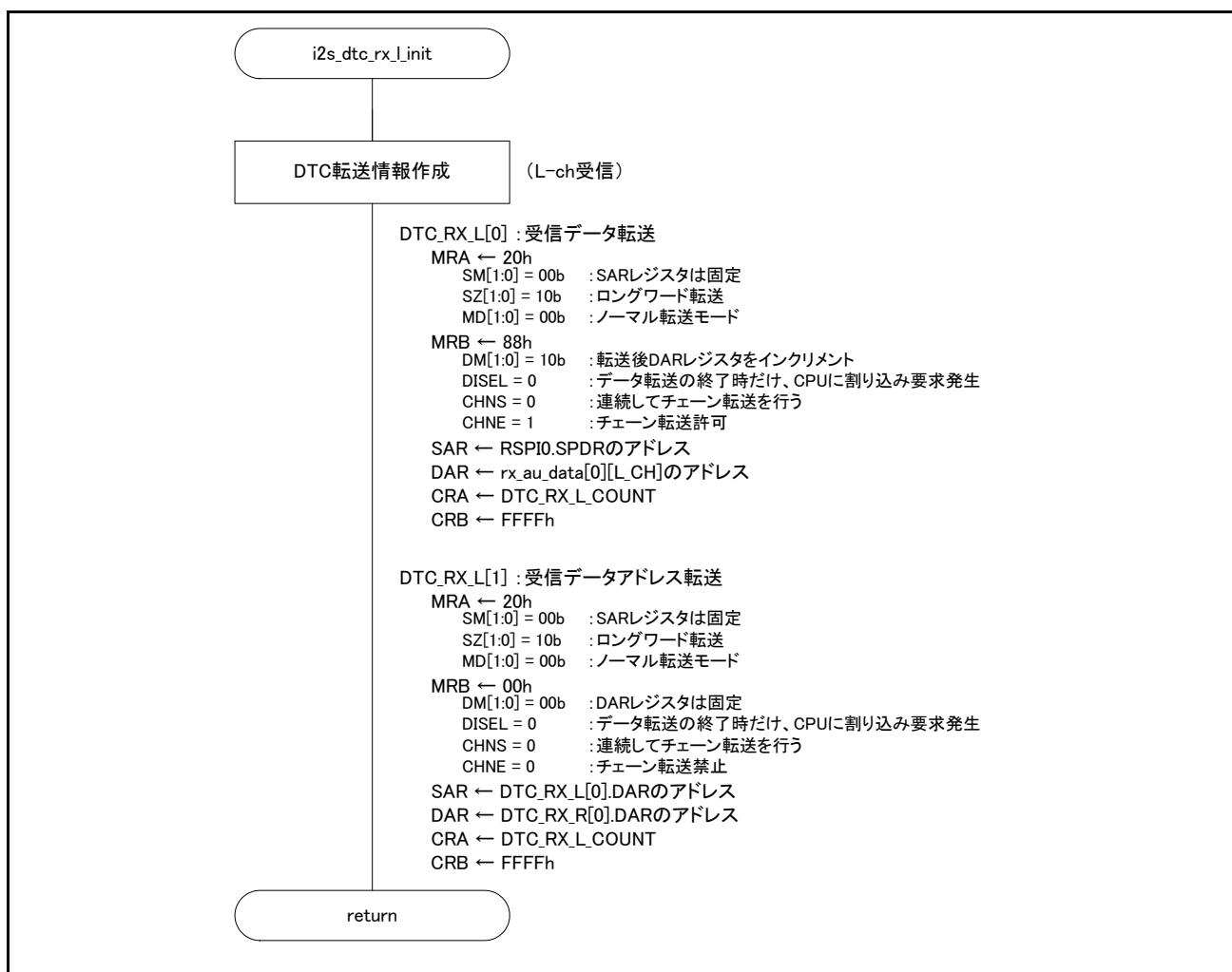


図 6.13 DTC 転送情報作成(L-ch 受信)

## 6.2.7 i2s\_dtc\_rx\_r\_init 関数

図 6.14 に i2s\_dtc\_rx\_r\_init 関数のフローチャートを示します。

i2s\_dtc\_rx\_r\_init 関数は R-ch 受信の DTC 転送情報を作成します。

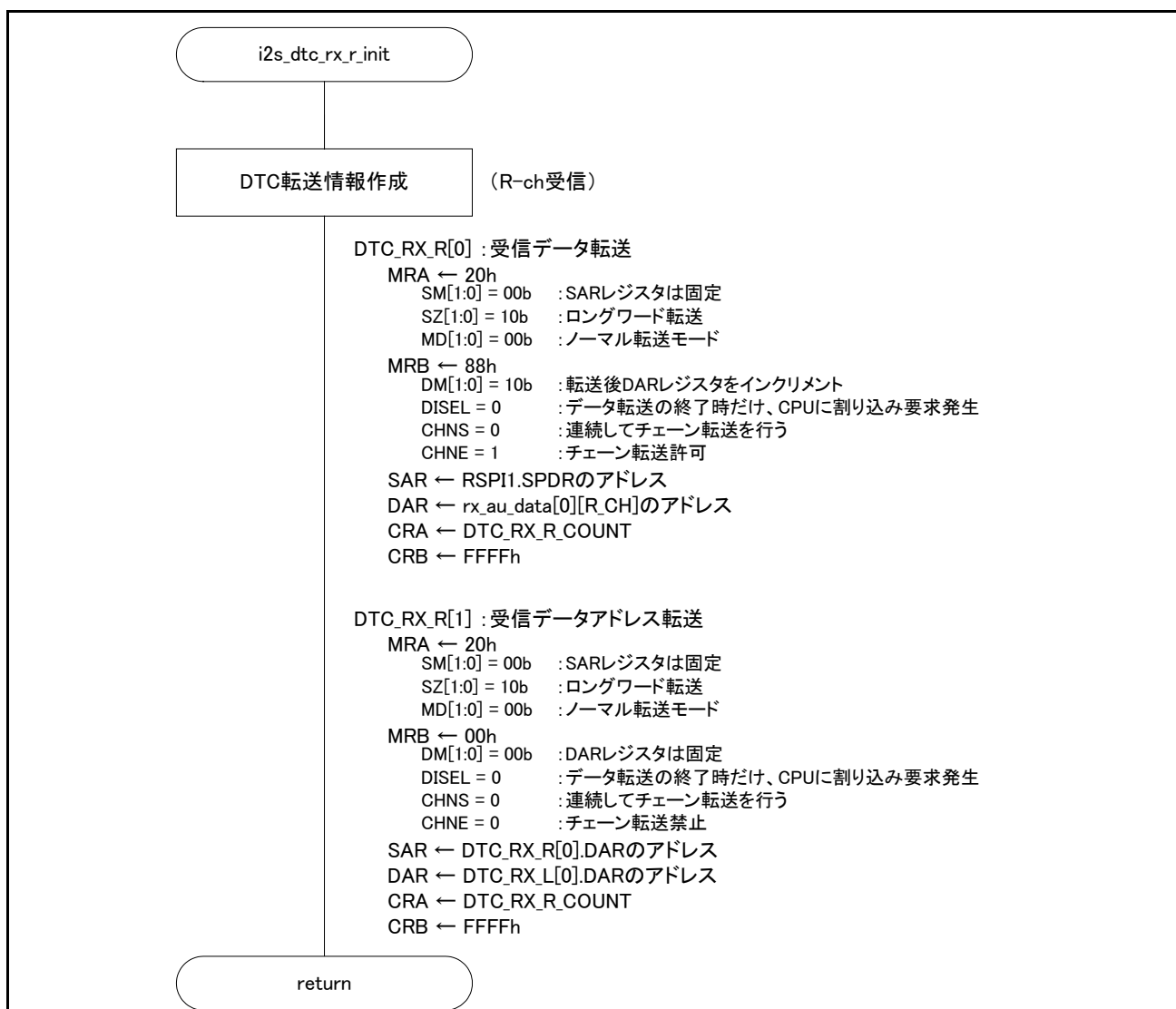


図 6.14 DTC 転送情報作成(R-ch 受信)

6.2.8 i2s\_mtu2\_init 関数

図 6.15 に i2s\_mtu2\_init 関数のフローチャートを示します。

i2s\_mtu2\_init 関数は MTU2a の初期設定を行います。

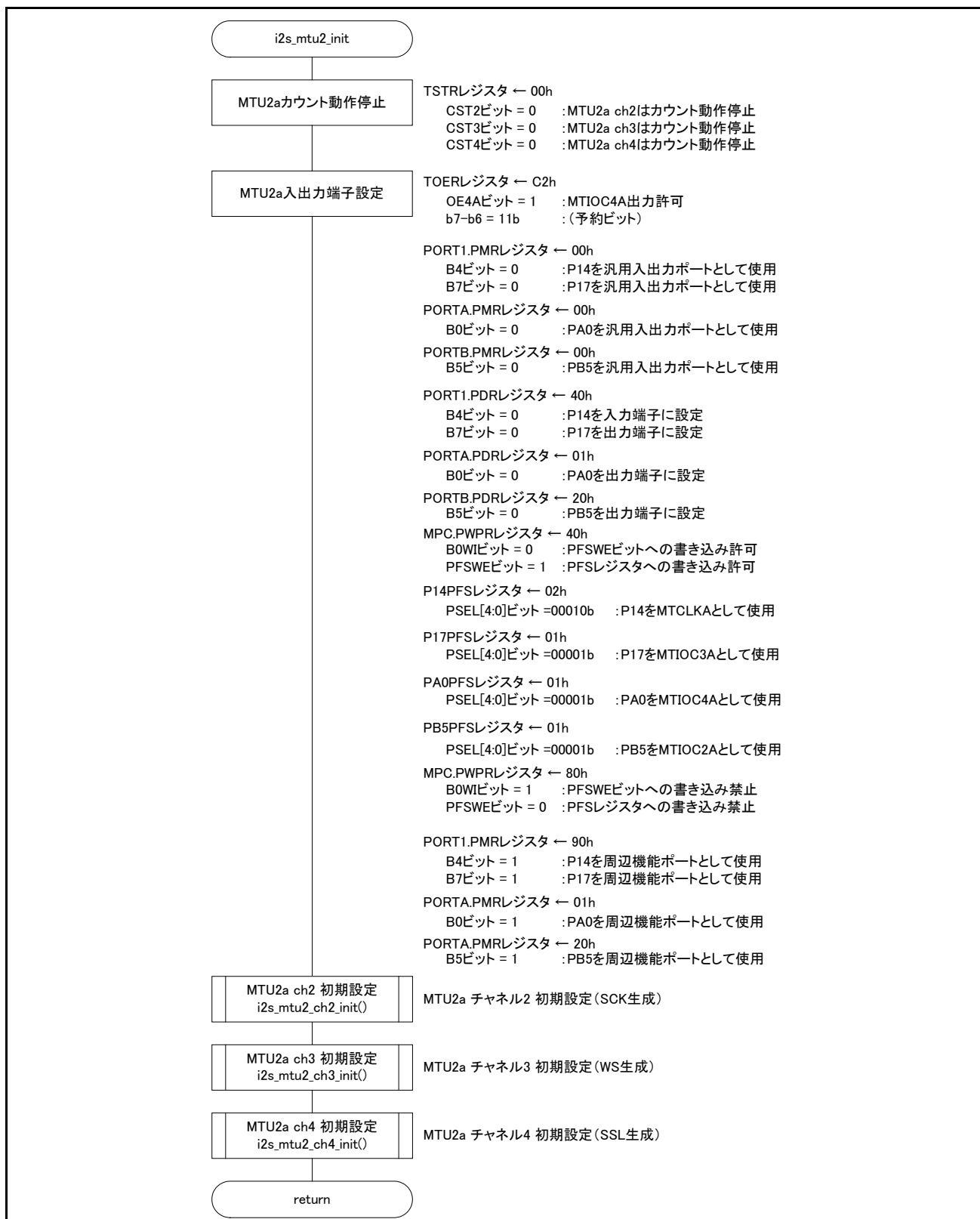


図 6.15 MTU2a 初期設定

## 6.2.9 i2s\_mtu2\_ch2\_init 関数

図 6.16 に i2s\_mtu2\_ch2\_init 関数のフローチャートを示します。

i2s\_mtu2\_ch2\_init 関数は、MTU2a チャンネル 2 を使用して SCK を生成する設定を行います。

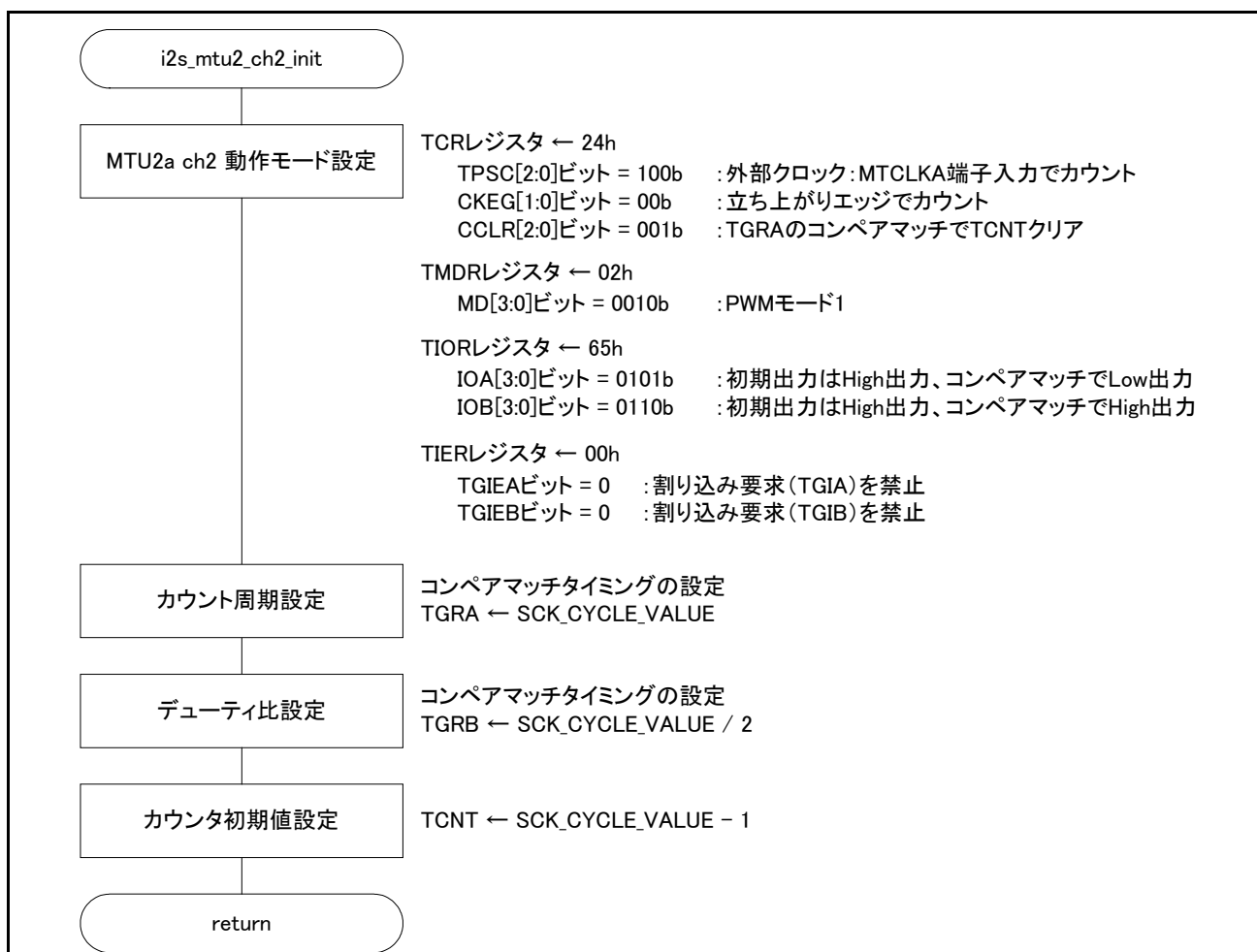


図 6.16 MTU2a チャンネル 2 設定(SCK 生成)

## 6.2.10 i2s\_mtu2\_ch3\_init 関数

図 6.17 に i2s\_mtu2\_ch3\_init 関数のフローチャートを示します。

i2s\_mtu2\_ch3\_init 関数は、MTU2a チャンネル 3 を使用して WS を生成する設定を行います。

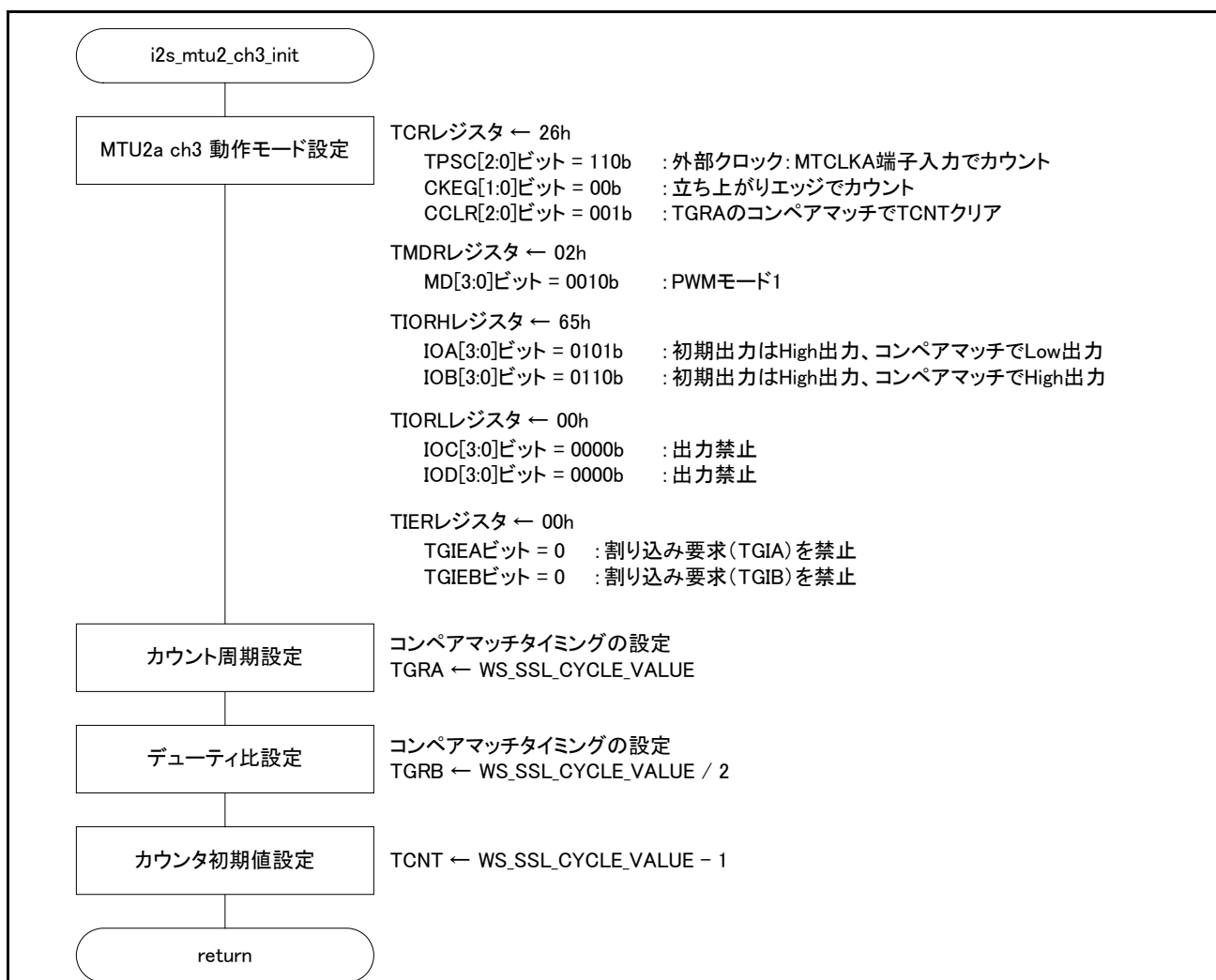


図 6.17 MTU2a チャンネル 3 設定(WS 生成)

## 6.2.11 i2s\_mtu2\_ch4\_init 関数

図 6.18 に i2s\_mtu2\_ch4\_init 関数のフローチャートを示します。

i2s\_mtu2\_ch4\_init 関数は、MTU2a チャンネル 4 を使用して SSL を生成する設定を行います。

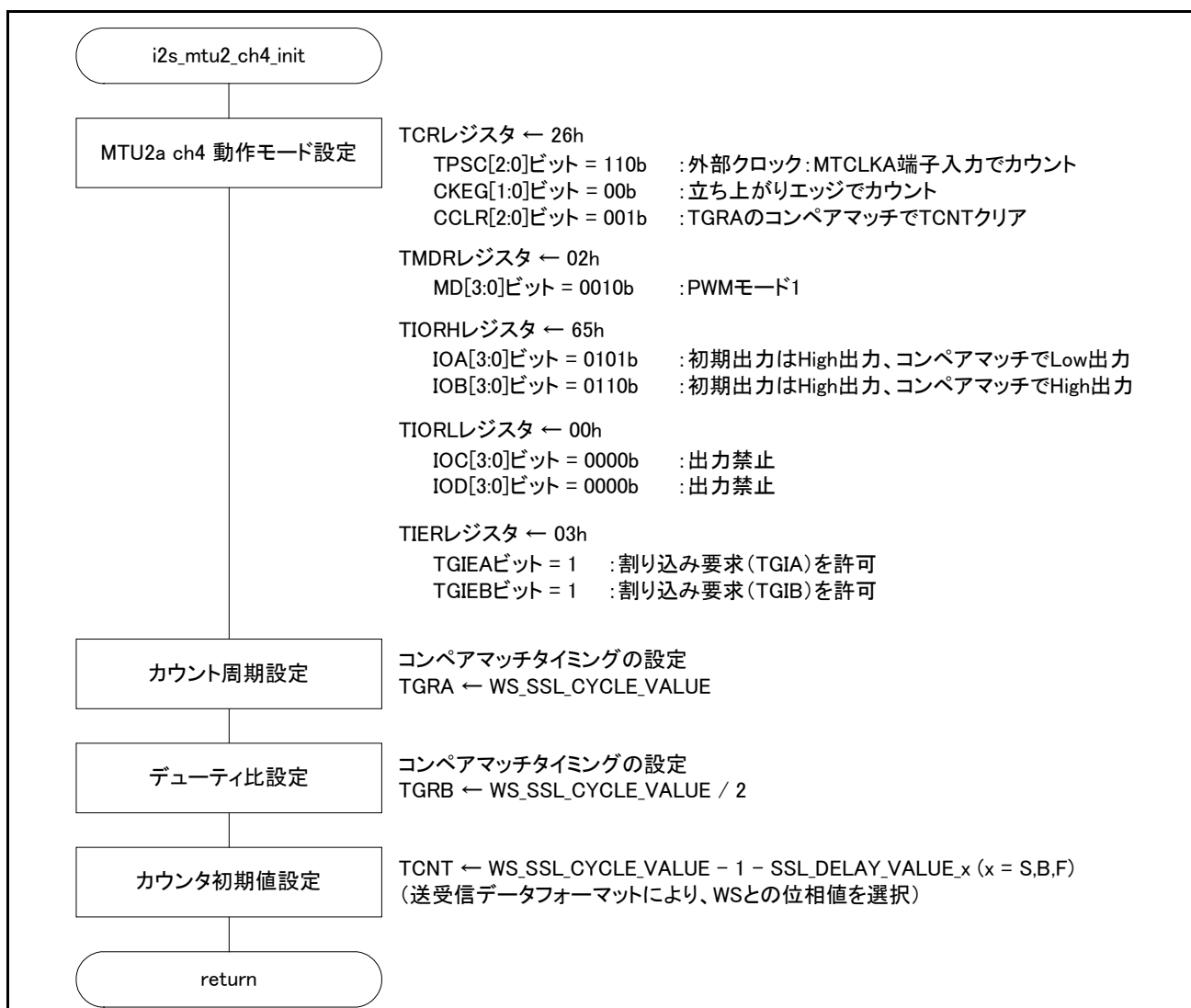


図 6.8 MTU2a チャンネル 4 設定(SSL 生成)

## 6.2.12 i2s\_rspi\_init 関数

図 6.19 に i2s\_rspi\_init 関数のフローチャートを示します。

i2s\_rspi\_init 関数は RSPI の初期設定を行います。

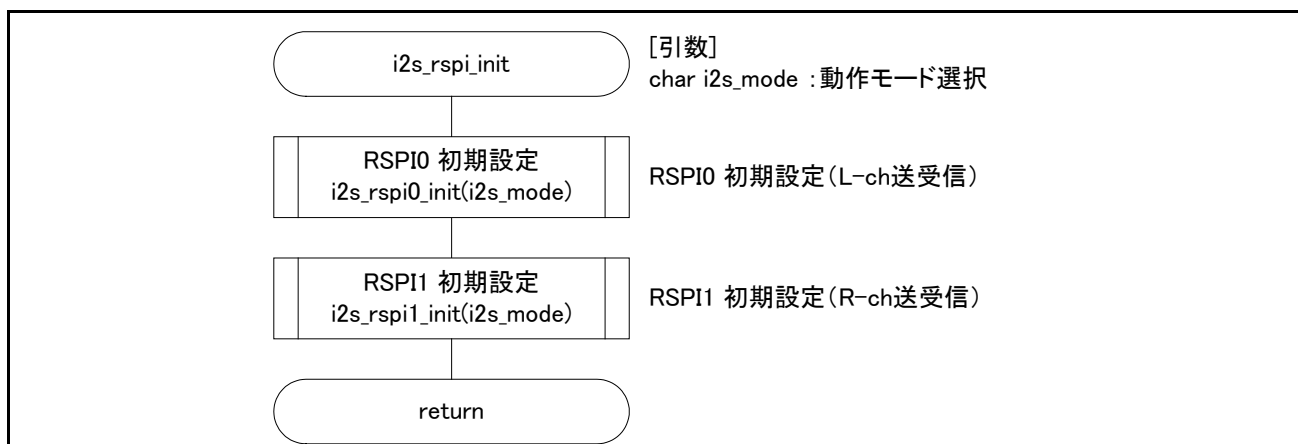


図 6.19 RSPI 初期設定



## 6.2.13 i2s\_rspi0\_init 関数

図 6.20、図 6.21 に i2s\_rspi0\_init 関数のフローチャートを示します。  
i2s\_rspi0\_init 関数は RSPI0 の初期設定を行います。

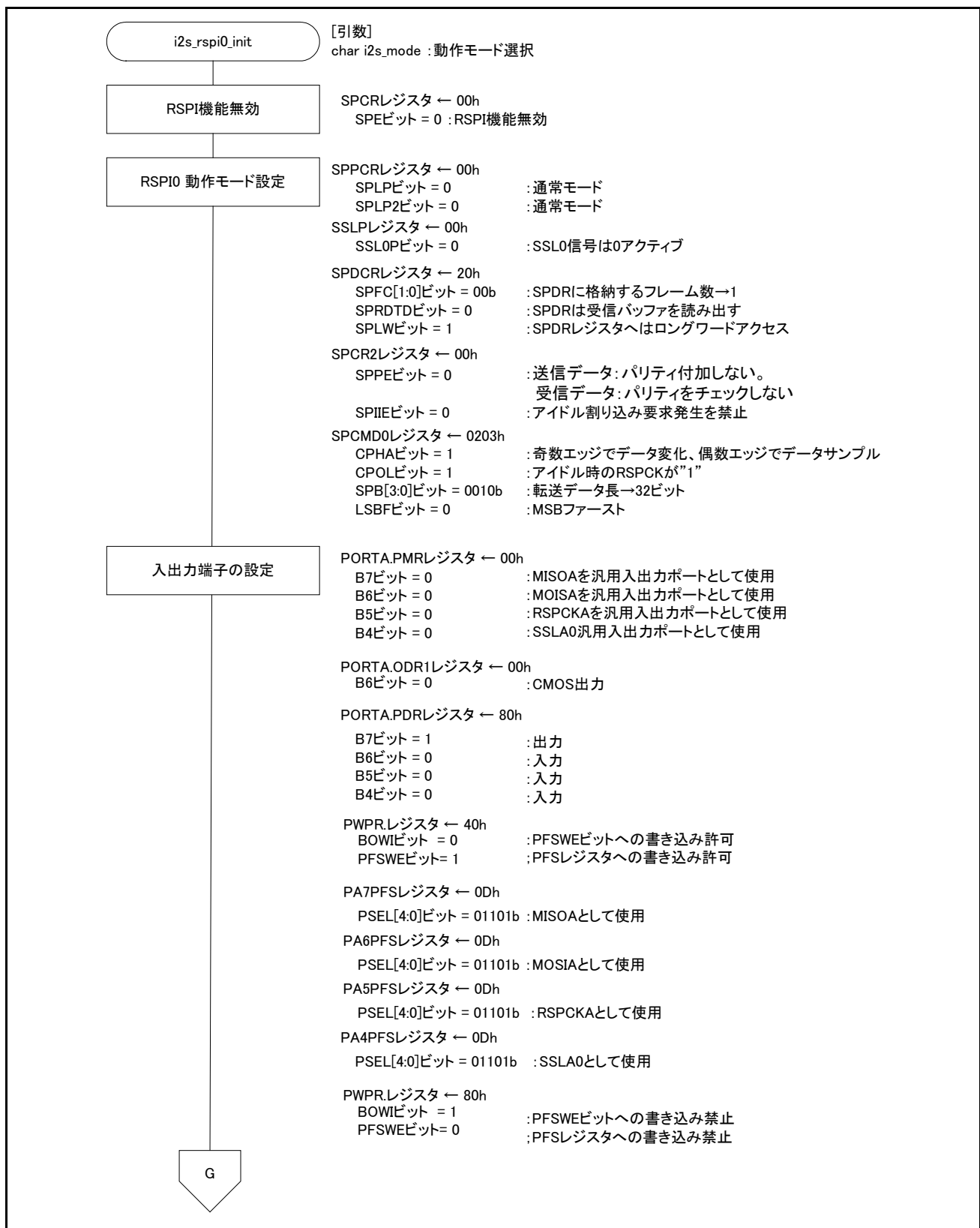


図 6.20 RSPI0 初期設定(L-ch 送受信)(1/2)

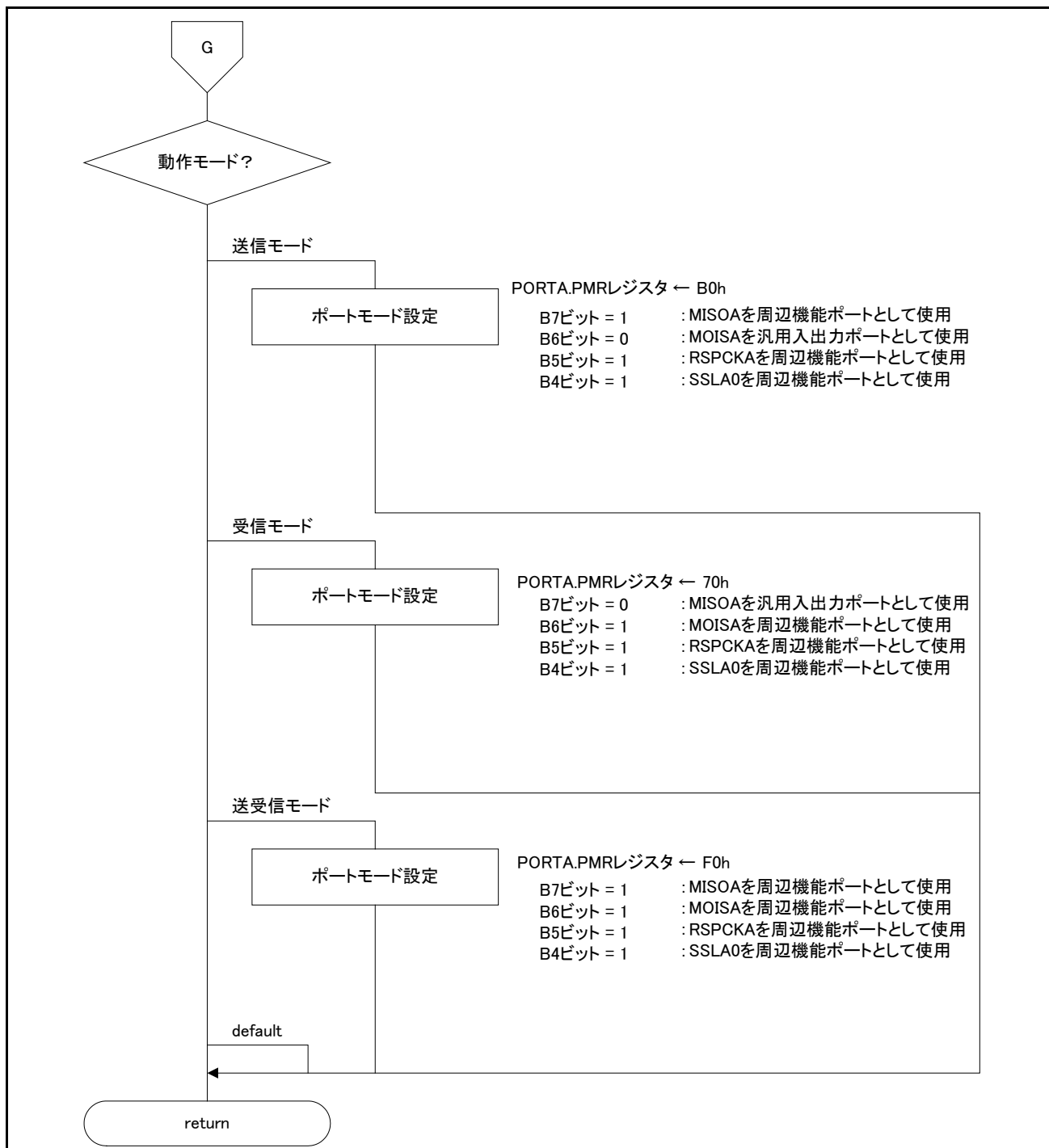


図 6.21 RSPI0 初期設定(L-ch 送受信)(2/2)

6.2.14 i2s\_rspl1\_init 関数

図 6.22、図 6.23 に i2s\_rspl1\_init 関数のフローチャートを示します。  
i2s\_rspl1\_init 関数は RSPI1 の初期設定を行います。

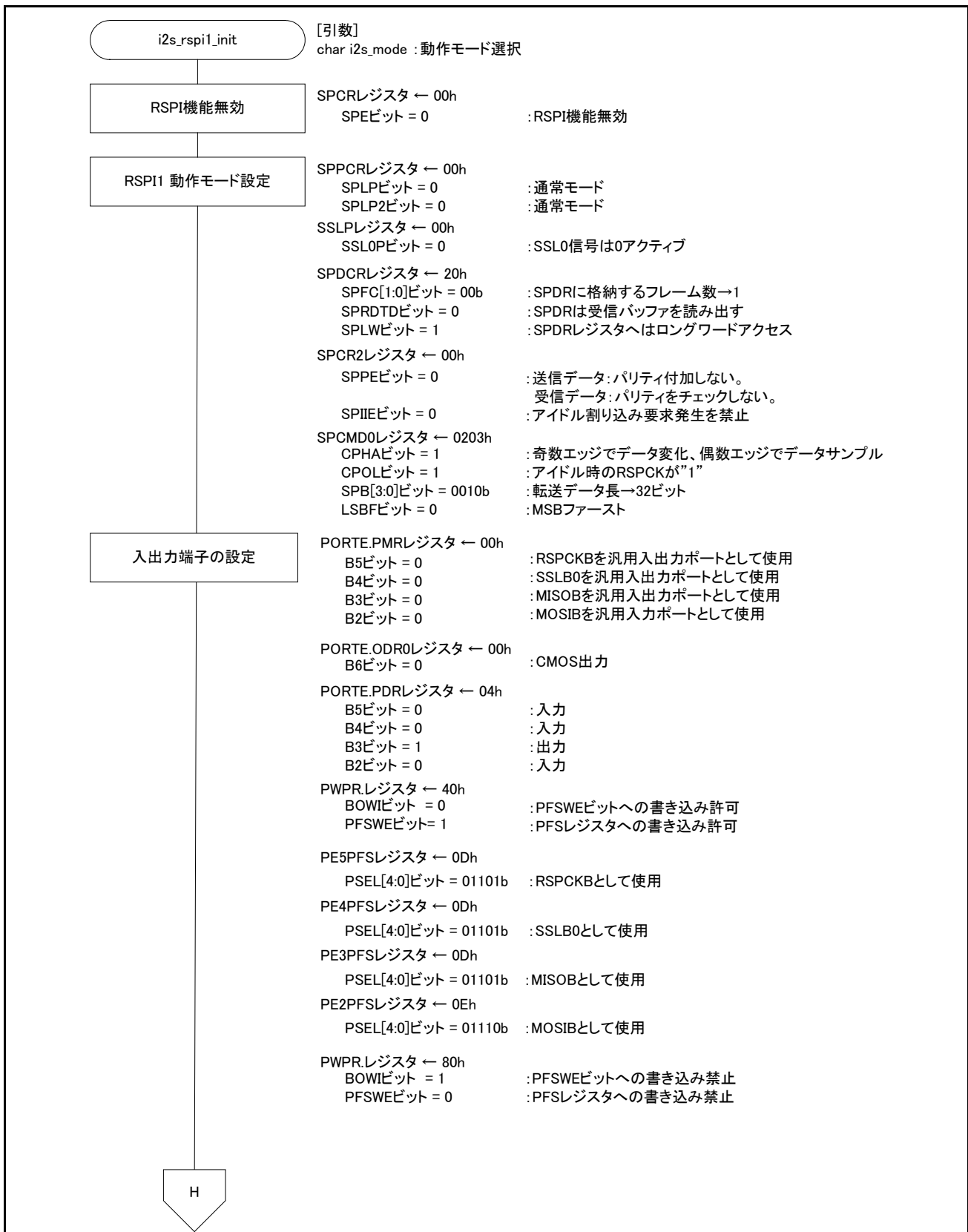


図 6.22 RSPI1 初期設定(R-ch 送受信)(1/2)

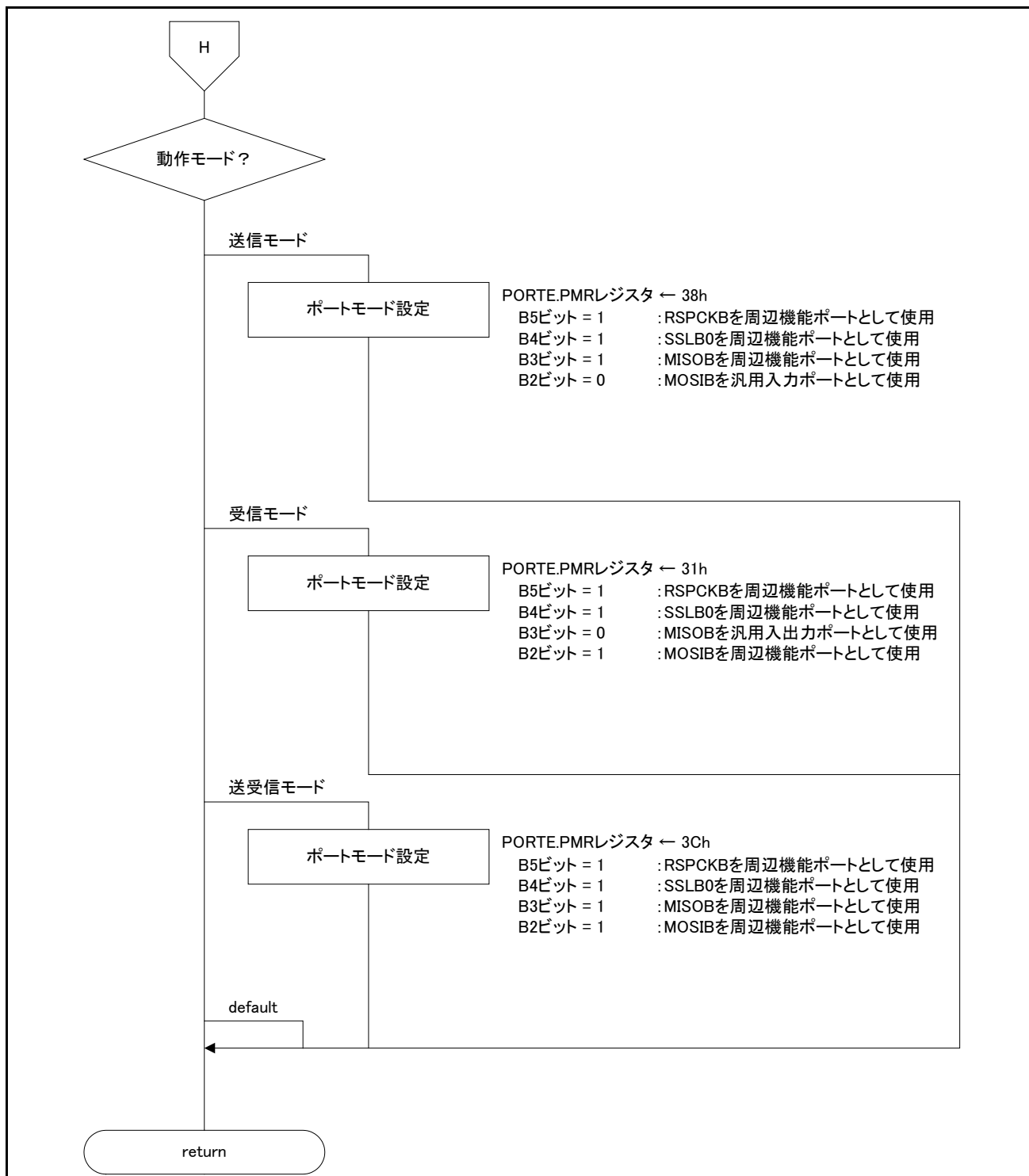


図 6.23 RSPI1 初期設定(R-ch 送受信)(2/2)

7. 付録

本サンプルコードに Renesas Starter Kit for RX63N を用いる場合、以下 2 箇所のボード上の変更を行ってください。図 7.1、図 7.2 に変更点を示します。

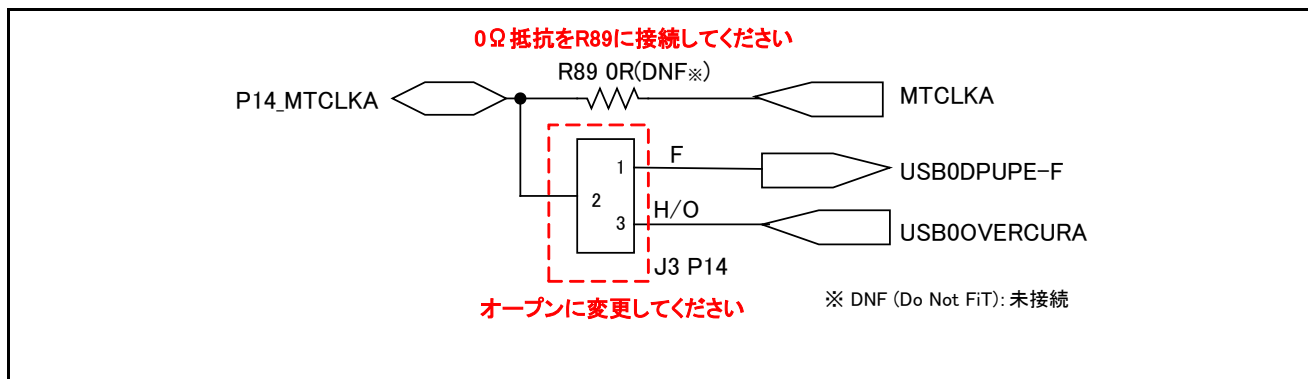


図 7.1 P14 を MTCLKA で使用するためのボードの変更点

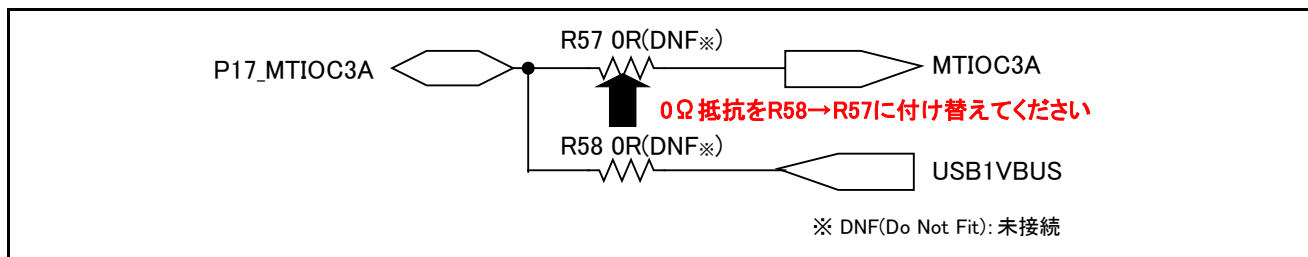


図 7.2 P17 を MTIOC3A で使用するためのボードの変更点

## 8. 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。

## 9. 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 ユーザーズマニュアル ハードウェア編 Rev.1.50 (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 | RX63N グループ、RX631 グループ アプリケーションノート<br>RSPI、DTCa および MTU2a を用いた I2S 通信 |
|------|--|

| Rev. | 発行日        | 改訂内容 |      |
|------|------------|------|------|
|      |            | ページ  | ポイント |
| 1.00 | 2013.04.05 | —    | 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（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）がありません。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. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。  
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、  
家電、工作機械、パーソナル機器、産業用ロボット等  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、  
防災・防犯装置、各種安全装置等  
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っていません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事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・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

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

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

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