

要旨

本アプリケーションノートでは、RZ/T1 に搭載されている FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF) の調歩同期式通信を使用したサンプルプログラムについて説明します。

サンプルプログラムの特長を以下に示します。

- PC と接続し、ターミナルソフトを用いることで、PC との通信が可能になります。

対象デバイス

RZ/T1 グループ

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	4
2.	動作環境	5
3.	関連アプリケーションノート	6
4.	周辺機能説明	7
5.	ハードウェア説明	8
5.1	ハードウェア構成例	8
5.2	使用端子一覧	8
6.	ソフトウェア説明	9
6.1	動作概要	9
6.1.1	プロジェクト設定	10
6.1.2	使用準備	10
6.2	メモリマップ	12
6.2.1	サンプルプログラムのセクション配置	12
6.2.2	MPU の設定	12
6.2.3	例外処理ベクタテーブル	12
6.3	固定幅整数一覧	12
6.4	定数／エラーコード一覧	13
6.5	大域変数一覧	13
6.6	関数一覧	14
6.7	関数仕様	15
6.7.1	main	15
6.7.2	lolnitScifa2	15
6.7.3	R_SCIFA_UART_Init	16
6.7.4	R_SCIFA_UART_Open	17
6.7.5	R_SCIFA_UART_Send	17
6.7.6	R_SCIFA_UART_Receive	18
6.7.7	userdef_scifa2_uart_init	18
6.7.8	userdef_scifa2_uart_open	19
6.7.9	userdef_scifa2_uart_send	19
6.7.10	userdef_scifa2_uart_receive	19
6.8	フローチャート	20
6.8.1	メイン処理	20
6.8.2	SCIFA チャンネル 2 初期設定処理	20
6.8.3	SCIFA 初期設定処理	21
6.8.4	SCIFA 起動処理	22
6.8.5	データ送信処理	23
6.8.6	データ受信処理	23
6.8.7	SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義)	24
6.8.8	SCIFA チャンネル 2 UART モード起動処理 (ユーザ定義)	26
6.8.9	SCIFA チャンネル 2 データ送信処理 (ユーザ定義)	26

6.8.10	SCIFA チャンネル 2 データ受信処理 (ユーザ定義)	27
7.	サンプルプログラム.....	28
8.	参考ドキュメント	29

1. 仕様

表 1.1 に使用する周辺機能と用途を、図 1.1 に動作環境を示します。

表 1.1 使用する周辺機能と用途

周辺機能	用途
クロック発生回路 (CPG)	CPUクロックおよび低速オンチップオシレータで使用
FIFO内蔵シリアルコミュニケーションインタフェース (SCIFA)	SCIFAの歩調同期式を使用し、RS-232CインタフェースによるCOMポート通信に使用
エラーコントロールモジュール (ECM)	ERROROUT#端子の初期化

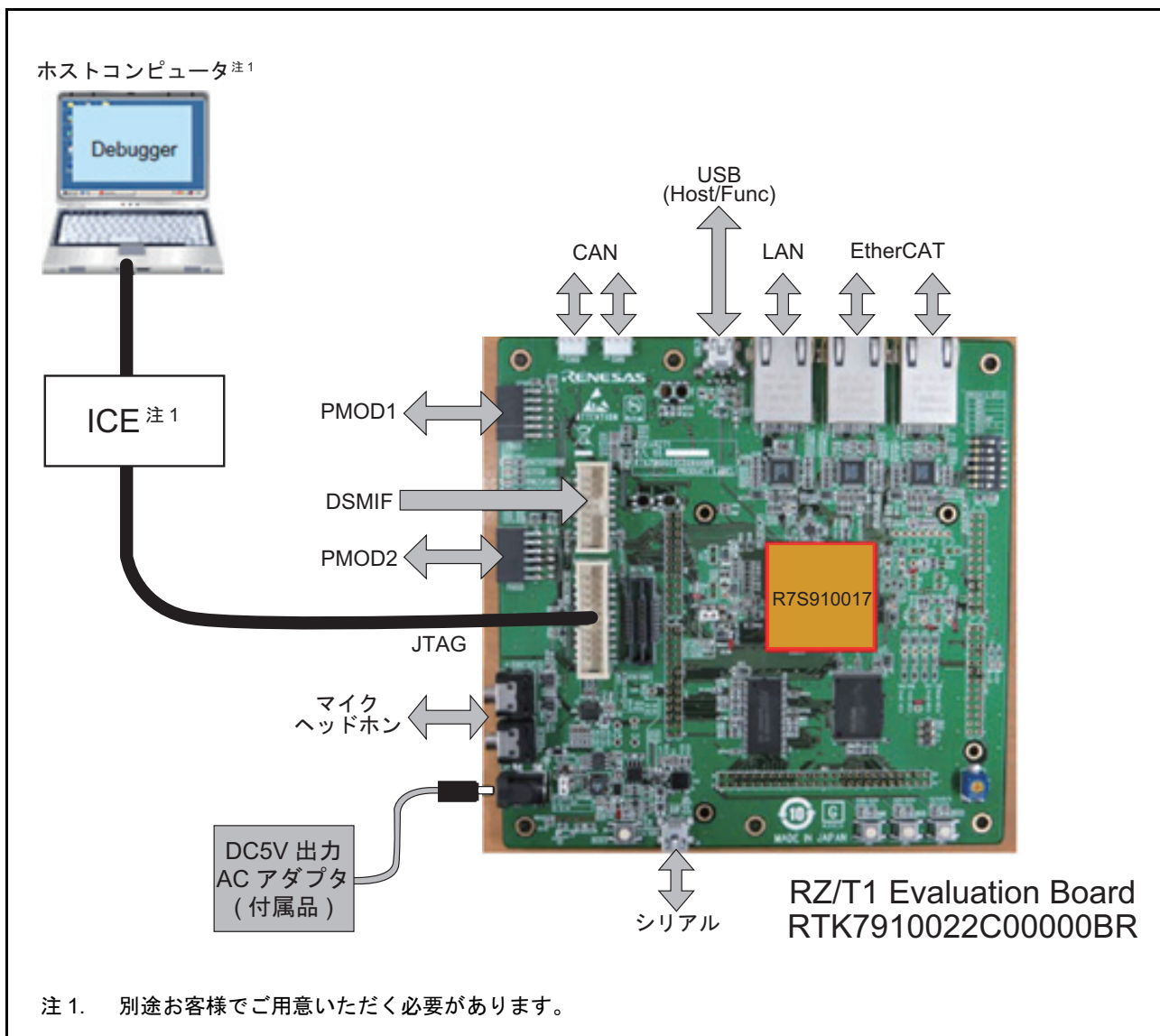


図 1.1 動作環境

2. 動作環境

本アプリケーションのサンプルコードは、下記の環境を想定しています。

表2.1 動作環境

項目	内容
使用マイコン	RZ/T1グループ
動作周波数	CPUCLK = 450MHz
動作電圧	3.3V
統合開発環境	IARシステムズ製 Embedded Workbench® for Arm Version 8.20.2 Arm製 DS-5™ 5.26.2 RENESAS製 e2studio 6.1.0
動作モード	SPIブートモード 16ビットバスブートモード
ターミナルソフトの通信設定	<ul style="list-style-type: none">通信速度：115200bpsデータ長：8ビットパリティ：なしストップビット長：1ビットフロー制御：なし
使用ボード	RZ/T1 Evaluation Board (RTK7910022C00000BR)
使用デバイス (ボード上で使用する機能)	<ul style="list-style-type: none">シリアルインタフェース (USB-Mini Bコネクタ J8)NORフラッシュメモリ (CS0、CS1空間に接続) メーカー名：Macronix International Co...、型名：MX29GL512FLT2I-10QSDRAM (CS2、CS3空間に接続) メーカー名：Integrated Silicon Solution Inc、型名：IS42S16320D-7TLシリアルフラッシュメモリ メーカー名：Macronix International Co...、型名：MX25L51245G
PC用USBシリアルポートドライバ	<ul style="list-style-type: none">RTK7910018C00000BE向けRTK7910022C00000BR向け

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。

- RZ/T1 グループ初期設定

4. 周辺機能説明

クロック発生回路 (CPG)、FIFO 内蔵シリアルコミュニケーションインタフェース (SCIFA)、エラーコントロールモジュール (ECM) についての基本的な内容は、RZ/T1 グループ・ユーザズマニュアルハードウェア編を参照してください。

5. ハードウェア説明

5.1 ハードウェア構成例

図 5.1 にハードウェア構成例を示します。

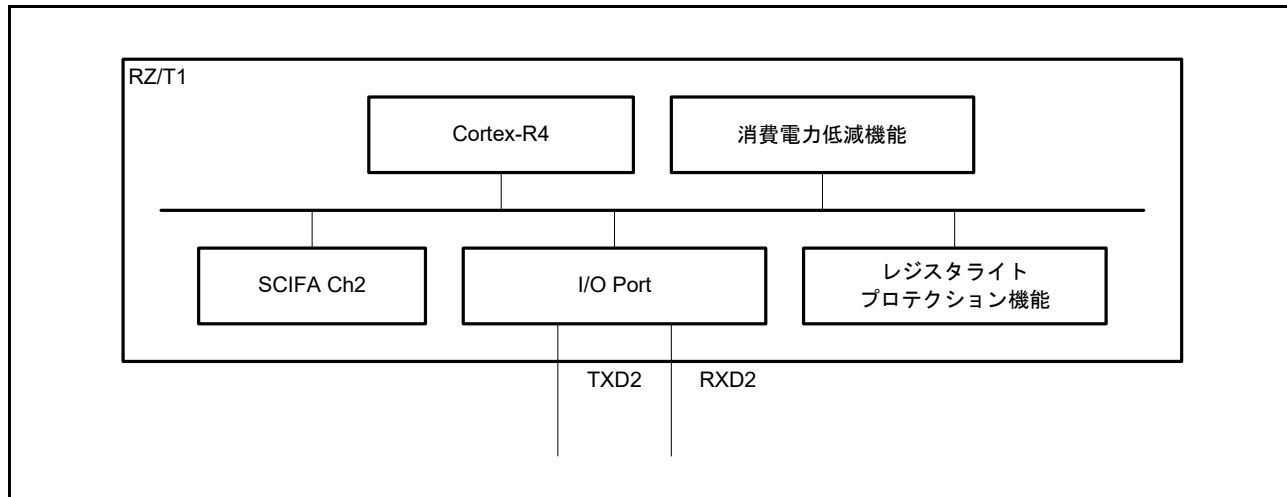


図 5.1 ハードウェア構成例

5.2 使用端子一覧

表 5.1 に使用端子と機能を示します。

表 5.1 使用端子と機能

端子名	入出力	内容
MD0	入力	動作モードの選択 MD0 = "L"、MD1 = "L"、MD2 = "L" (SPI ブートモード) MD0 = "L"、MD1 = "H"、MD2 = "L" (16ビットバスブートモード)
MD1	入力	
MD2	入力	
TXD2	出力	シリアル送信データ信号
RXD2	入力	シリアル受信データ信号

6. ソフトウェア説明

6.1 動作概要

本サンプルプログラムでは、FIFO 内蔵シリアルコミュニケーションインタフェース (SCIFA) の調歩同期式通信を用い、ホスト PC と RS-232 インタフェースの COM ポート通信を行うための設定を行い、通信可能状態とします。設定完了後は、ホスト PC からの入力キーに応じて、応答します。

本サンプルプログラムの機能概要を示します。

表6.1 動作概要

機能	概要
チャンネル	チャンネル2 (SCIFA2)
シリアル通信方式	調歩同期式
クロック	SERICKL = 150MHz
送信/受信	シリアルデータ送信/受信 LSB ファースト
転送速度	115200bps
キャラクタ長	8ビット
ストップビット長	1ビット
パリティ機能	なし
ハードウェアフロー制御	なし
割り込み	BRIF2 禁止 RXIF2 禁止 TXIF2 禁止 DRIF2 禁止

6.1.1 プロジェクト設定

開発環境となる EWARM 上で使用されるプロジェクト設定については、アプリケーションノート「RZ/T1グループ初期設定」に記載しています。

6.1.2 使用準備

本サンプルプログラムでは、PC との通信動作を行いますので、その実行準備を説明します。

(1) ホスト PC にてターミナルソフトを起動し、次のように設定します。(Tera Term で COM4 使用の場合)

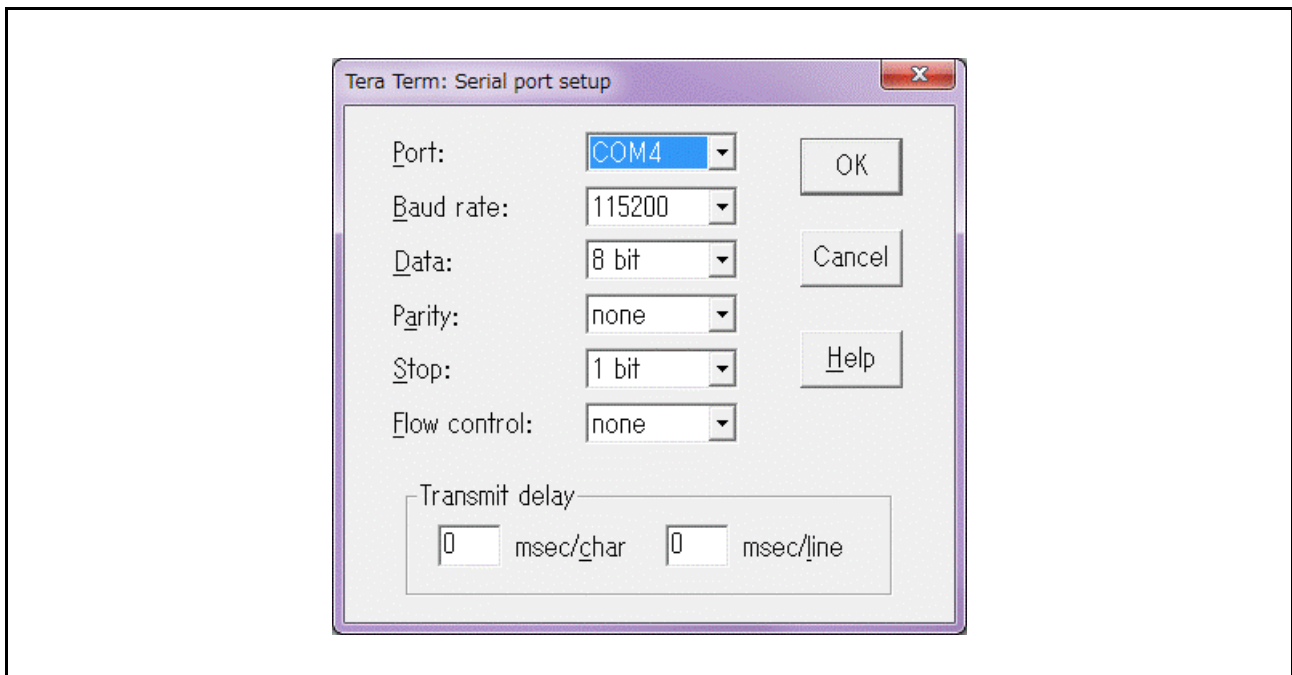


図 6.1 シリアルポートの設定

(2) サンプルプログラムを実行し通信可能状態になると、下図に示すようにサンプルプログラムからの受信データがターミナルソフトに表示されます。

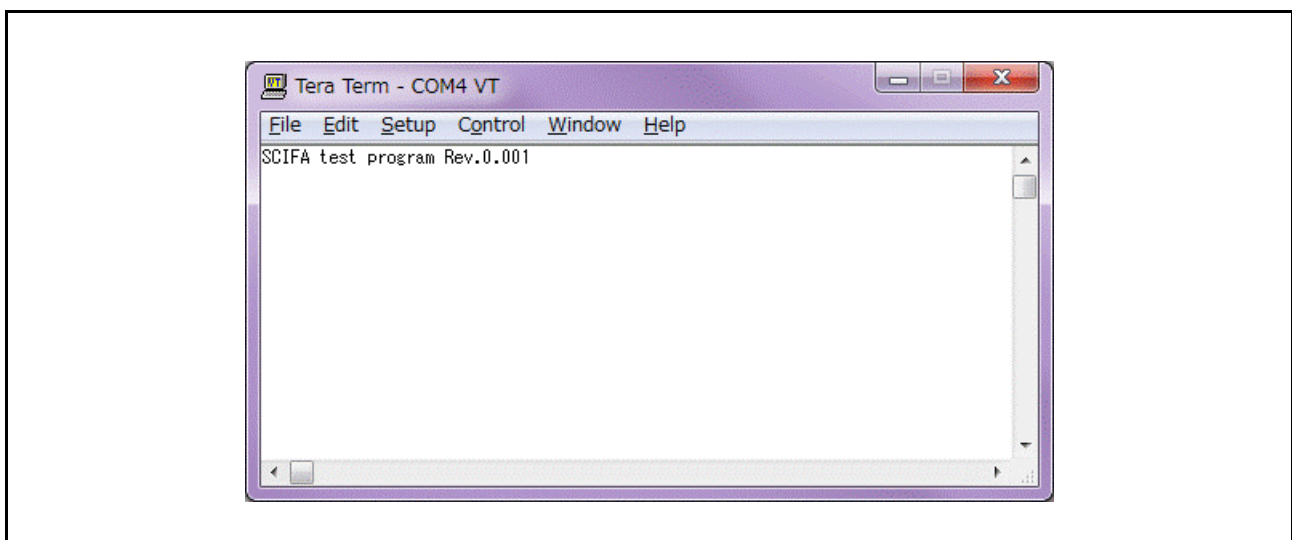


図 6.2 SCIFA 設定後のターミナルソフト上の表示

- (3) ホスト PC からテンキーや、英数字などを入力し、Enter キーを入力すると、下図に示すようにターミナルソフト上で、サンプルプログラムからの応答 (input = 1234567890) を確認することができます。

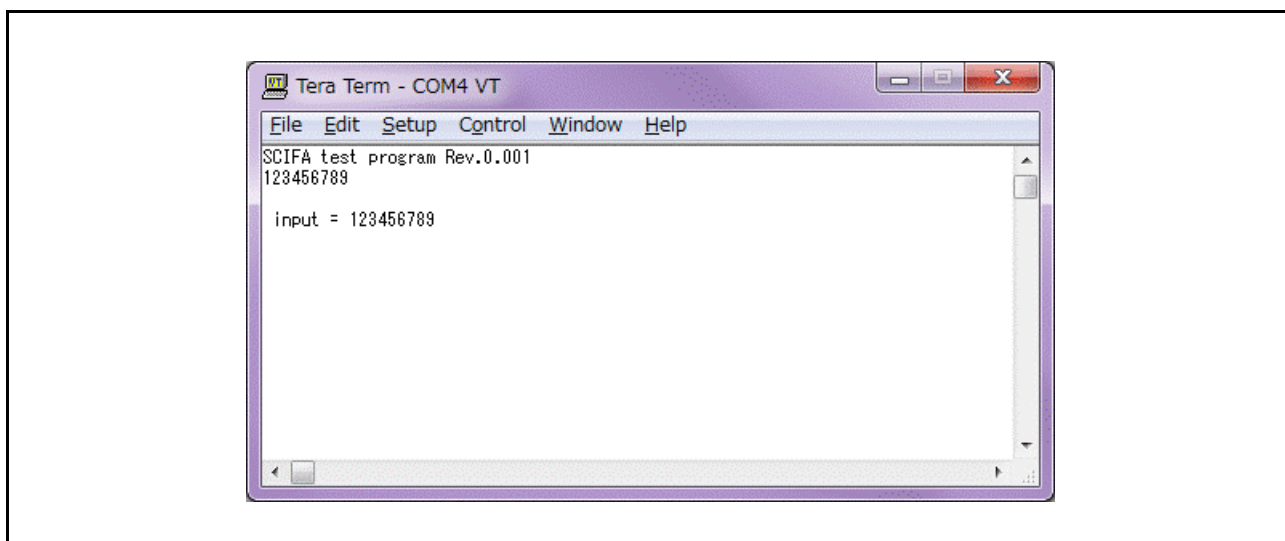


図 6.3 キーボード入力後のターミナルソフト上の表示

6.2 メモリマップ

RZ/T1 グループのアドレス空間と RZ/T1 評価ボードのメモリマッピングについては、アプリケーションノート「RZ/T1 グループ初期設定」に記載しています。

6.2.1 サンプルプログラムのセクション配置

サンプルプログラムで使用するセクションおよびサンプルプログラムの初期状態のセクション配置（ロードビュー）、スキップロード機能を使用した後のセクション配置（実行ビュー）は、アプリケーションノート「RZ/T1 グループ初期設定」に記載しています。

6.2.2 MPU の設定

MPU の設定は、アプリケーションノート「RZ/T1 グループ初期設定」に記載しています。

6.2.3 例外処理ベクタテーブル

例外処理のベクタテーブルは、アプリケーションノート「RZ/T1 グループ初期設定」に記載しています。

6.3 固定幅整数一覧

表 6.2 にサンプルプログラムで使用する固定幅整数を示します。

表6.2 サンプルプログラムで使用する固定幅整数

シンボル	内容
int8_t	8ビット整数、符号あり（標準ライブラリにて定義）
int16_t	16ビット整数、符号あり（標準ライブラリにて定義）
int32_t	32ビット整数、符号あり（標準ライブラリにて定義）
int64_t	64ビット整数、符号あり（標準ライブラリにて定義）
uint8_t	8ビット整数、符号なし（標準ライブラリにて定義）
uint16_t	16ビット整数、符号なし（標準ライブラリにて定義）
uint32_t	32ビット整数、符号なし（標準ライブラリにて定義）
uint64_t	64ビット整数、符号なし（標準ライブラリにて定義）

6.4 定数/エラーコード一覧

表 6.3 にサンプルプログラムで使用する定数を、表 6.4 にサンプルプログラムのエラーコードを示します。

表 6.3 サンプルプログラムで使用する定数

定数名	設定値	内容
SCIFA_UART_CH_TOTAL	(5)	SCIFAのチャンネル数
SCIFA_UART_CH_0	(0)	SCIFAのチャンネル0を指定するための定数です。
SCIFA_UART_CH_1	(1)	SCIFAのチャンネル1を指定するための定数です。
SCIFA_UART_CH_2	(2)	SCIFAのチャンネル2を指定するための定数です。
SCIFA_UART_CH_3	(3)	SCIFAのチャンネル3を指定するための定数です。
SCIFA_UART_CH_4	(4)	SCIFAのチャンネル4を指定するための定数です。
SCIFA_UART_MODE_R	(1)	SCIFAを受信モードで使用するための定数です。 SCIFAチャンネル初期化関数およびSCIFAチャンネルオープン関数の引数に使用します。
SCIFA_UART_MODE_W	(2)	SCIFAを送信モードで使用するための定数です。 SCIFAチャンネル初期化関数およびSCIFAチャンネルオープン関数の引数に使用します。
SCIFA_UART_MODE_RW	(3)	SCIFAを送受信モードで使用するための定数です。 SCIFAチャンネル初期化関数およびSCIFAチャンネルオープン関数の引数に使用します。
SCIFA_UART_CKS_DIVISION_1	(0)	SCIFAのボーレートジェネレータのクロックソースをSERICLKクロックに設定するための定数です。SCIFAチャンネル初期化関数の引数に使用します。
SCIFA_UART_CKS_DIVISION_4	(1)	SCIFAのボーレートジェネレータのクロックソースをSERICLK/4クロックに設定するための定数です。SCIFAチャンネル初期化関数の引数に使用します。
SCIFA_UART_CKS_DIVISION_16	(2)	SCIFAのボーレートジェネレータのクロックソースをSERICLK/16クロックに設定するための定数です。SCIFAチャンネル初期化関数の引数に使用します。
SCIFA_UART_CKS_DIVISION_64	(3)	SCIFAのボーレートジェネレータのクロックソースをSERICLK/64クロックに設定するための定数です。SCIFAチャンネル初期化関数の引数に使用します。

表 6.4 サンプルプログラムのエラーコード

定数名	設定値	内容
SCIFA_UART_SUCCESS	(0)	SCIFA関連のAPI関数の戻り値のための定数です。 関数実行が成功したことを意味します。
SCIFA_UART_ERR	(-1)	SCIFA関連のAPI関数の戻り値のための定数です。 関数実行が失敗したことを意味します。
SCIFA_UART_ERR_RECEIVE	(-2)	SCIFA関連のAPI関数の戻り値のための定数です。 データ受信関数実行が失敗したことを意味します。

6.5 大域変数一覧

表 6.5 に大域変数一覧を示します。

表 6.5 大域変数一覧

型	変数名	内容	使用関数
char	gbuff[16]	ターミナルソフトからの受信データ	main

6.6 関数一覧

表 6.6 に関数一覧を示します。

表 6.6 関数一覧

関数名	ページ番号
main	15
IoInitScifa2	15
R_SCIFA_UART_Init	16
R_SCIFA_UART_Open	17
R_SCIFA_UART_Send	17
R_SCIFA_UART_Receive	18
userdef_scifa2_uart_init	18
userdef_scifa2_uart_open	19
userdef_scifa2_uart_send	19
userdef_scifa2_uart_receive	19

6.7 関数仕様

6.7.1 main

main	
概要	メイン処理
宣言	int main (void)
説明	ECM 機能、SCIFA を初期化し、SCIFA テストプログラム開始を表すデータを送信します。 その後、PC のキーボード入力によるデータを受信するまでループし、データ受信があったとき、受信したデータを送信します。
引数	なし
リターン値	なし
補足	なし

6.7.2 lolnitScifa2

lolnitScifa2	
概要	SCIFA チャンネル 2 初期設定処理
宣言	void lolnitScifa2(void)
説明	UART モードとして SCIFA のチャンネル 2 を初期化した後、送受信を有効にし、通信を開始します。
引数	なし
リターン値	なし
補足	なし

6.7.3 R_SCIFA_UART_Init

R_SCIFA_UART_Init

概要	SCIFA 初期設定処理
ヘッダ	r_scifa_uart.h
宣言	uint32_t R_SCIFA_UART_Init(uint32_t channel, uint32_t mode, uint16_t cks, uint8_t brr)
説明	SCIFA を調歩同期式通信モードに初期化し、SCIFA で使用するポート設定を行います。
引数	uint32_t channel : 初期化する SCIFA チャンネルを指定します。 設定可能範囲 (0 ~ 4)
	uint32_t mode : SCIFA の動作モードを指定します。 SCIFA_UART_MODE_R : 受信モード SCIFA_UART_MODE_W : 送信モード SCIFA_UART_MODE_RW : 送受信モード
	uint16_t cks : SCIFA のボーレートジェネレータのクロックソースを選択します。 SCIFA_UART_CKS_DIVISION_1 : SERICLK SCIFA_UART_CKS_DIVISION_4 : SERICLK / 4 SCIFA_UART_CKS_DIVISION_16 : SERICLK / 16 SCIFA_UART_CKS_DIVISION_64 : SERICLK / 64
	uint8_t brr : SCIFA ビットレートレジスタ (BRR) に設定する値を指定します。 設定可能範囲 (ハードウェアマニュアルを参照)
リターン値	SCIFA_UART_SUCCESS : 初期化成功 SCIFA_UART_ERR : 引数エラー
補足	<ul style="list-style-type: none"> SCIFA チャンネル 2 のみ実装しています。チャンネル 1、3 ~ 4 は未実装です。 SCIFA チャンネル 2 で使用する端子は以下の通りです。 TXD2 : P91 RXD2 : P92

6.7.4 R_SCIFA_UART_Open

R_SCIFA_UART_Open

概要	SCIFA 起動処理
ヘッダ	r_scifa_uart.h
宣言	int32_t R_SCIFA_UART_Open(uint32_t channel, uint32_t mode)
説明	引数にて指定された動作モード（送信／受信／送受信）で SCIFA を起動し、調歩同期式通信を開始します。
引数	uint32_t channel : 初期化する SCIFA チャンネルを指定します。 設定可能範囲 (0 ~ 4) uint32_t mode : SCIFA の動作モードを指定します。 SCIFA_UART_MODE_R : 受信モード SCIFA_UART_MODE_W : 送信モード SCIFA_UART_MODE_RW : 送受信モード
リターン値	SCIFA_UART_SUCCESS : 起動成功 SCIFA_UART_ERR : 引数エラー
補足	<ul style="list-style-type: none"> SCIFA チャンネル 2 のみ実装しています。チャンネル 1、3 ~ 4 は未実装です。

6.7.5 R_SCIFA_UART_Send

R_SCIFA_UART_Send

概要	データ送信処理
ヘッダ	r_scifa_uart.h
宣言	int32_t R_SCIFA_UART_Send(uint32_t channel, uint8_t data)
説明	RS-232C インタフェースによる COM ポート通信で、1 バイトのデータを送信します。
引数	uint32_t channel : 初期化する SCIFA チャンネルを指定します。 設定可能範囲 (0 ~ 4) uint8_t data : 送信データを指定します。
リターン値	SCIFA_UART_SUCCESS : 送信成功 SCIFA_UART_ERR : 引数エラー
補足	<ul style="list-style-type: none"> SCIFA チャンネル 2 のみ実装しています。チャンネル 1、3 ~ 4 は未実装です。

6.7.6 R_SCIFA_UART_Receive

R_SCIFA_UART_Receive

概要	データ受信処理
ヘッダ	r_scifa_uart.h
宣言	int32_t R_SCIFA_UART_Receive(uint32_t channel, uint8_t *data)
説明	RS-232C インタフェースによる COM ポート通信で、1 バイトのデータを受信します。
引数	uint32_t channel : 初期化する SCIFA チャンネルを指定します。 設定可能範囲 (0 ~ 4)
	uint8_t *data : 受信したデータを格納する領域を指定します。
リターン値	SCIFA_UART_SUCCESS : 受信成功 SCIFA_UART_ERR : 引数エラー SCIFA_UART_ERR_RECEIVE : 受信エラー
補足	<ul style="list-style-type: none"> SCIFA チャンネル 2 のみ実装しています。チャンネル 1、3 ~ 4 は未実装です。

6.7.7 userdef_scifa2_uart_init

userdef_scifa2_uart_init

概要	SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義)
宣言	void userdef_scifa2_uart_init(uint32_t mode, uint16_t cks, uint8_t brr)
説明	
引数	uint32_t mode : SCIFA の動作モードを指定します。 SCIFA_UART_MODE_R : 受信モード SCIFA_UART_MODE_W : 送信モード SCIFA_UART_MODE_RW : 送受信モード
	uint16_t cks : SCIFA のボーレートジェネレータのクロックソースを選択します。 SCIFA_UART_CKS_DIVISION_1 : SERICLK SCIFA_UART_CKS_DIVISION_4 : SERICLK / 4 SCIFA_UART_CKS_DIVISION_16 : SERICLK / 16 SCIFA_UART_CKS_DIVISION_64 : SERICLK / 64
	uint8_t brr : SCIFA ビットレートレジスタ (BRR) に設定する値を指定します。 設定可能範囲 (ハードウェアマニュアルを参照)
リターン値	なし
補足	なし

6.7.8 userdef_scifa2_uart_open

userdef_scifa2_uart_open

概要	SCIFA チャンネル 2 UART モード起動処理 (ユーザ定義)
宣言	void userdef_scifa2_uart_open(uint32_t mode)
説明	引数 mode に応じて、送信動作、受信動作を許可に設定します。
引数	uint32_t mode : SCIFA の動作モードを指定します。 SCIFA_UART_MODE_R : 受信モード SCIFA_UART_MODE_W : 送信モード SCIFA_UART_MODE_RW : 送受信モード
リターン値	なし
補足	

6.7.9 userdef_scifa2_uart_send

userdef_scifa2_uart_send

概要	SCIFA チャンネル 2 データ送信処理 (ユーザ定義)
宣言	void userdef_scifa2_uart_send(uint8_t data)
説明	引数で指定したデータを送信します。
引数	uint8_t data : 送信データを指定します。
リターン値	なし
補足	なし

6.7.10 userdef_scifa2_uart_receive

userdef_scifa2_uart_receive

概要	SCIFA チャンネル 2 データ受信処理
宣言	int32_t userdef_scifa2_uart_receive(uint8_t *data)
説明	受信エラーのとき、受信エラー状態をクリアし、戻り値に受信エラーを設定します。 受信データを取得したとき、引数で指定した領域に受信データを格納します。
引数	uint8_t *data : 受信したデータを格納する領域を指定します。
リターン値	SCIFA_UART_SUCCESS : 受信成功 SCIFA_UART_ERR_RECEIVE : 受信エラー
補足	なし

6.8 フローチャート

6.8.1 メイン処理

図 6.4 にメイン処理のフローチャートを示します。

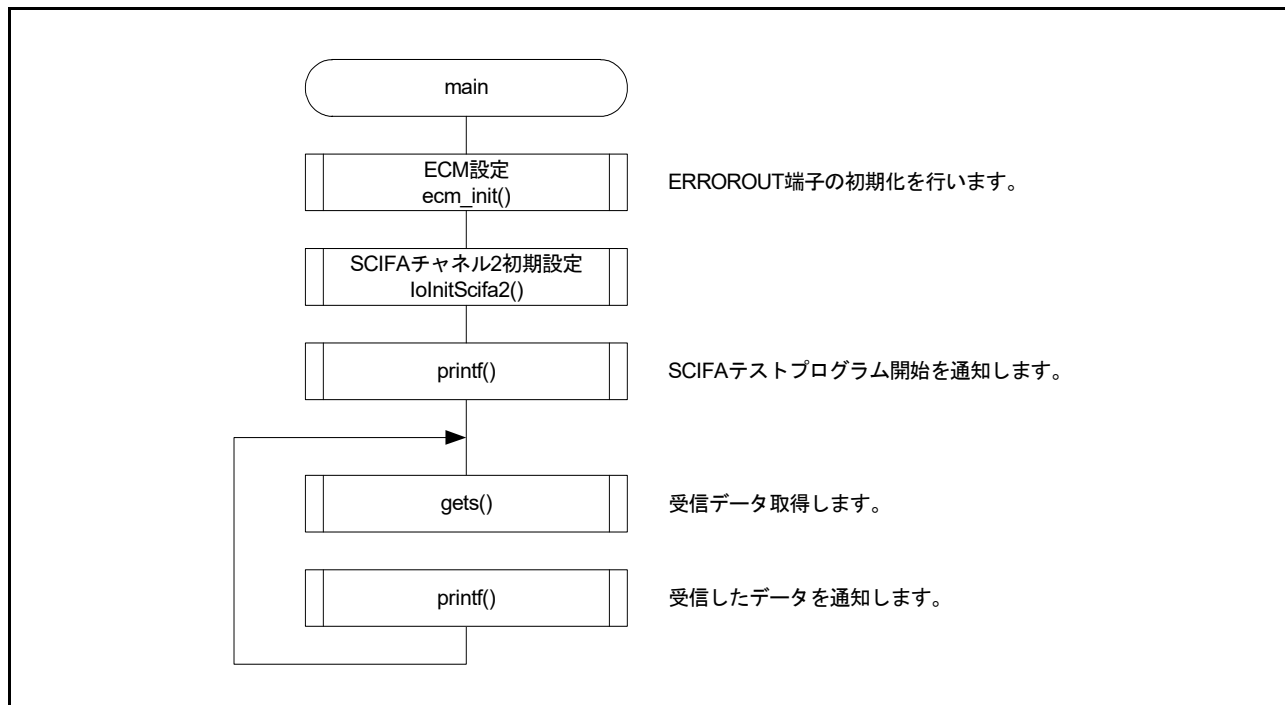


図 6.4 メイン処理

6.8.2 SCIFA チャンネル 2 初期設定処理

図 6.5 に SCIFA チャンネル 2 初期設定処理のフローチャートを示します。

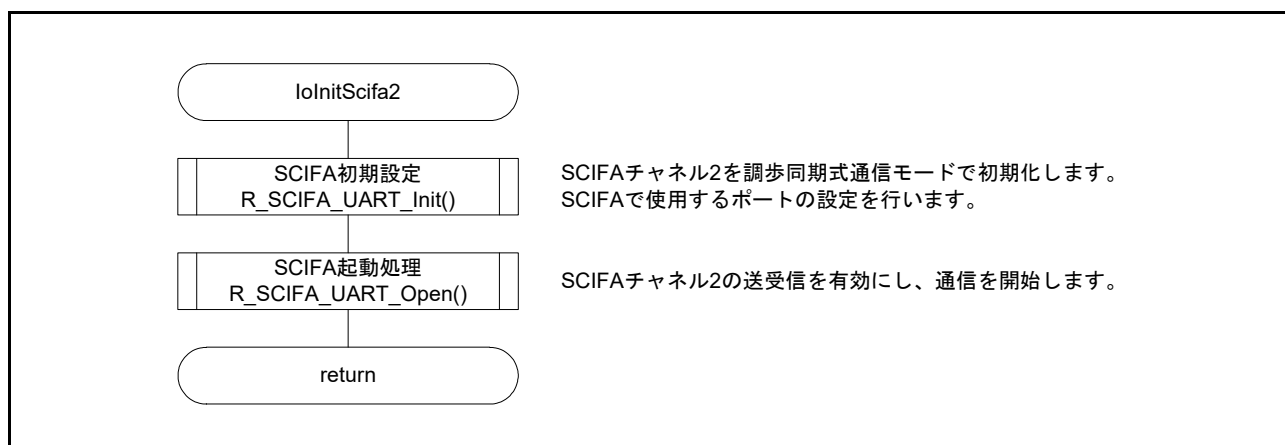


図 6.5 SCIFA チャンネル 2 初期設定処理

6.8.3 SCIFA 初期設定処理

図 6.6 に SCIFA 初期設定処理のフローチャートを示します。

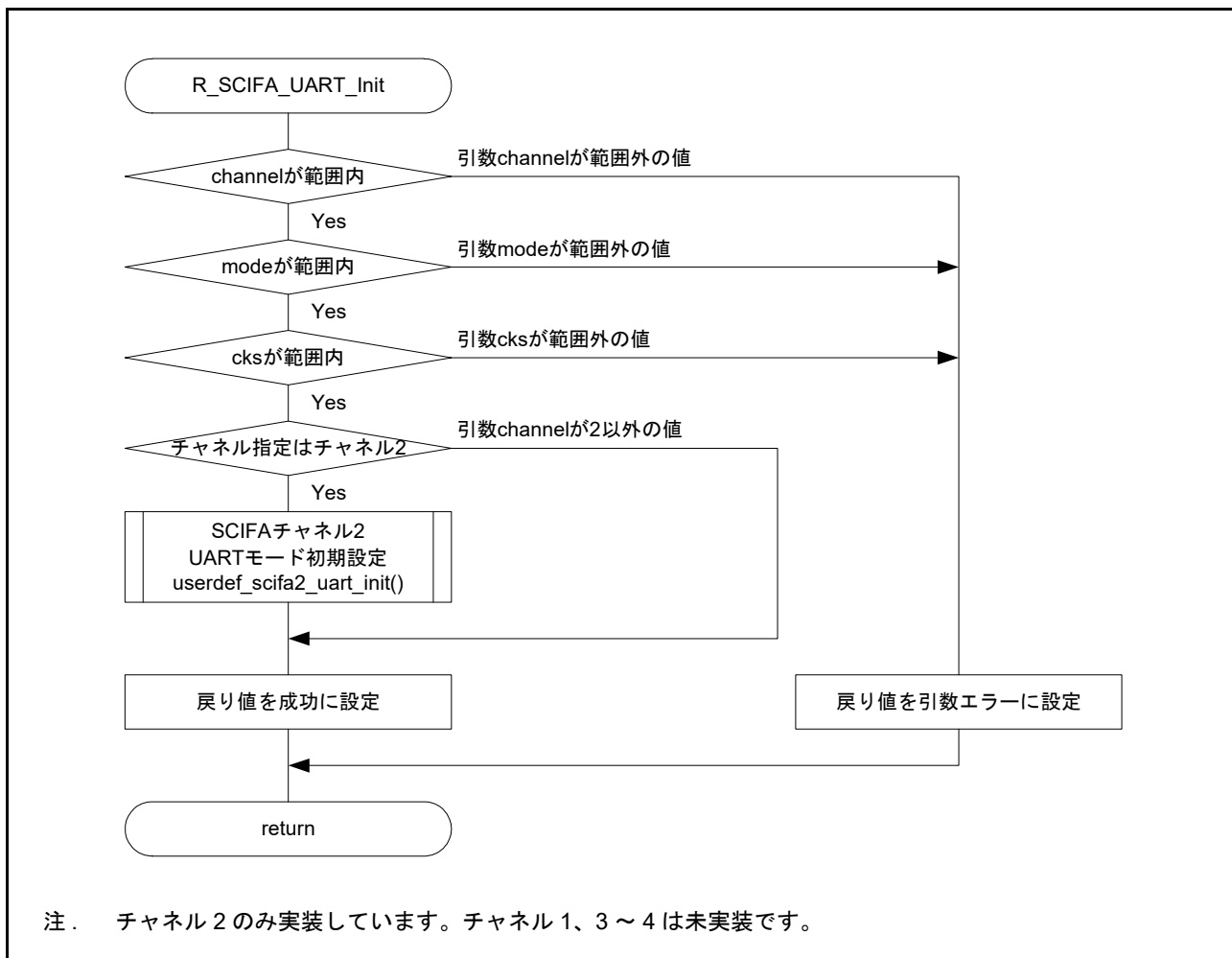


図 6.6 SCIFA 初期設定処理

6.8.4 SCIFA 起動処理

図 6.7 に SCIFA 起動処理のフローチャートを示します。

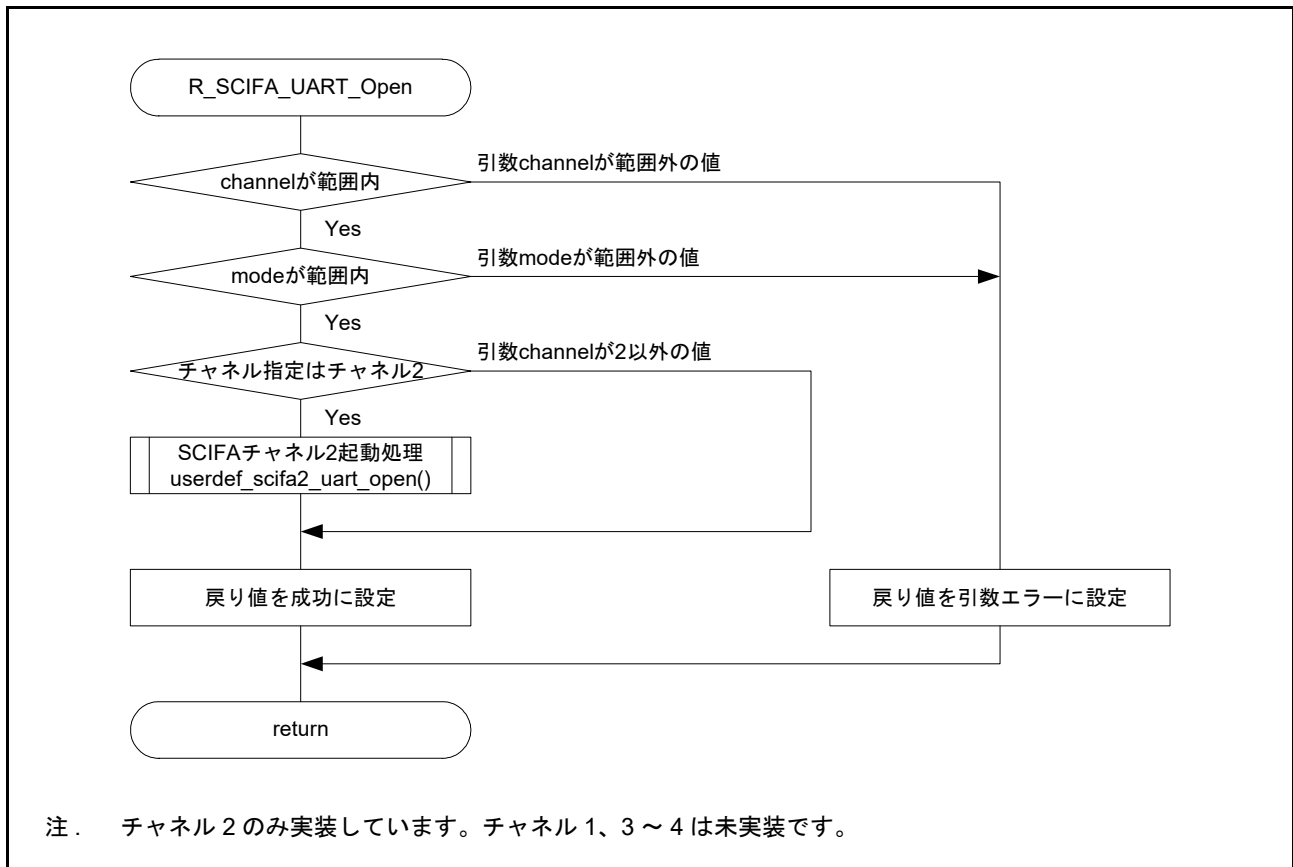


図 6.7 SCIFA 起動処理

6.8.5 データ送信処理

図 6.8 にデータ送信処理のフローチャートを示します。

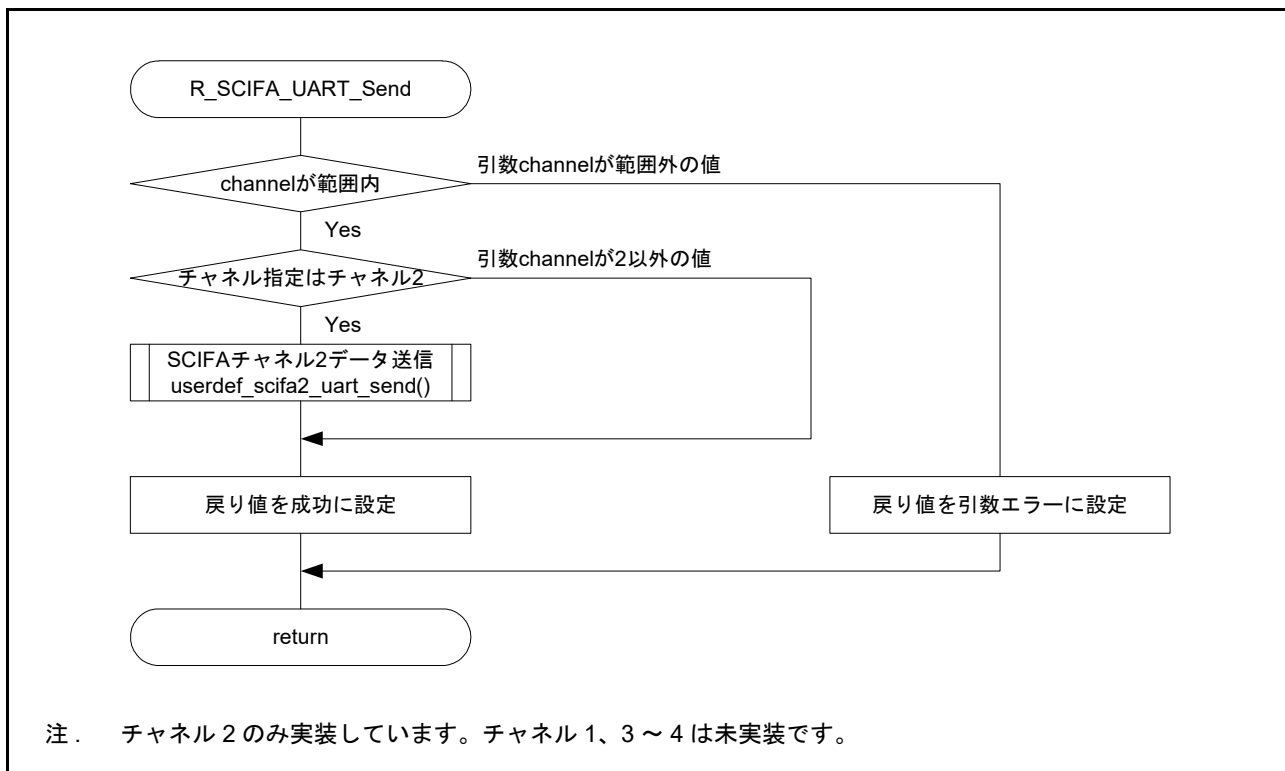


図 6.8 データ送信処理

6.8.6 データ受信処理

図 6.9 にデータ受信処理のフローチャートを示します。

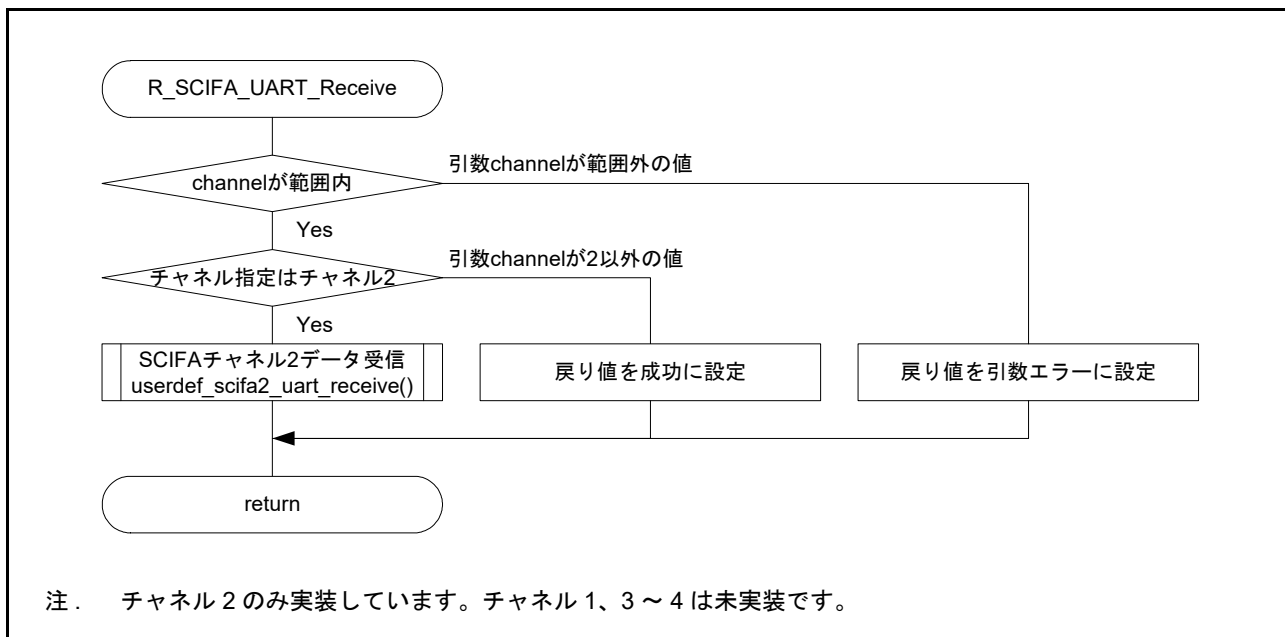


図 6.9 データ受信処理

6.8.7 SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義)

図 6.10、図 6.11 に SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義) のフローチャートを示します。

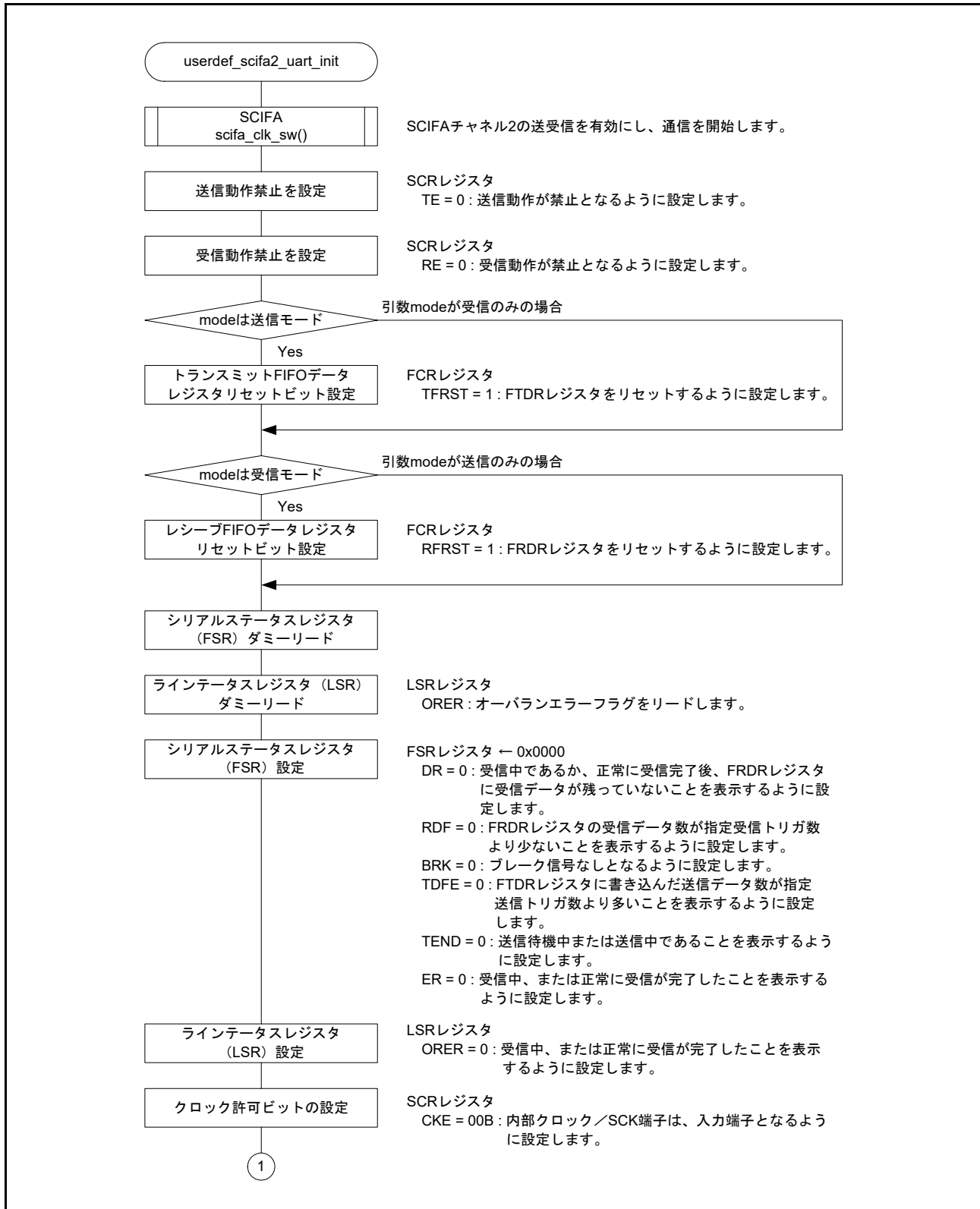


図 6.10 SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義) (1)

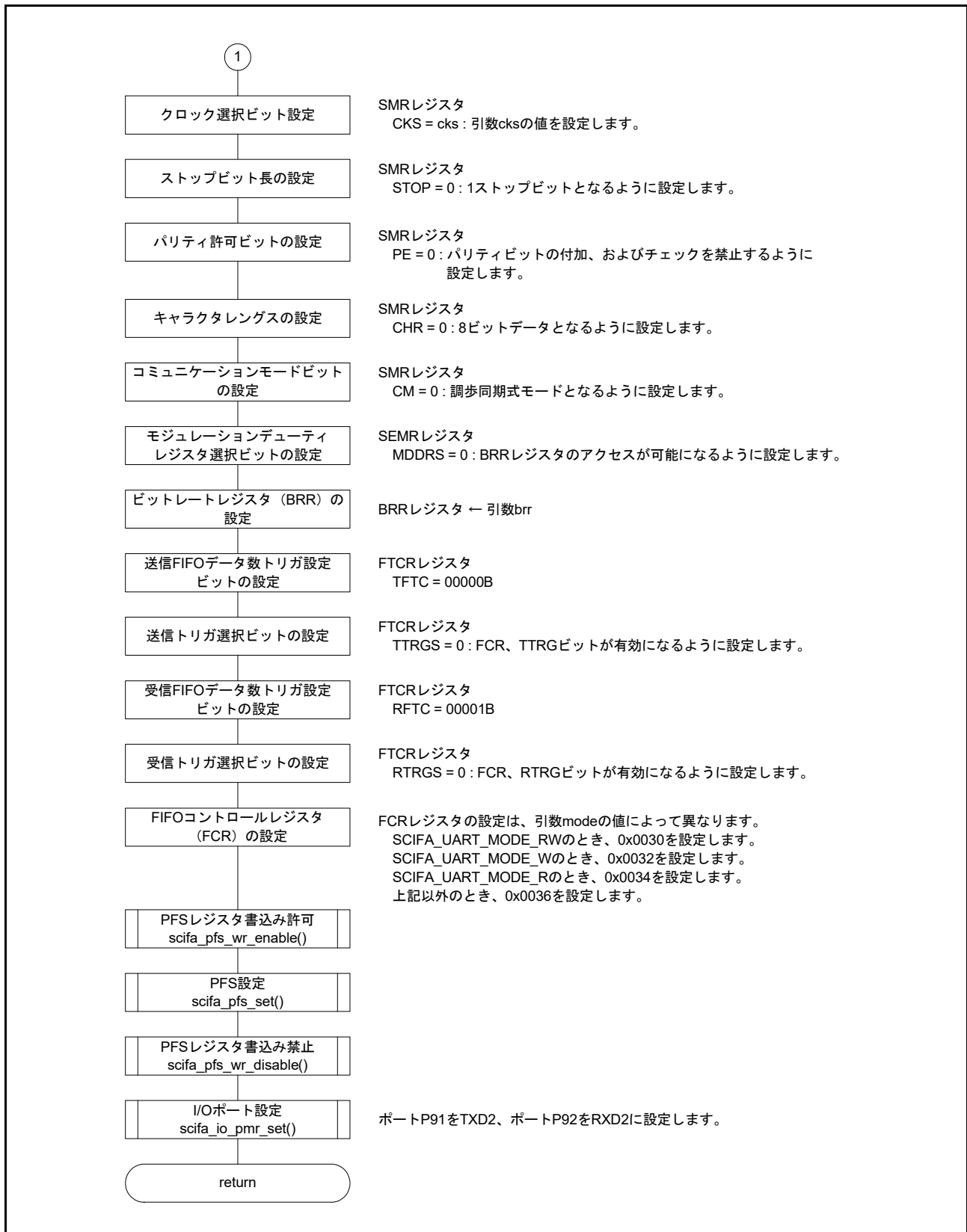


図 6.11 SCIFA チャンネル 2 UART モード初期設定処理 (ユーザ定義) (2)

6.8.8 SCIFA チャンネル 2 UART モード起動処理 (ユーザ定義)

図 6.12 に SCIFA チャンネル 2 UART モード起動処理 (ユーザ定義) のフローチャートを示します。

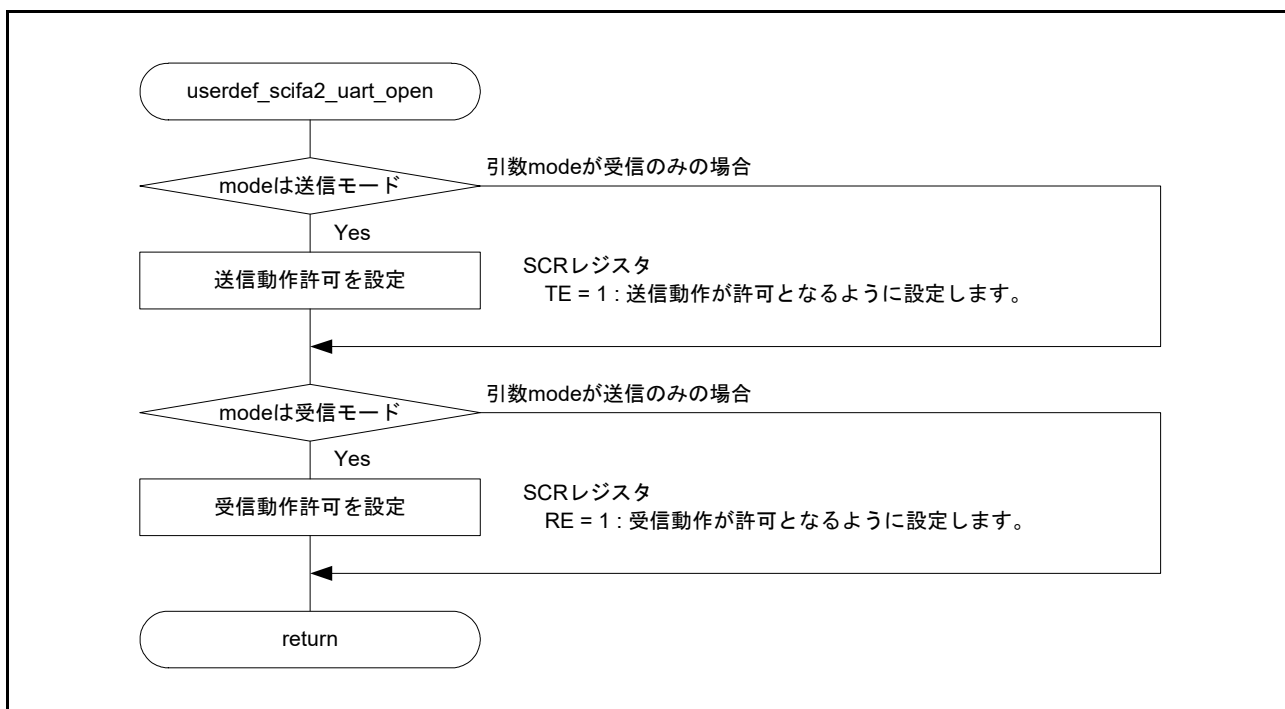


図 6.12 SCIFA チャンネル 2 UART モード起動処理 (ユーザ定義)

6.8.9 SCIFA チャンネル 2 データ送信処理 (ユーザ定義)

図 6.13 に SCIFA チャンネル 2 データ送信処理 (ユーザ定義) のフローチャートを示します。

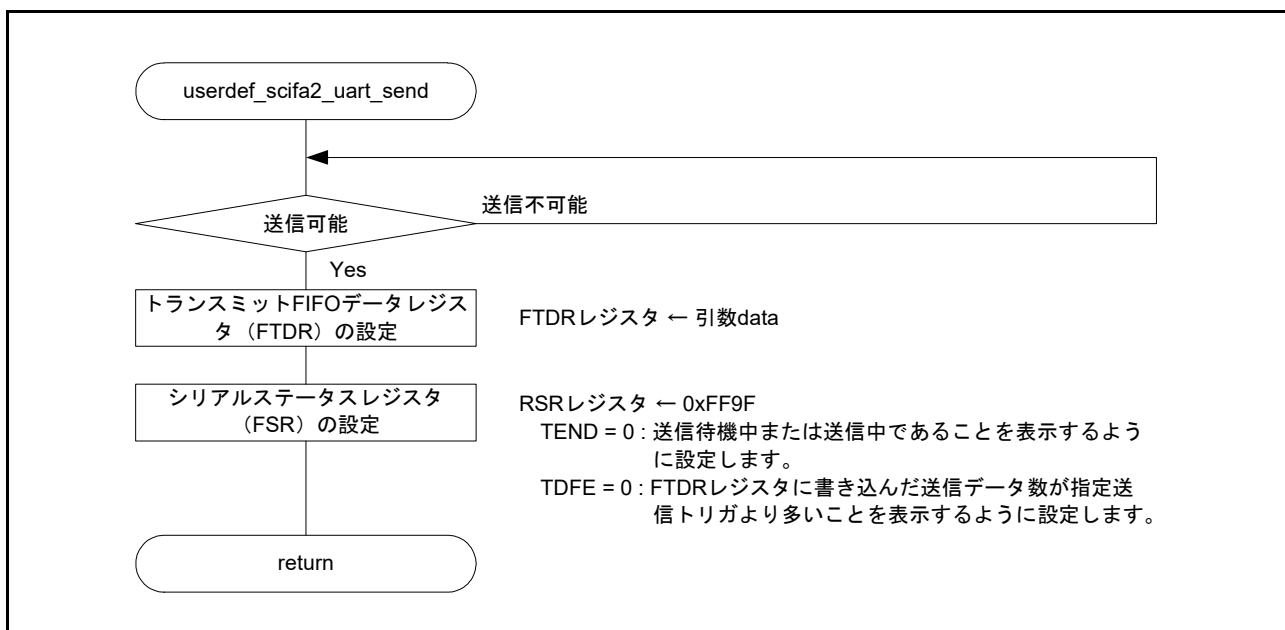


図 6.13 SCIFA チャンネル 2 データ送信処理 (ユーザ定義)

6.8.10 SCIFA チャンネル 2 データ受信処理 (ユーザ定義)

図 6.14 に SCIFA チャンネル 2 データ受信処理 (ユーザ定義) のフローチャートを示します。

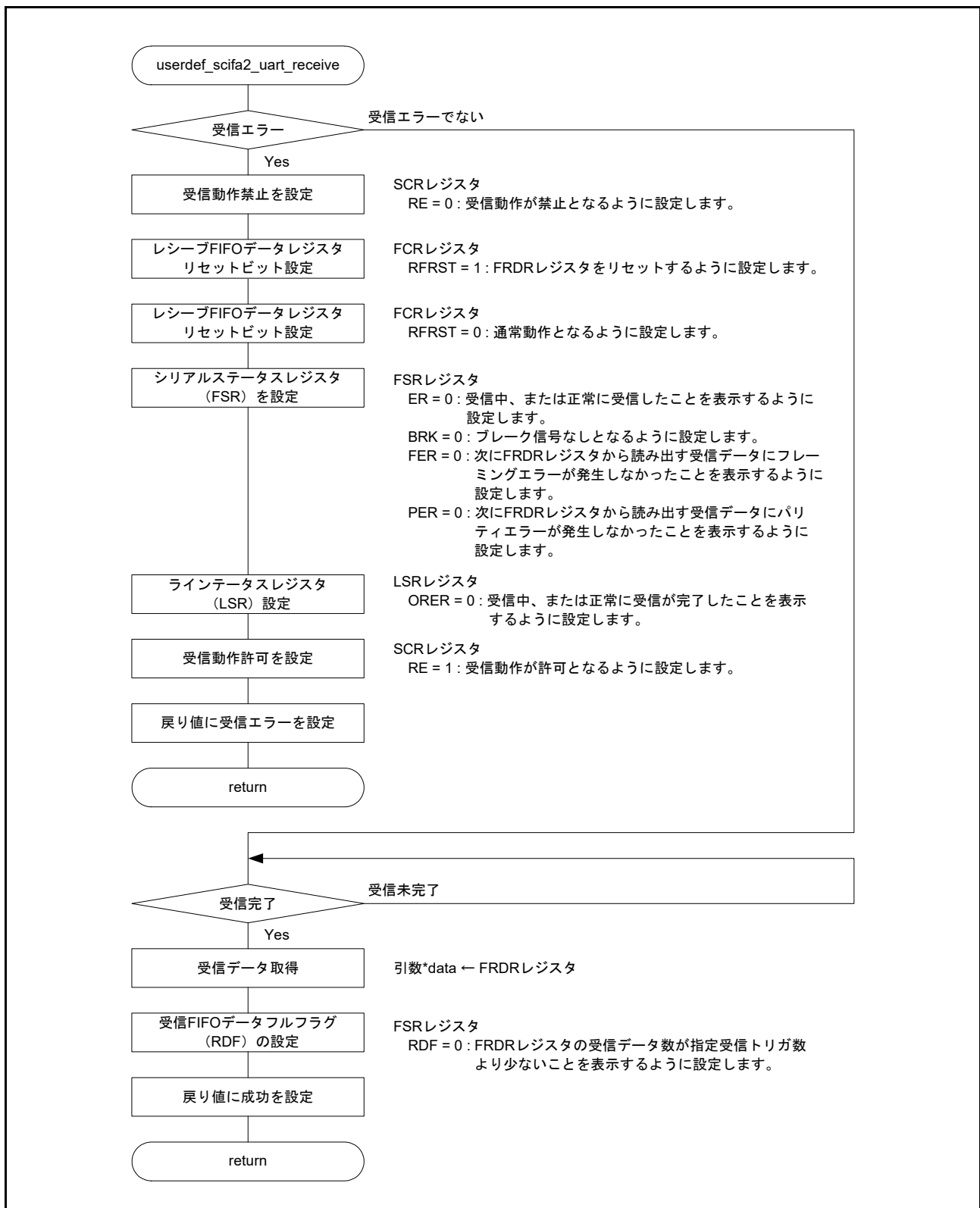


図 6.14 SCIFA チャンネル 2 データ受信処理 (ユーザ定義)

7. サンプルプログラム

サンプルプログラムは、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。

8. 参考ドキュメント

- ユーザーズマニュアル：ハードウェア

RZ/T1 グループ ユーザーズマニュアルハードウェア編

(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。)

RZ/T1 Evaluation Board RTK7910022C00000BR ユーザーズマニュアル

(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

IAR 統合開発環境 (IAR Embedded Workbench® for Arm) に関しては、IAR ホームページから入手してください。

(最新版を IAR 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	FIFO 内蔵シリアルコミュニケーションインタフェース (SCIFA) アプリケーションノート
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
0.10	2015.02.03	—	初版発行
0.20	2015.03.10	12	6.2.4 必要メモリサイズ 表6.2 必要メモリサイズ 修正
		12	6.3 固定幅整数一覧 表6.3 サンプルプログラムで使用する固定幅整数 修正
1.00	2015.04.10	—	Web掲載に際しRevのみ変更
1.10	2015.07.06	2. 動作環境	
		5	表2.1 動作環境 統合開発環境 表記一部修正、追加
		6. ソフトウェア説明	
		12	6.2.4 説明文 参照を追加
		12	表6.2 タイトルを一部追加
		12	表6.3 追加
		13	表6.4 追加
1.20	2015.12.03	2. 動作環境	
		5	表2.1 動作環境 統合開発環境 一部修正
1.30	2017.04.05	2. 動作環境	
		5	表2.1 動作環境 統合開発環境、PC用USBシリアルポートドライバの内容変更
		6. ソフトウェア説明	
		—	6.2.4 必要メモリサイズ 削除
1.40	2018.06.07	2. 動作環境	
		5	表2.1 動作環境 統合開発環境の内容変更
		5. ハードウェア説明	
		8	図5.1 ハードウェア構成例 モジュール名変更
		8. 参考ドキュメント	
		29	IAR 統合開発環境名変更

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）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

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

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違っていると、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。型名が異なる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、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、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。
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。

6. 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。
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、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。
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。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. 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。
10. 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。
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。
12. 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.4.0-1 2017.11)



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

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

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■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：<https://www.renesas.com/contact/>