

RH850/U2C Group

FlexRay アプリケーションノート

要旨

本アプリケーションノートは、FlexRay を使用した動作例をまとめたものです。FlexRay プログラムはユーザエリア上にあるものとします。

本資料およびプログラムは、RH850/U2C 搭載機能の理解促進を意図するものであり、量産設計を対象とするものではありません。

また、最新のマニュアル、正誤表、テクニカルアップデートや、開発環境の更新を反映しておりません。該当機能を使用される場合には、本プログラムは参考として扱い、最新のドキュメントや開発環境にて、お客様の責任において行ってください。

適用

この資料は、RH850/U2Cx に適用されます。

【注】 Configuration Setting Area へダウンロードする場合は set_csa.c に任意のオプションバイトを設定し、ダウンロードを許可してオプションバイトを書き換えてください。詳細については RH850/U2C シリーズ Startup アプリケーションノートを参照してください。

- (1) プロジェクト・ツリーから「*****(デバッグ・ツール)」を選択
- (2) 「ダウンロード・ファイル設定」のタブを選択
- (3) 「Configuration Setting Area へのダウンロードを許可する」 = “はい” に設定

目次

1.	はじめに	4
1.1	使用機能	4
2.	初期設定	5
2.1	ポート設定	5
2.2	FlexRay モジュールの初期化	6
2.3	FlexRay クロック	7
2.3.1	FlexRay サンプリングクロック、および FlexRay 通信時間単位の設定	7
2.4	FlexRay 通信パラメータなどの設定	9
2.4.1	コミュニケーションサイクル設定	9
2.4.2	ノードの役割 (Startup ノードの設定)	11
2.5	Message RAM の設定	13
2.5.1	MessageRAM のバッファ構成	13
	『設定手順例の説明』	14
2.5.2	データ部とヘッダ部の構成	17
2.5.3	データポインタ	18
3.	通信コントローラの状態	22
3.1	通信コントローラ状態遷移図	22
3.2	Default Config~Ready 状態	23
3.3	Wakeup 状態	24
3.3.1	RECEIVED_HEADER または COLLISION_HEADER 時の処理例	29
3.3.2	RECEIVED_WUP、COLLISION_WUP または TRANSMITTED 時の処理例	30
3.3.3	COLLISION_UNKNOWN 時の処理例	31
3.4	Startup 状態	32
3.4.1	Cold_start ノードの動作	33
3.4.2	Non-Cold_start ノードの動作	37
4.	MessageRAM へのデータ設定と MessageRAM からのデータ読み出し	39
4.1	送信フレームデータの構成	39
4.2	IBF (インプットバッファ) : MessageRAM への転送	41
4.2.1	転送方法	41
4.3	OBF (アウトプットバッファ) : MessageRAM からの読み出し	43
4.3.1	転送方法	43
5.	フレーム送受信	45
5.1	フレーム送信	45
5.2	フレーム受信	46
6.	割り込み	48
6.1	割り込み制御	48
6.2	FlexRay0 割り込み、FlexRay1 割り込み	48
6.3	割り込み要求クリアについて	49
7.	タイマ	50

7.1	タイマ 0	50
7.2	タイマ 1	51
8.	ストップウォッチタイマ	52
9.	ネットワーク管理機能	53
10.	受信 FIFO	54
10.1	FIFO フィルタリング	54
10.1.1	FIFO リジェクションフィルタの構成	54
10.2	FIFO バッファの読み出し	56
10.3	FIFO バッファへの書き込み	58
12.	改訂記録	59

1. はじめに

本アプリケーションノートでは、RH850/U2Cx の FlexRay の使用方法およびソフトウェアの作成例を掲載しています。

1.1 使用機能

本アプリケーションノートで使用する RH850/U2Cx のハードウェア機能を以下に示します。

- FlexRay

2. 初期設定

RH850/U2Cx では、FlexRay 通信機能を使用するためには、以下の初期設定を行います。

- ポート設定
- 割り込み設定
- FlexRay モジュールの初期化
- FlexRay クロック設定
- FlexRay 通信パラメータなどの設定
- MessageRAM の設定

以下に、それぞれの初期化の方法を示します。

2.1 ポート設定

本動作例では、ポート 24 に FlexRay 端子を割り当てています。FlexRay 端子の設定を表 2-1 に、各 FlexRay 端子の機能を表 2-2 に示します。

表 2-1 FlexRay モジュールの端子設定

ポート名	端子名	ポートコントロールレジスタ (PCRn_m) の設定値
P24_12	FLX0TXDA	PCR24_12 = 0x00000046
P24_11	FLX0RXDA	PCR24_11 = 0x00000056
P24_10	FLX0TXENA	PCR24_10 = 0x00000045
P24_7	FLX0TXDB	PCR24_7 = 0x00000045
P24_8	FLX0RXDB	PCR24_8 = 0x00000056
P24_9	FLX0TXENB	PCR24_9 = 0x00000045

表 2-2 FlexRay モジュールの端子説明

端子名	機能
FLX0TXDA	チャンネル A 送信データ出力端子
FLX0RXDA	チャンネル A 受信データ入力端子
FLX0TXENA	チャンネル A 送信データ許可端子 “H”：送信禁止 “L”：送信許可
FLX0TXDB	チャンネル B 送信データ出力端子
FLX0RXDB	チャンネル B 受信データ入力端子
FLX0TXENB	チャンネル B 送信データ許可端子 “H”：送信禁止 “L”：送信許可

2.2 FlexRay モジュールの初期化

パワーオンリセット直後、FLXAnFRSUCC1 レジスタによる CLEAR_RAMs コマンド実行直後は、FlexRay モジュールの内部 RAM の初期化（すべて"0"）が実行されます。

内部 RAM の初期化が実行されている間は、FlexRay の各レジスタへの設定ができないため、内部 RAM の初期化処理が完了したことを確認します。

2.3 FlexRay クロック

FlexRay に供給されるサンプリングクロックは、高速周辺クロック CLKC_HSB (80MHz) です。

2.3.1 FlexRay サンプリングクロック、および FlexRay 通信時間単位の設定

FlexRay モジュールで使用する各クロックの設定について説明します。これらのクロックは、FlexRay モジュール内で高速周辺クロック (CLKC_HSB) を分周して生成します。

FlexRay で通信を行う前に、FlexRay バス値をサンプリングするクロック、FlexRay ネットワーク内で共通の時間単位である Macrotick (以下 MT)、ノード内のローカルな時間単位である Microtick (以下 uT) などを、それぞれ設定します。

高速周辺クロック (CLKC_HSB) と FlexRay 通信速度との関係を表 2-3 に示します。

表 2-3 高速周辺クロックと FlexRay 通信速度との関係

高速周辺クロック (CLKC_HSB)	FLXAnFRPRTC1 レジスタ BRP[1:0]設定値	ビットクロック (Bit/s)	通信速度
80MHz	00	100ns	10Mbps
80MHz	01	200ns	5Mbps
80MHz	1x	400ns	2.5Mbps

■設定例

高速周辺クロック（CLKC_HSB）80MHz、通信ビットレート 10Mbps 時の各時間単位の設定値例を表 2-4 に、設定手順例を図 2-1 に示します。

表 2-4 各時間単位の設定値例

名称	設定値
サンプリング周期	12.5ns
通信速度	10Mbps
Microtick	25ns
Macrotick	1us
コミュニケーションサイクル	1ms

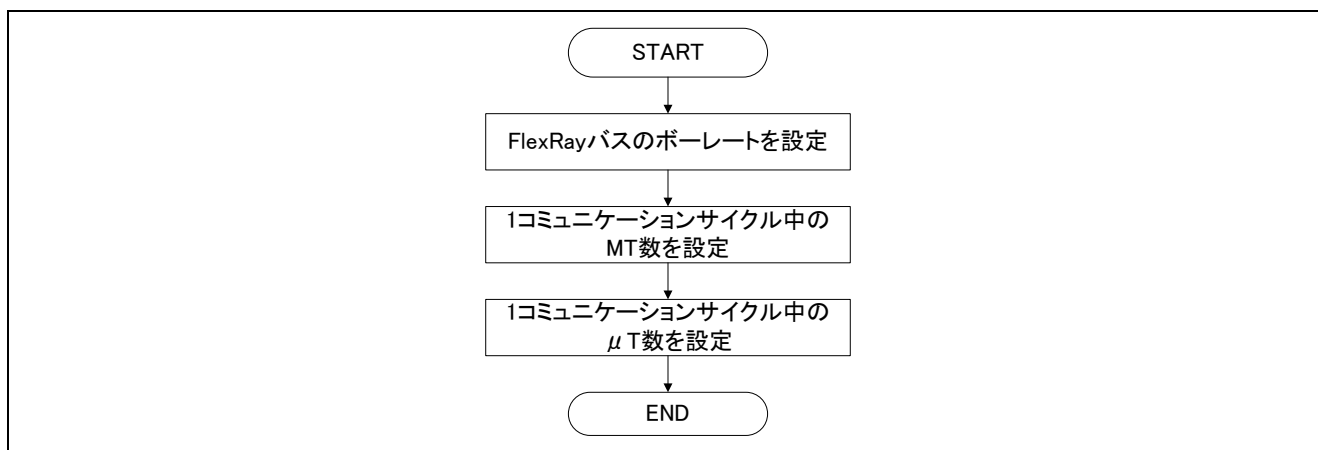


図 2-1 各時間単位の設定手順例

『設定手順例の説明』

高速周辺クロック（CLKC_HSB）80MHz 時、FLXAnFRPRTC1 レジスタの BRP ビット="00"を設定すると、サンプリング周期は 80MHz（12.5ns）（分周なし）となります。

通信ビットレートはサンプリング周期の 1/8（FlexRay 仕様にて固定）なので $80\text{MHz}/8 \Rightarrow 10\text{Mbps}$ です。

uT は、FLXAnFRPRTC1 レジスタの BRP ビット = "00"の場合、高速周辺クロック（CLKC_HSB）の 2 分周（固定）なので、1uT 長は $12.5\text{ns} \times 2 = 25\text{ns}$ となります。

コミュニケーションサイクル長は uT 長（25ns） $\times 40,000 \Rightarrow 1\text{ms}$ です。

Macrotick 周期は 1 コミュニケーションサイクル $1,000\text{MT} = 40,000\text{uT}$ なので、 $1\text{MT} \Rightarrow 40 \times \text{uT} \Rightarrow 1\text{us}$ となります。

- 【注】
- 1MT 当たりの uT 数 (pMicroPerMacroNom) を設定するレジスタはありません。pMicroPerMacroNom は、1 コミュニケーションサイクル当たりの MT 数および 1 コミュニケーションサイクル当たりの uT 数を設定することにより、通信コントローラが自動的に計算します。
 - 設定は、CONFIG 状態 (FLXAnFRCCSV レジスタの POCS ビット="001111")で行ってください。それ以外の状態では FLXAnFRPRTC1、FLXAnFRGTUC1、FLXAnFRGTUC2 レジスタに書き込みを行うことはできません。

2.4 FlexRay 通信パラメータなどの設定

2.4.1 コミュニケーションサイクル設定

FlexRay モジュールでは、Static セグメント・Dynamic セグメントおよび、NIT 開始位置を設定することでコミュニケーションサイクル構成を決定します。コミュニケーションサイクルの構成を図 2-2 に示します。

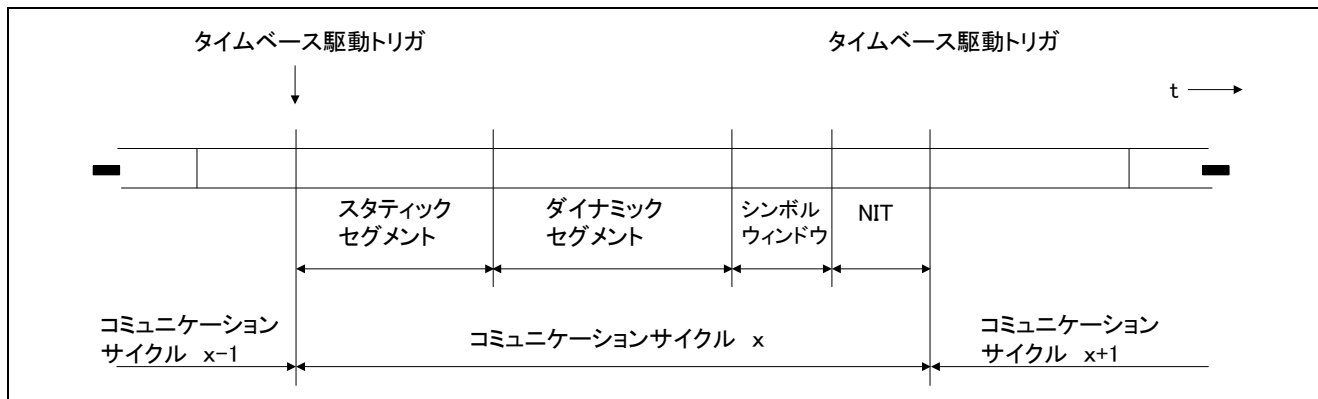


図 2-2 コミュニケーションサイクル構成

コミュニケーションサイクルの設定手順例を以下に示します。コミュニケーションサイクルの設定値例を表 2-5 に、設定手順例を図 2-3 に示します。

表 2-5 コミュニケーションサイクルの設定値例

名称	設定値
1 コミュニケーションサイクル長	1000 MT
Static スロット長	54MT
Static スロット数	6 slot
ActionPoint 位置	5MT
ミニスロット長	24MT
ミニスロット数	6slot
ミニスロット ActionPoint 位置	2MT
NIT 開始位置	990MT

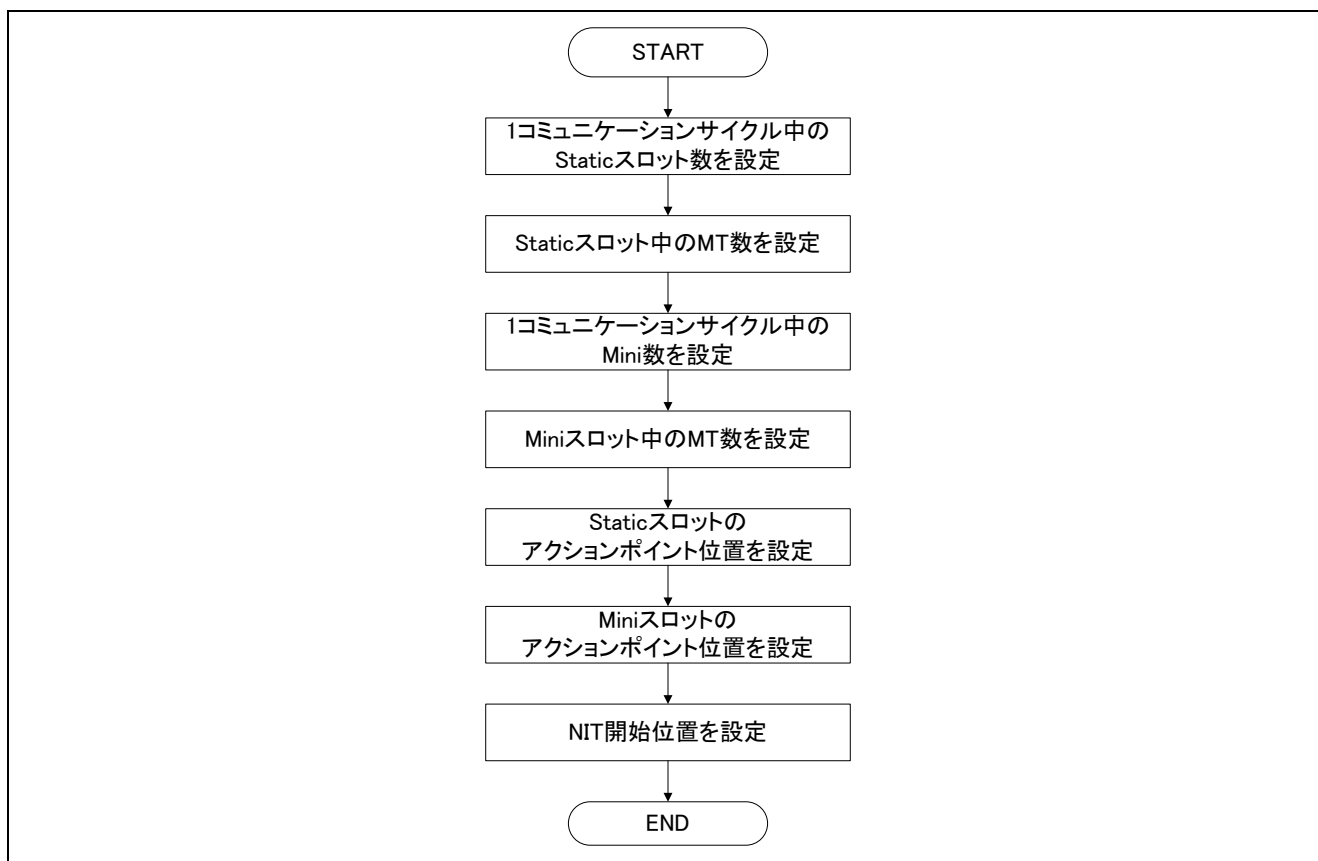


図 2-3 コミュニケーションサイクル設定手順例

- 【注】
1. Symbol window の開始位置・長さを設定するレジスタはありません。Dynamic セグメント終端～NIT 開始位置までが Symbol window になります。
 2. コミュニケーションサイクルの設定は、CONFIG 状態（FLXAnFRCCSV レジスタの POCS ビット = “001111”）で行ってください。それ以外の状態では FLXAnFRGTUC2、FLXAnFRGTUC4、FLXAnFRGTUC7～9 レジスタに書き込みを行うことはできません。

2.4.2 ノードの役割 (Startup ノードの設定)

FlexRay では、「Startup フレーム送信の可否」に関して、ノードの役割を初期設定時に決定しておく必要があります。この設定により、Startup 時のノードの状態遷移経路が変化します。

Startup フレームは、必ず同時に Sync フレームである必要があります。

FLXAnFRSUCC1 の各種フラグ設定時の状態遷移経路を図 2-4 に示します。

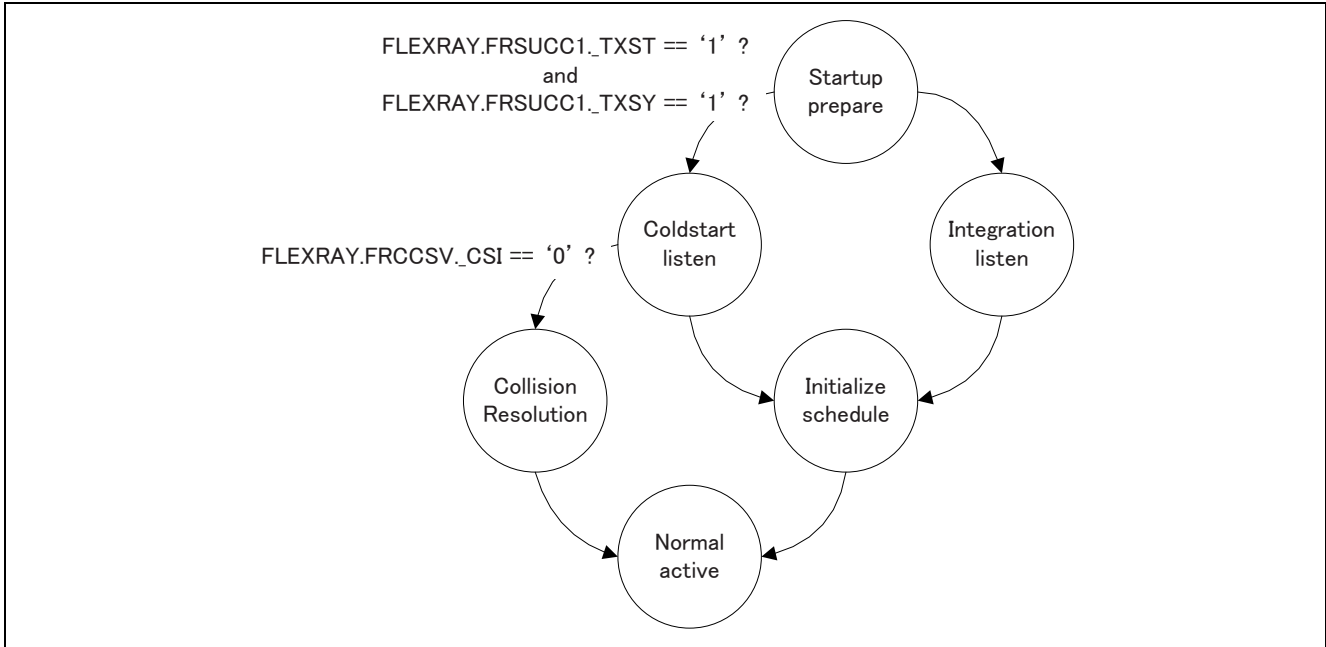


図 2-4 ノードの役割設定

Startup ノードとして構成する場合の設定手順例と参考プログラムを示します。

表 2-6 Startup ノードとして構成する場合の設定例

名称	設定値
Startup フレーム送信	yes
Sync フレーム送信	yes
Coldstart Inhibit フラグ	yes
Startup リトライ回数	31 回

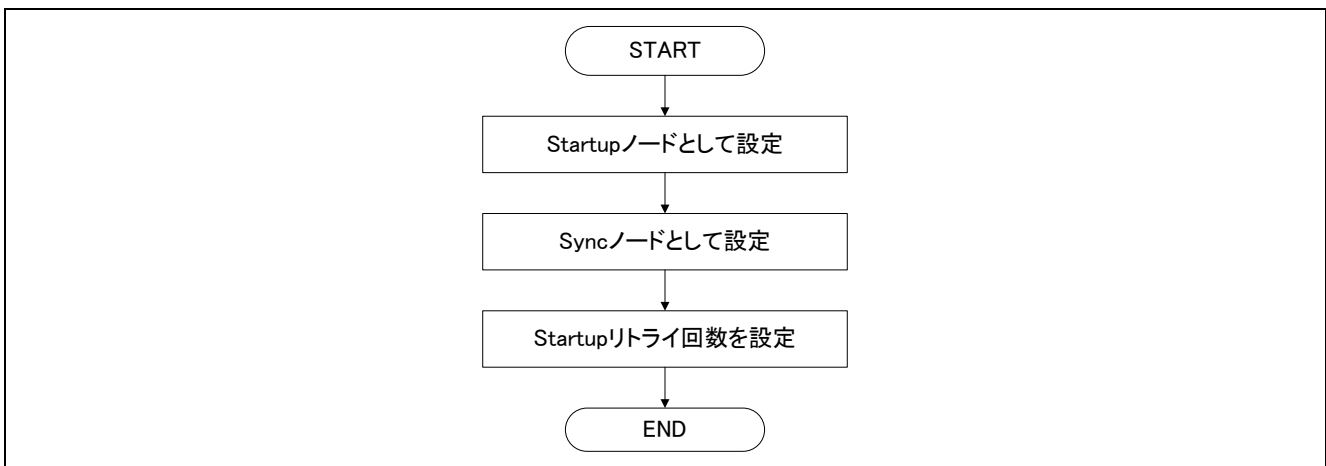


図 2-5 制御手順例

『設定手順例の説明』

FLXAnFRSUCC1 レジスタの TXST ビットと TXSY ビットで Startup ノードおよび Sync ノードとしての動作を許可します。

Startup ノードとして設定する場合、FLXAnFRSUCC1 レジスタの TXST ビットを"1"に、Sync ノードとして設定する場合 TXSY ビットを"1"に設定してください。

Startup/Sync フレームはメッセージバッファ 0、およびメッセージバッファ 1 からのみ送信されます。

Startup/Sync フレーム として使用するフレームの ID (pKeySlotId) を、メッセージバッファ 0、およびメッセージバッファ 1 のヘッダ部分で設定してください。

FLXAnFRMRC レジスタの SPLM ビットに"0"を設定した場合、メッセージバッファ 0 のみ使用されません。"1"を設定した場合、メッセージバッファ 0 とメッセージバッファ 1 が使用されます。この場合、異なるペイロードデータの Startup / Sync フレームをチャンネル A、チャンネル B に送信可能です。

- 【注】
1. 設定は、CONFIG 状態 (FLXAnFRCCSV レジスタの POCS ビット = "001111") で行ってください。それ以外の状態では FLXAnFRSUCC1 レジスタの TXST、TXSY、CSA に書き込みを行うことはできません。
 2. FLXAnFRCCSV レジスタの CSI (Coldstart Inhibit) ビットが"1"のとき、Cold_start 禁止状態です。このフラグは、READY 状態に遷移すると自動的に"1"になります。Cold_start を許可にするためには、FLXAnFRSUCC1 レジスタの CMD ビットに"1001" (ALLOW_COLDSTART コマンド) を書き込むことにより、FLXAnFRCCSV レジスタの CSI ビットを"0"クリアしてください。
 3. FLXAnFRSUCC1 レジスタの TXSY ビット = 0、FLXAnFRSUCC1 レジスタの TXST ビット = 1 の設定は無効です。
 4. FLXAnFRMRC レジスタの SPLM ビットの設定と FLXAnFRWRHS1 レジスタの CH ビット (チャンネルフィルタ) の設定の整合性を確認してください。

2.5 Message RAM の設定

2.5.1 MessageRAM のバッファ構成

FlexRay モジュールには、8KB の Message RAM が内蔵されており、バッファ個数は 0~128 個の範囲で任意に設定可能です。

FLXAnFRMRC レジスタの SEC ビットの設定により、特定メッセージバッファへの書き込み、および Static セグメントへの送信の許可/禁止設定をすることができます。

MessageRAM 内のバッファ構成を図 2-6 に示します。

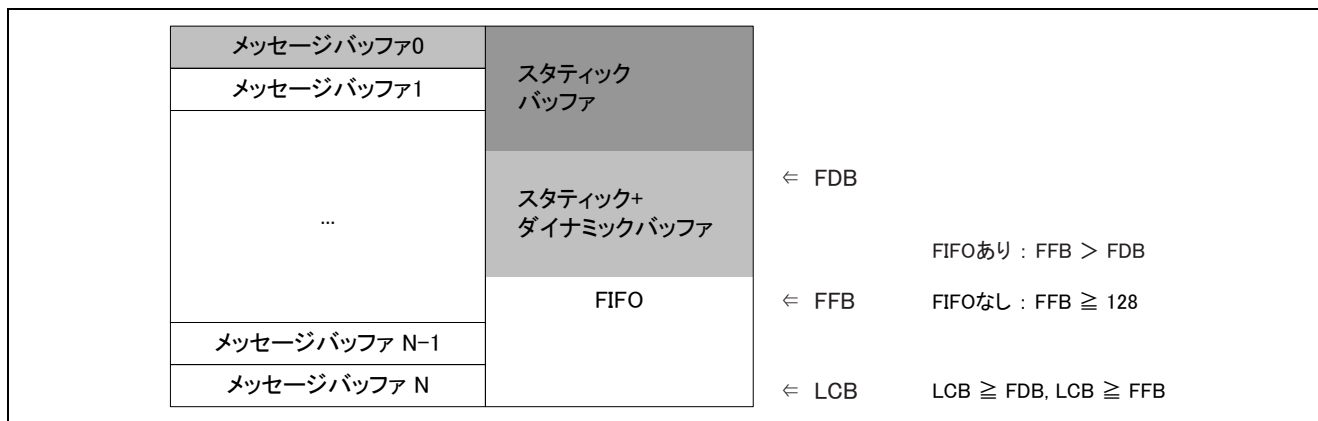


図 2-6 メッセージバッファ構成

【注】

- FLXAnFRMRC レジスタの設定時、下記に注意してください。
FIFO バッファのサイズは、FFB ビットと LCB ビットで決まります。
Dynamic バッファと FIFO バッファを使う場合、FFB ビットの設定値は FDB ビットの設定値より大きい値を設定してください。
Dynamic バッファを使用しない場合は、FDB ビットに 128 か、128 より大きい値を設定してください。
FIFO バッファを使用しない場合は、FFB ビットに 128 か、128 より大きい値を設定してください。
LCB ビットの設定値は、FIFO バッファを使用する場合は FFB ビットの設定値より、さらに Dynamic バッファを使用する場合は FDB ビットの設定値より大きい値を設定してください。
- FlexRay モジュールは、MessageRAM の構成の設定が間違っているかどうかはチェックしません。
- MessageRAM の設定は、CONFIG 状態 (FLXAnFRCCSV レジスタの POCS ビット = "001111") で行ってください。
それ以外の状態 FLXAnFRMHDC、FLXAnFRMRC レジスタに書き込みを行うことはできません。
- ハードウェアリセット後、Default_config 状態 (FLXAnFRCCSV レジスタの POCS ビット = "000000") となります。
初期設定前に CHI CONFIG コマンド (FLXAnFRSUCC1 レジスタの CMD ビット = "0001") により Config 状態へ遷移してください。

■ヘッダ部構成例（1）（MessageRAM に Static バッファのみ構成する場合）

MessageRAM に Static バッファのみを構成する場合の構成例を表 2-6、設定例を図 2-8 に示します。

図 2-7 Message RAM の構成例

RAM (4byte)	バッファ番号	構成	内容
0...3	メッセージバッファ 0	↓Static バッファ	ヘッダ部
4...7	メッセージバッファ 1		
		
52...55	メッセージバッファ 6		
56...59	メッセージバッファ 7		←LCB
60			データ部
		
2047			

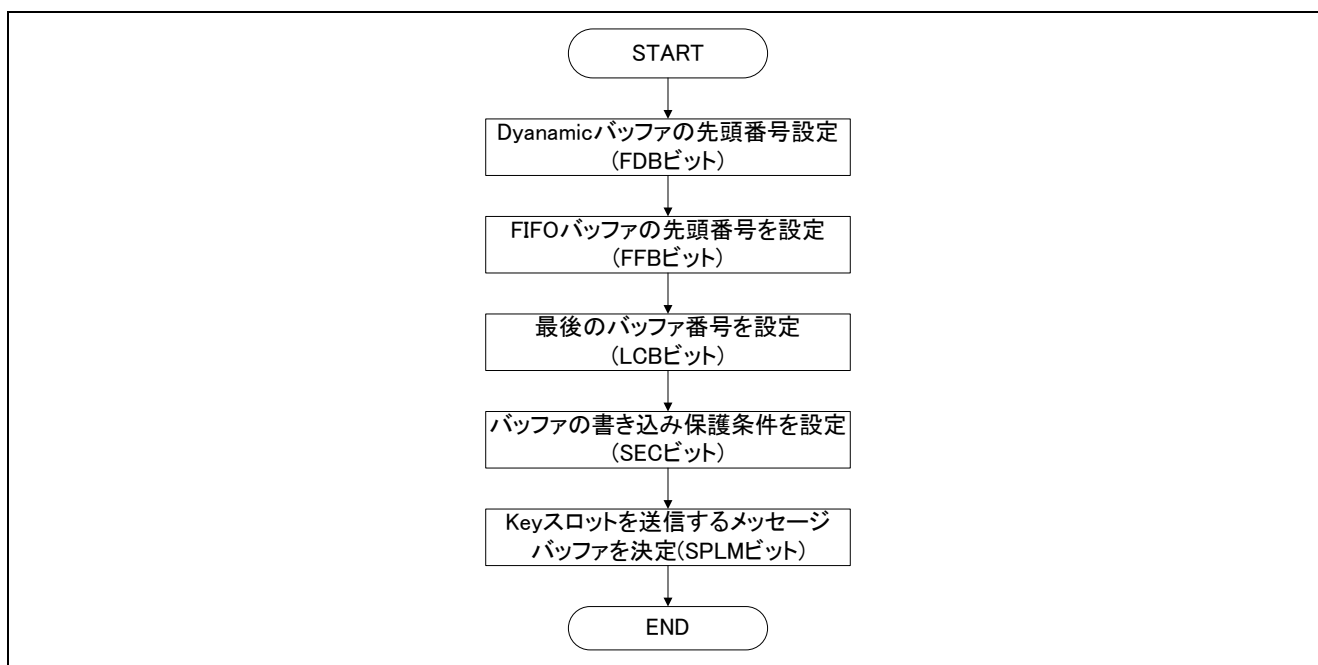


図 2-8 設定手順例

『設定手順例の説明』

本例では、Static バッファのみ構成するため、Dynamic バッファと FIFO バッファは構成しません。FLXAnFRMRC レジスタの FDB ビット = "128"、FLXAnFRMRC レジスタの FFB ビット = "128" を設定します。

メッセージバッファの個数が 8 個のため、FLXAnFRMRC レジスタの LCB ビット = "7" を設定します。

- ヘッダ部構成例 (2) (MessageRAM に Static バッファと FIFO バッファを構成する場合)
MessageRAM に Static バッファと FIFO バッファを構成する場合の構成例を表 2-7 に示します。

表 2-7 Message RAM の構成例

RAM (4byte)	バッファ番号	構成	内容
0...3	メッセージバッファ 0	↓Static バッファ	ヘッダ部 ←FFB ←LCB
4...7	メッセージバッファ 1		
8...11	メッセージバッファ 2		
12...15	メッセージバッファ 3		
16			データ部
		
2047			

『設定手順例の説明とプログラム例』

メッセージバッファの個数が 4 個のため、FLXAnFRMRC レジスタの LCB ビット = "3" を設定します。

本例では、Static バッファを 2 個、FIFO バッファを 2 個構成します。Dynamic バッファは構成しないため、FLXAnFRMRC レジスタの FDB ビット = "128" を設定します。

FIFO バッファは 2 個のため、FLXAnFRMRC レジスタの FFB ビット = "2"、FLXAnFRMRC レジスタの LCB ビット = "3" となります。

必要に応じて flexray.c ファイルのマクロ定義値 : MRAMC_SET を変更してください。

■ヘッダ構成例 (3) (Static バッファ、Dynamic バッファを構成する場合)

MessageRAM に Static バッファ、Dynamic バッファを構成する場合の構成例を表 2-8 に示します。

表 2-8 Message RAM の構成例

RAM (4byte)	バッファ番号	構成	内容
0...3	メッセージバッファ 0	↓Static バッファ	ヘッダ部
4...7	メッセージバッファ 1		
8...11	メッセージバッファ 2		
12...15	メッセージバッファ 3		
16...19	メッセージバッファ 4		
20...23	メッセージバッファ 5		
24...27	メッセージバッファ 6	↓Static+Dynamic バッファ	←FDB
28...31	メッセージバッファ 7		
32...35	メッセージバッファ 8		
36...39	メッセージバッファ 9		
40...43	メッセージバッファ 10		
44...47	メッセージバッファ 11		
48			←LCB
		データ部
2047			

『設定手順例の説明とプログラム例』

メッセージバッファの個数が 12 個のため、FLXAnFRMRC レジスタの LCB ビット = "11" を設定します。

本例では、Static バッファを 6 個、Dynamic バッファを 6 個構成します。FIFO バッファは構成しないため FLXAnFRMRC レジスタの FFB ビット = "128" を設定します。

Dynamic バッファのヘッダ部の開始番号は "6" のため、FLXAnFRMRC レジスタの FDB ビット = "6" を設定します。Dynamic バッファは 6 個のため、FLXAnFRMRC レジスタの LCB ビット = "11" です。

必要に応じて flexray.c ファイルのマクロ定義値 : MRAMC_SET を変更してください。

2.5.2 データ部とヘッダ部の構成

FlexRay モジュールでは、各バッファのデータ長を 0~254 バイトの範囲 2 バイト単位で任意に設定することができます。

Message RAM 内部では、フレーム情報はヘッダ部とデータ部に分かれています。

メッセージバッファごとにヘッダ部（サイズは 16bytes 固定）とデータ部（サイズは可変）が必要になります。

送受信フレーム設定時、ヘッダ部は FLXAnFRWRHS1~FLXAnFRWRHS3 レジスタで、データ部は FLXAnFRWRDS1~FLXAnFRWRDS64 レジスタで設定します。

各レジスタで設定したデータは、IBF（インプットバッファ）を経由して、MessageRAM 中のメッセージバッファへ転送されます。転送方法については、『3. MessageRAM へのデータ設定と MessageRAM からのデータ読み出し』を参照してください。本章では、構成についてのみ説明します。

Message RAM 内部の構成例を図 2-9 に、表 2-9 にバッファ当たりのデータ長と確保可能なバッファ数を示します。

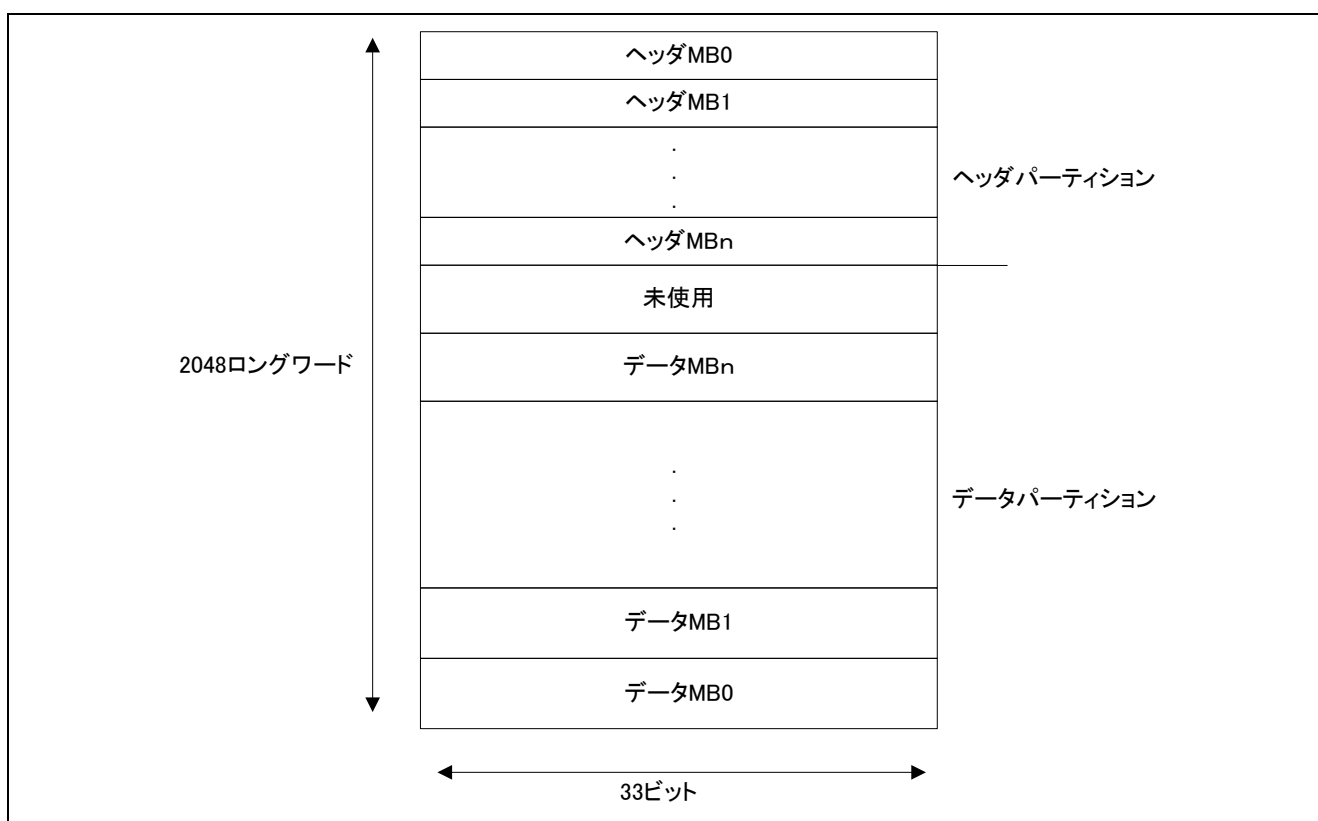


図 2-9 Message RAM 内部の構成

表 2-9 バッファ当たりのデータ長と確保可能なバッファ数

バッファ当たりのデータ長	確保可能なバッファ数	備考
254 bytes	30 個	
128 bytes	56 個	
48 bytes	128 個	

- 【注】
- 各バッファのデータ長は、全バッファの「ヘッダ部+データ部」の合計が 8KB に収まるように設定してください。全バッファのデータ長を同一に設定した場合、確保可能なバッファ個数を示します。
 - FIFO バッファのヘッダ部に設定された Payload length 値とデータ部の長さは、すべて同じにしてください。それらは、FLXAnFRWRHS2 レジスタの PLC0~PLC6 ビットと FLXAnFRWRHS3 レジスタの DP0~DP10 ビットで設定できます。

2.5.3 データポインタ

FlexRay モジュールは、フレーム情報をメッセージバッファのデータ部に読み書きする際に、ヘッダ部のデータポインタからデータ部の先頭アドレスを算出します。

データポインタは、Message RAM の先頭アドレス（メッセージバッファ 0 のアドレス）を「H'0」として、「データ部先頭アドレス÷4 バイト」の値を設定します。FLXAnFRWRHS3 レジスタの DP0~DP10 ビットにデータポインタ値を設定してください。

- 【注】
1. 上記仕様のため、データ部先頭アドレスは必ず「4 の倍数」となるように設定してください。
 2. 動作中は、FLXAnFRWRHS3 レジスタの DP0~DP10 ビットの値を変更しないでください。
 3. データポインタがヘッダ部を指している複数のバッファで同じアドレスを指定しているなど、間違った値を設定した場合も、FlexRay モジュールは、メッセージバッファヘッダデータ格納を実行します。その場合、データ部の Payload データがロストしたり、不正な上書きが実行されたりする場合があります。
誤った構成のために、ヘッダ部にフレームデータを書こうとすると、FLXAnFRMHDF レジスタの WAHP ビットが"1"になります。この場合、ヘッダ部は上書きに対して保護され、データはロストします。

■データポインタ算出例（1）

MessageRAM の構成例と各バッファのデータポインタ値を表 2-10 に示します。

表 2-10 Message RAM の構成例

RAM (4byte)	バッファ番号	構成		内容	
0...3	メッセージバッファ 0	Static バッファ、DP = 12		ヘッダ部	
4...7	メッセージバッファ 1	Static バッファ、DP = 14			
8...11	メッセージバッファ 2	Static バッファ、DP = 16			
12		MB0 データ 1	MB0 データ 0	データ部←DP MB0	
13		未使用*	MB0 データ 2		
14		MB1 データ 1	MB1 データ 0		←DP MB1
15		未使用*	MB1 データ 2		
16		MB2 データ 1	MB2 データ 0		←DP MB2
17		未使用*	MB2 データ 2		
18...2046		未使用			
2047		未使用			

【注】 * この領域にはアクセスできません。

『制御手順例の説明』

メッセージバッファの個数が 3 個のため、FLXAnFRMRC レジスタの LCB ビット = "2" となります。この場合、データポインタは (FLXAnFRMRC レジスタの LCB + 1) * 4 = "12" 以上である必要があります。

Static フレームのデータ長は 6 バイトであるため、FLXAnFRMHDC レジスタの SFDL ビット = "3" を設定してください。

各メッセージバッファのデータポインタは下記となります。

メッセージバッファ 0 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "12"

メッセージバッファ 1 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "14"

メッセージバッファ 2 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "16"

本例では、FIFO バッファ、Dynamic バッファは使用しないため、FLXAnFRMRC レジスタの FDB ビット = "128"、FLXAnFRMRC レジスタの FFB ビット = "128" となります。

■ データポインタ算出例 (2)

MessageRAM の構成例と各バッファのデータポインタ値を表 2-11 に示します。

表 2-11 Message RAM の構成例

RAM (4byte)	バッファ番号	構成		内容
0...3	メッセージバッファ 0	Static バッファ、DP = 2046		ヘッダ部 ←FDB ←LCB
4...7	メッセージバッファ 1	Static バッファ、DP = 2044		
8...11	メッセージバッファ 2	Dynamic バッファ、DP = 2042		
12...15	メッセージバッファ 3	Dynamic バッファ、DP = 2040		
16...2039		未使用		データ部 ←DP MB3 ←DP MB2 ←DP MB1 ←DP MB0
2040		MB3 データ 1	MB3 データ 0	
2041		MB3 データ 3	MB3 データ 2	
2042		MB2 データ 1	MB2 データ 0	
2043		MB2 データ 3	MB2 データ 2	
2044		MB1 データ 1	MB1 データ 0	
2045		未使用*	MB1 データ 2	
2046		MB0 データ 1	MB0 データ 0	
2047		未使用*	MB0 データ 2	

【注】 * この領域にはアクセスできません。

『制御手順例の説明』

メッセージバッファの個数が 4 個のため、FLXAnFRMRC レジスタの LCB ビット = "3" となります。この場合、データポインタは (FLXAnFRMRC レジスタの LCB + 1) * 4 = "16" 以上である必要があります。

本例では、2 個の Static バッファ、2 個の Dynamic バッファを設定します。FIFO バッファは、使用しません。FLXAnFRMRC レジスタの FDB ビット = "2"、FLXAnFRMRC レジスタの LCB ビット = "3" および FLXAnFRMRC レジスタの FFB ビット = "128" を設定します。

Static フレームのデータ長が 6 バイトのため、FLXAnFRMHDC レジスタの SFDL ビット = "3" を設定します。

Dynamic バッファの Payload 長が 8 バイトのため、メッセージバッファ 2,3 の FLXAnFRWRHS2 レジスタの PLC ビット = "4" を設定します。

各メッセージバッファのデータポインタは下記となります。

メッセージバッファ 0 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "2046"

メッセージバッファ 1 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "2044"

メッセージバッファ 2 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "2042"

メッセージバッファ 3 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "2040"

■データポインタ算出例 (3)

MessageRAM の構成例と各バッファのデータポインタ値を表 2-12 に示します。

表 2-12 Message RAM の構成例

RAM (4byte)	バッファ番号	構成		内容	
0...3	メッセージバッファ 0	Static バッファ、DP = 1100		ヘッダ部 ←FDB ←FFB ←LCB	
4...7	メッセージバッファ 1	Static バッファ、DP = 1101			
8...11	メッセージバッファ 2	Dynamic バッファ、DP = 1985			
12...15	メッセージバッファ 3	Dynamic バッファ、DP = 1987			
16...19	メッセージバッファ 4	FIFO バッファ、DP = 79			
20...23	メッセージバッファ 5	FIFO バッファ、DP = 81			
24...27	メッセージバッファ 6	FIFO バッファ、DP = 83			
28...31	メッセージバッファ 7	FIFO バッファ、DP = 85			
32		未使用		データ部	
33...78		未使用			
79		MB4 データ 1	MB4 データ 0		←DP MB4
80		MB4 データ 3	MB4 データ 2		
81		MB5 データ 1	MB5 データ 0		←DP MB5
82		MB5 データ 3	MB5 データ 2		
83		MB6 データ 1	MB6 データ 0		←DP MB6
84		MB6 データ 3	MB6 データ 2		
85		MB7 データ 1	MB7 データ 0		←DP MB7
86		MB7 データ 3	MB7 データ 2		
87...1099		未使用			
1100		MB0 データ 1	MB0 データ 0		←DP MB0
1101		MB1 データ 1	MB1 データ 0		←DP MB1
1102...1984		未使用			
1985		MB2 データ 1	MB2 データ 0		←DP MB2
1986		未使用*	MB2 データ 2		
1987		MB3 データ 1	MB3 データ 0		←DP MB3
1988		未使用*	MB3 データ 2		
1989...2046		未使用*			
2047		未使用*			

【注】 * この領域にはアクセスできません。

『制御手順例の説明』

メッセージバッファの個数が8個のため、FLXAnFRMRC レジスタの LCB ビット = "7" となります。この場合、データポインタは (FLXAnFRMRC レジスタの LCB ビット+1) * 4 = 32 以上である必要があります。

本例では、2個の Static バッファ、2個の Dynamic バッファ、4個の FIFO バッファを使用するため、FLXAnFRMRC レジスタの FDB ビット = "2"、FLXAnFRMRC レジスタの FFB ビット = "4"、FLXAnFRMRC レジスタの LCB ビット = "7"、FLXAnFRMRC レジスタの FFB ビット = "128" を設定します。

Static フレームのデータ長は4バイトであるため、FLXAnFRMHDC レジスタの SFDL ビット = "2" を設定します。

Dynamic バッファの Payload 長は6バイトであるため、メッセージバッファ 2,3 の FLXAnFRWRHS2 レジスタの PLC ビット = "3" を設定します。

FIFO バッファの Payload 長は8バイトであるため、メッセージバッファ 4~7 の FLXAnFRWRHS2 レジスタの PLC ビット = "4" を設定します。

各メッセージバッファのデータポインタは下記となります。

メッセージバッファ 0 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "1100"

メッセージバッファ 1 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "1101"

メッセージバッファ 2 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "1985"

メッセージバッファ 3 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "1987"

メッセージバッファ 4 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "79"

メッセージバッファ 5 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "81"

メッセージバッファ 6 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "83"

メッセージバッファ 7 のデータポインタは FLXAnFRWRHS3 レジスタの DP ビット = "85"

3. 通信コントローラの状態

3.1 通信コントローラ状態遷移図

各状態間の遷移は FLXAnFRSUCC1 レジスタ CMD ビットにコマンドを書き込むことで行います。

電源投入～ネットワーク通信開始までの流れを図 3-1 に示します。

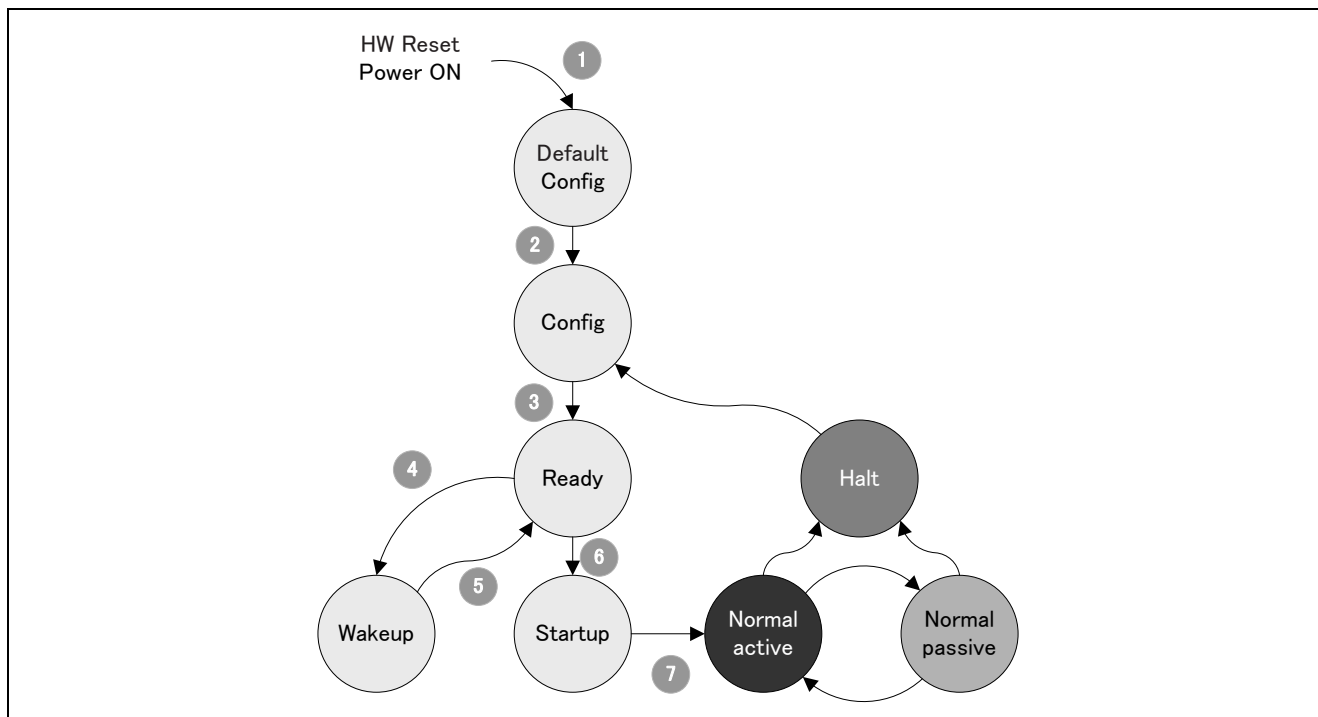


図 3-1 ネットワーク通信開始までの流れ

各状態間の遷移に必要な操作を表 3-1 に示します。

表 3-1 各状態間の遷移に必要な操作

遷移前の状態	遷移後の状態	必要な操作
①All States	Default_config	—
②Default Config	Config	FLXAnFRSUCC1 レジスタの CMD ビットに"0001" (Config) 書き込み
③Config	Ready	FLXAnFRLCK レジスタの CLK ビットのロック解除操作後、FLXAnFRSUCC1 レジスタの CMD ビットに"0010" (Ready) 書き込み
④Ready	Wakeup	FLXAnFRSUCC1 レジスタの CMD ビットに"0011" (Wakeup) 書き込み
⑤Wakeup	Ready	FLXAnFRSUCC1 レジスタの CMD ビットに"0010" (Ready) 書き込み、- (または Wakeup 完了、中断により自動的に遷移)
⑥Ready	Startup	FLXAnFRSUCC1 レジスタの CMD ビットに"0100" (Run) 書き込み
⑦Startup	Normal active	- (Startup 完了により、自動的に遷移)
Startup Normal active	Ready	FLXAnFRSUCC1 レジスタの CMD ビットに"0010" (Run) 書き込み

3.2 Default Config～Ready 状態

リセット後は、Default Config 状態です。

CHI CONFIG コマンド (FLXAnFRSUCC1 レジスタの CMD ビット) により Config 状態へ遷移した後に、各種設定レジスタへ初期設定を行ってください。CONFIG 状態から他の状態へ遷移するには、CMD ビットにコマンドを書き込む前に FLXAnFRLCK レジスタの CLK ビットに対してロック解除処理を行う必要があります。この機能により、ソフトウェア暴走などの意図しない動作による通信コントローラ誤操作を防止します。

操作手順例を図 3-2 に示します。

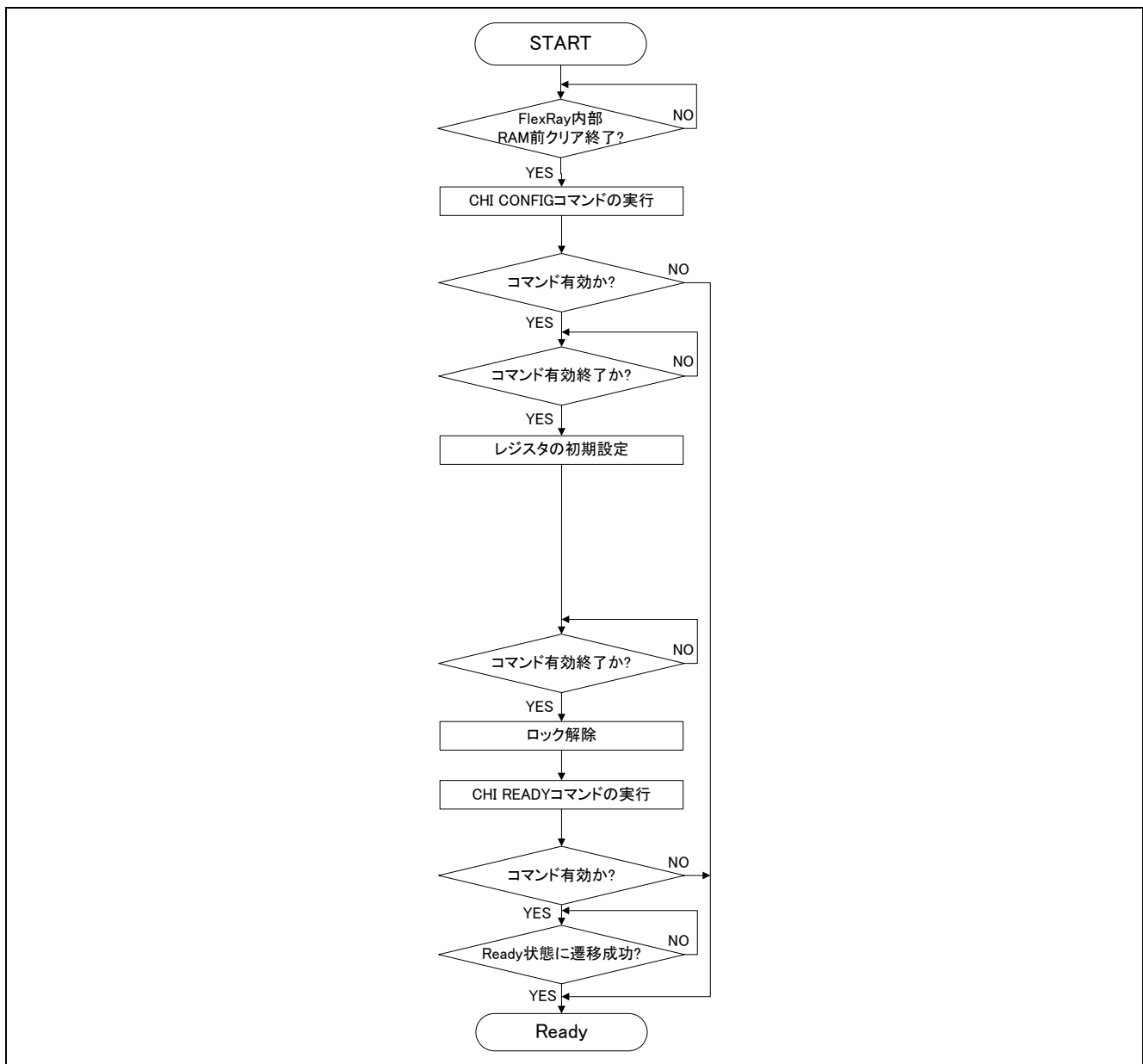


図 3-2 設定手順例

- 【注】
1. リセット後、Message RAM のクリア実行中は、各レジスタへの設定は行わないでください。FLXAnFRMHDS レジスタの CRAM ビット、FLXAnFRSUCC1 レジスタの PBSY ビットが“0”になったことを確認後、各レジスタへの設定を行ってください。
 2. Config 状態から他の状態へ遷移する場合、FLXAnFRLCK レジスタによりロックを解除する必要があります。ロックを解除する場合の FLXAnFRLCK レジスタへの書き込みは、連続して行ってください。2 回の書き込みの間に他の処理を実行した場合、再度最初の書き込みから実行する必要があります。

3.3 Wakeup 状態

FlexRay では、通信開始までにすべてのチャンネルがアクティブ状態である必要があります。

Wakeup 状態で、Wakeup マスタノードは、チャンネルに接続しているノードをアクティブ状態にします。

接続しているチャンネルがアクティブではなく、自ノードが Wakeup マスタノードである場合、CHI WAKEUP コマンド (FLXAnFRSUCC1 レジスタの CMD ビット) により Wakeup 状態に遷移し、Wakeup シーケンスを実行します。

片チャンネルのみアクティブであり、もう一方のチャンネルを Wakeup する場合は、Ready 状態から Config 状態に戻り、Wakeup パタンを送信するチャンネルを変更後、再度 Wakeup 状態へ遷移します。

Wakeup ステータスの変化は、FLXAnFRSIR レジスタの WST ビットと FLXAnFRCCSV レジスタの WSV0~WSV2 ビットによって確認できます。有効な Wakeup パタンを受信した場合は、FLXAnFRSIR レジスタの WUPA ビットと WUPB ビットがセットされます。

図 3-3 に Wakeup 時の状態遷移図を示します。

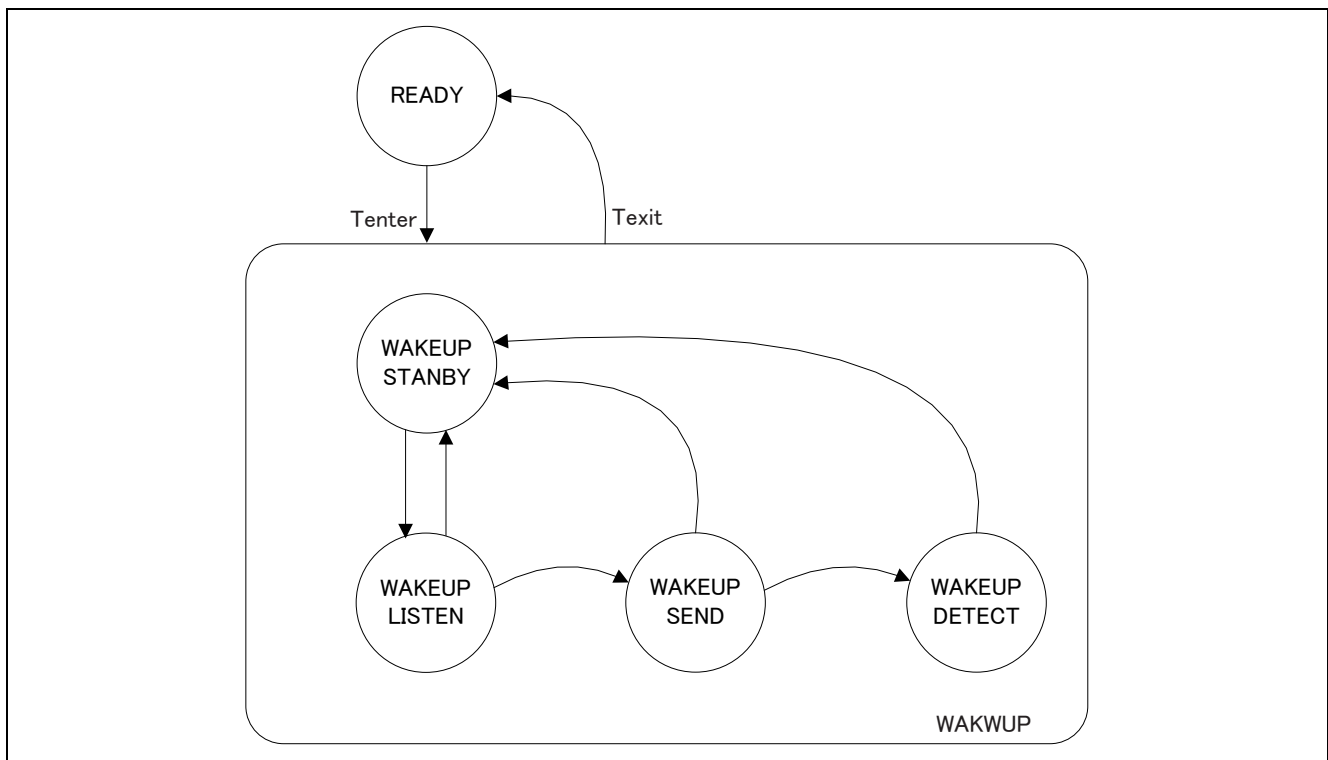


図 3-3 Wakeup 時の状態遷移図

- 【注】
1. Wakeup 状態中は、バスドライバへ Wakeup に必要な最低限の電源供給が必要です。
 2. FlexRay では、Wakeup 状態で、すべてのノードがアクティブ状態になったかどうかを確認できません。Cold_start ノードは、Startup 開始前に、全ノードが Wakeup をするために必要な時間を考慮し、ウェイトする必要があります。
 3. Wakeup チャンネルは、FLXAnFRSUCC1 レジスタの WUCS ビットにより選択してください。FlexRay では、1つのノードが同時に2つのチャンネルを Wakeup することは禁止されています。

図 3-4~図 3-7 に Wakeup マスタノード、Cold-start ノードとして動作する場合の Wakeup 手順例を示します。

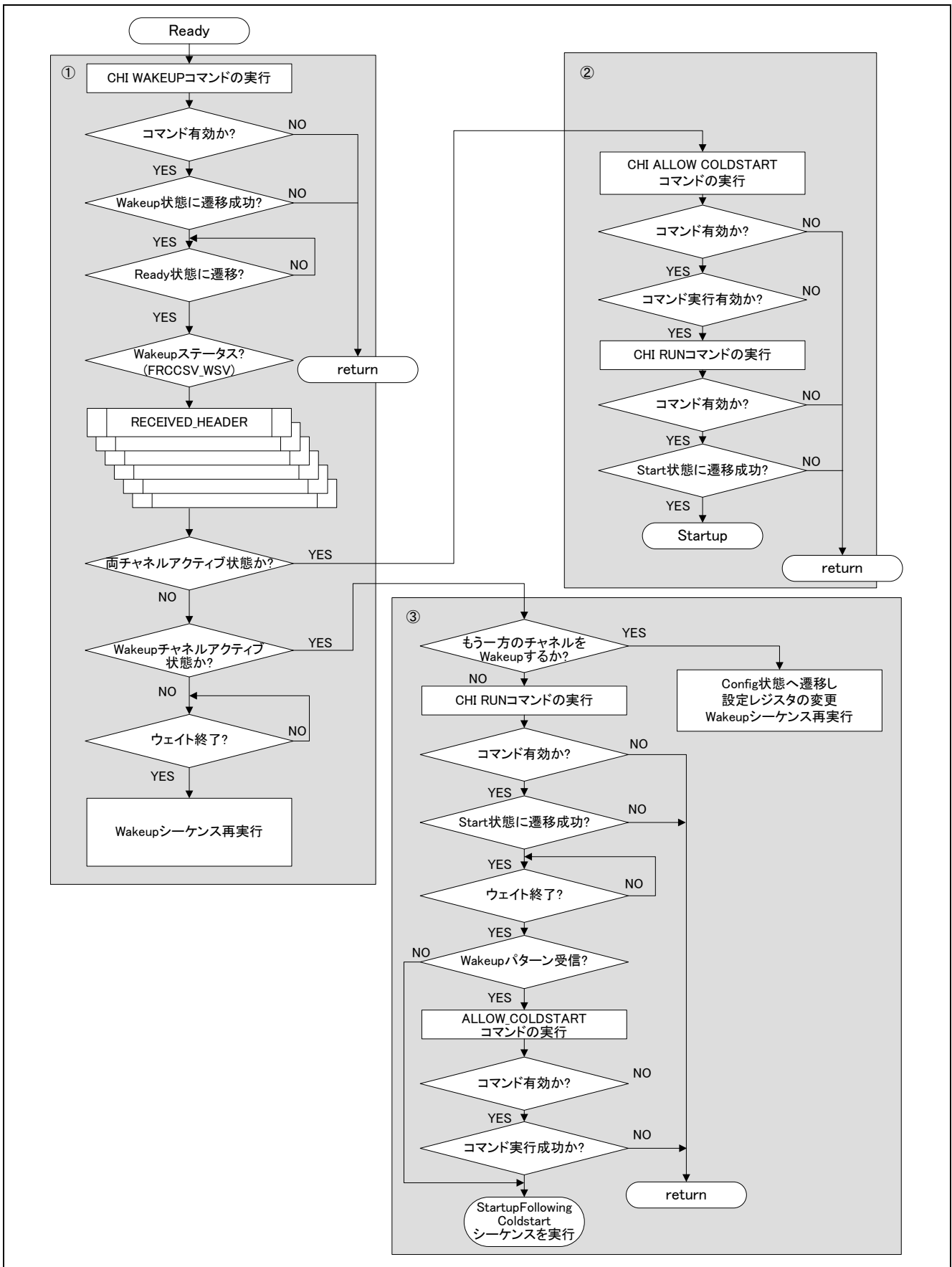


図 3-4 Wakeup シーケンス例 (1)

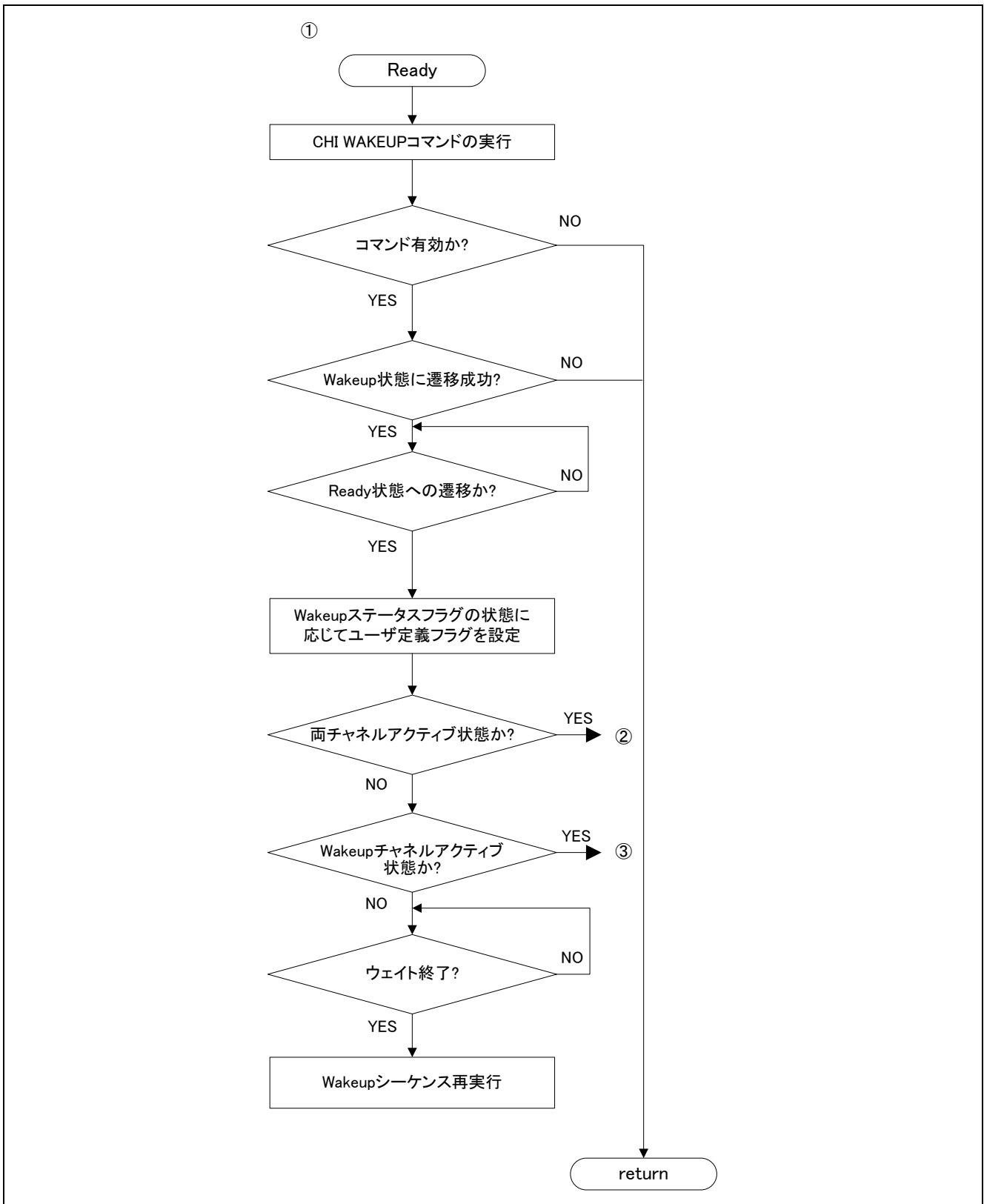


図 3-5 Wakeup シーケンス例 (2)

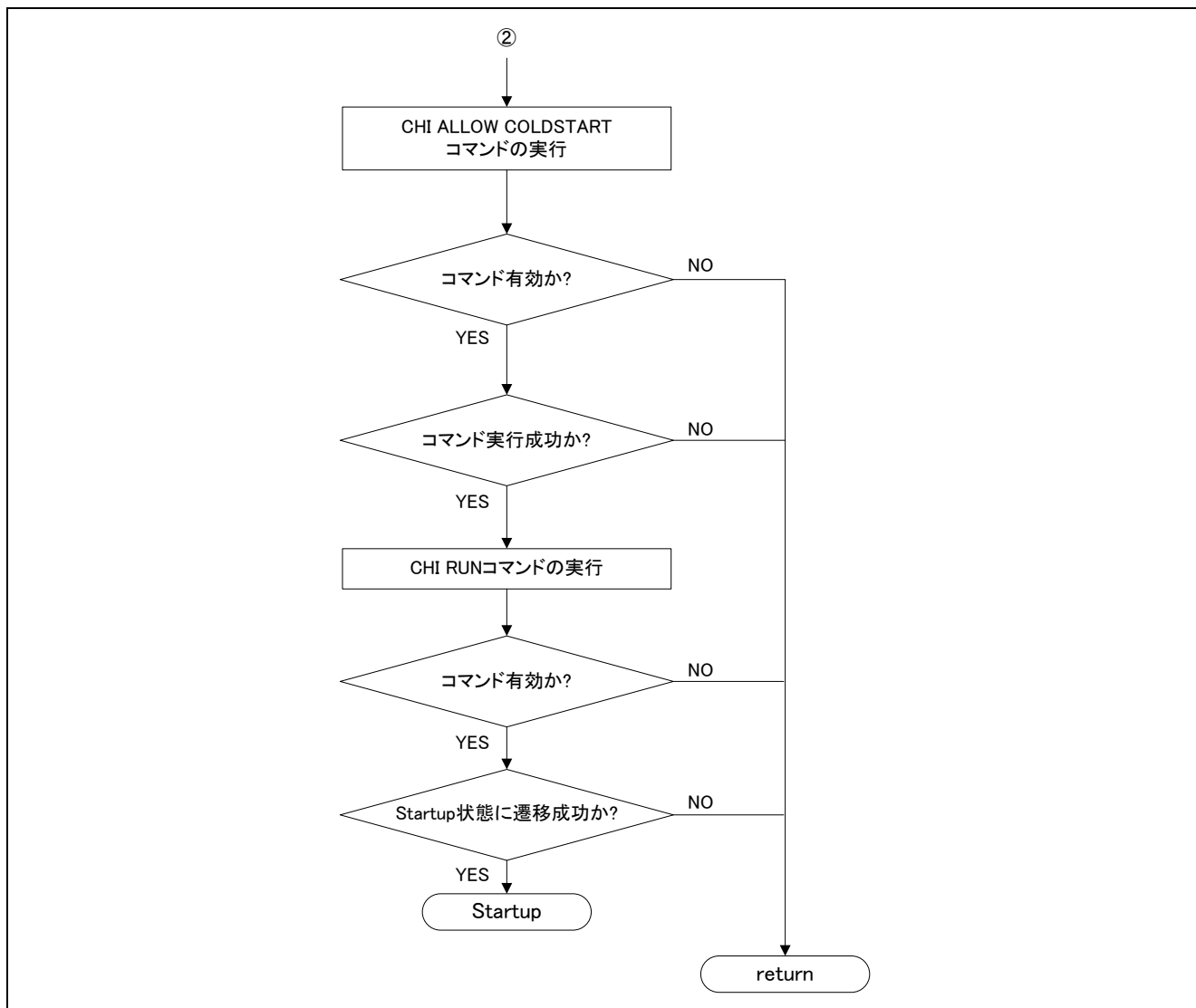


図 3-6 Wakeup シーケンス例 (3)

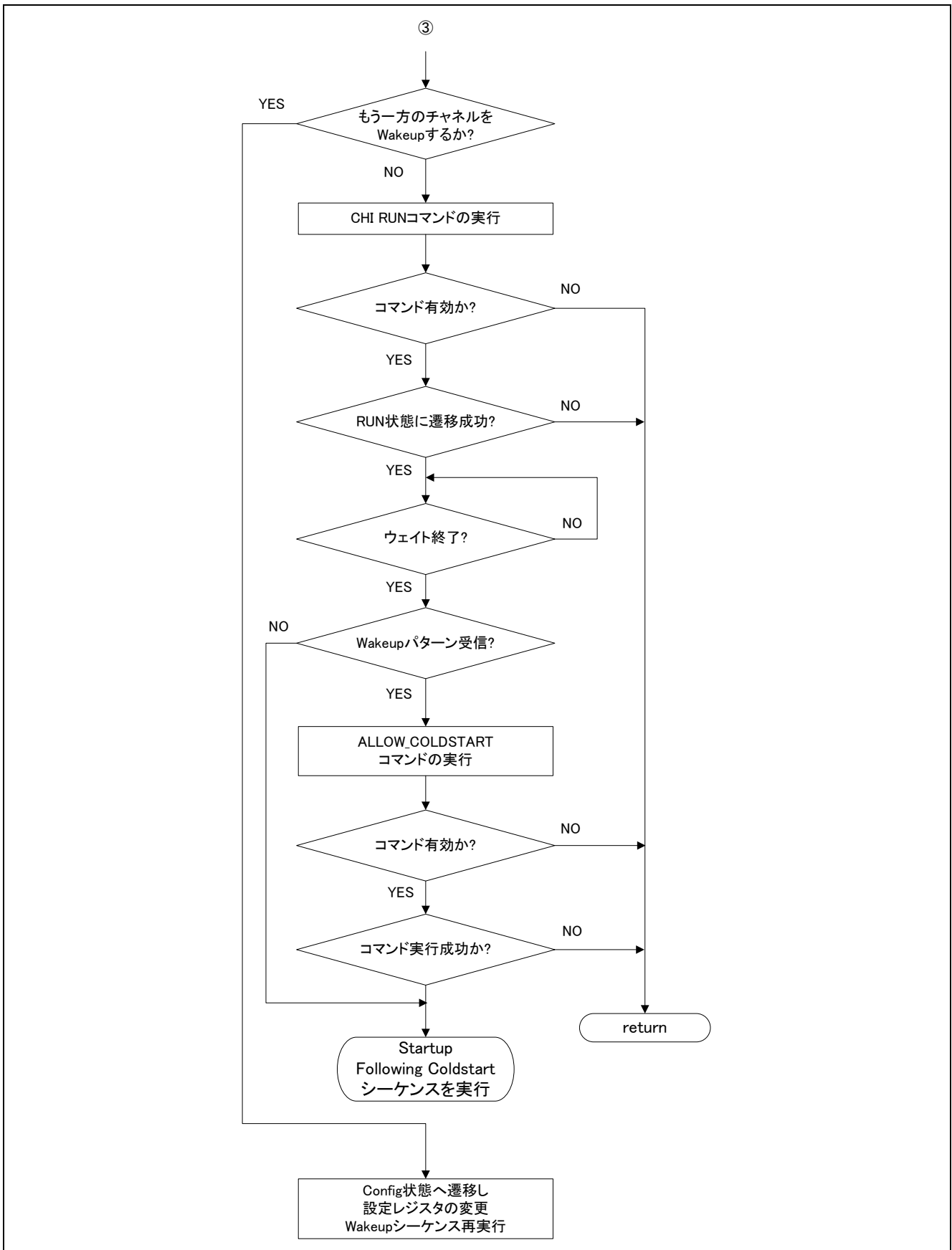


図 3-7 Wakeup シーケンス例 (4)

3.3.1 RECEIVED_HEADER または COLLISION_HEADER 時の処理例

RECEIVED_HEADER (FLXAnFRCCSV レジスタの WSV ビット = "001") または COLLISION_HEADER (FLXAnFRCCSV レジスタの WSV ビット = "011") 状態時、自ノードが Wakeup を実行しようとして試みているクラスタがすでに通信状態であるとして、Wakeup 処理をアボートします。

図 3-8 にて動作概略例を示します。

動作概略例では、自ノードが Leading Cold_start ノードとして、Startup 状態へ遷移します。

制御手順例については、図 3-4 を参照ください。両チャンネルアクティブなため、②の制御を行います。

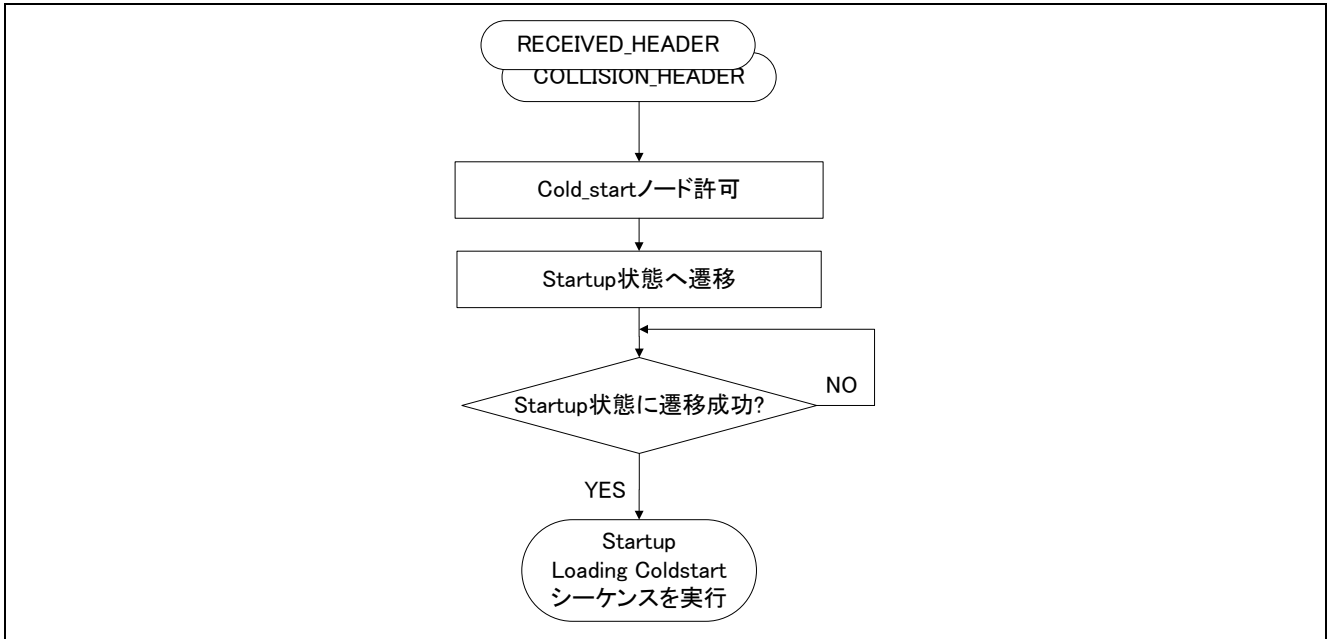


図 3-8 設定手順例

3.3.2 RECEIVED_WUP、COLLISION_WUP または TRANSMITTED 時の処理例

RECEIVED_WUP (FLXAnFRCCSV レジスタの WSV ビット = "010") または COLLISION_WUP (FLXAnFRCCSV レジスタの WSV ビット = "100") 状態時、自ノードが Wakeup を試みているチャンネルが、すでに他のノードによって Wakeup 中であるため、Wakeup 処理をアボートします。

TRANSMITTED (FLXAnFRCCSV レジスタの WSV ビット = "110") の場合、Wakeup パタンの送信を成功したため、Startup 状態へ遷移します。この場合、他のノードが Wakeup 処理を実行するための時間分ウェイトします。

片チャンネルのみアクティブであり、自ノードが、もう一方のチャンネルの Wakeup を実行しない場合、CHI RUN コマンド (FLXAnFRSUCC1 レジスタの CMD ビット) により Startup 状態に遷移します。この場合、他のノードがもう片チャンネルを Wakeup するのを待ちます。自身は Following Cold_start ノードとして、他の Cold_start ノードの Startup 開始を待ち、統合します。

図 3-9 に設定手順例を示します。

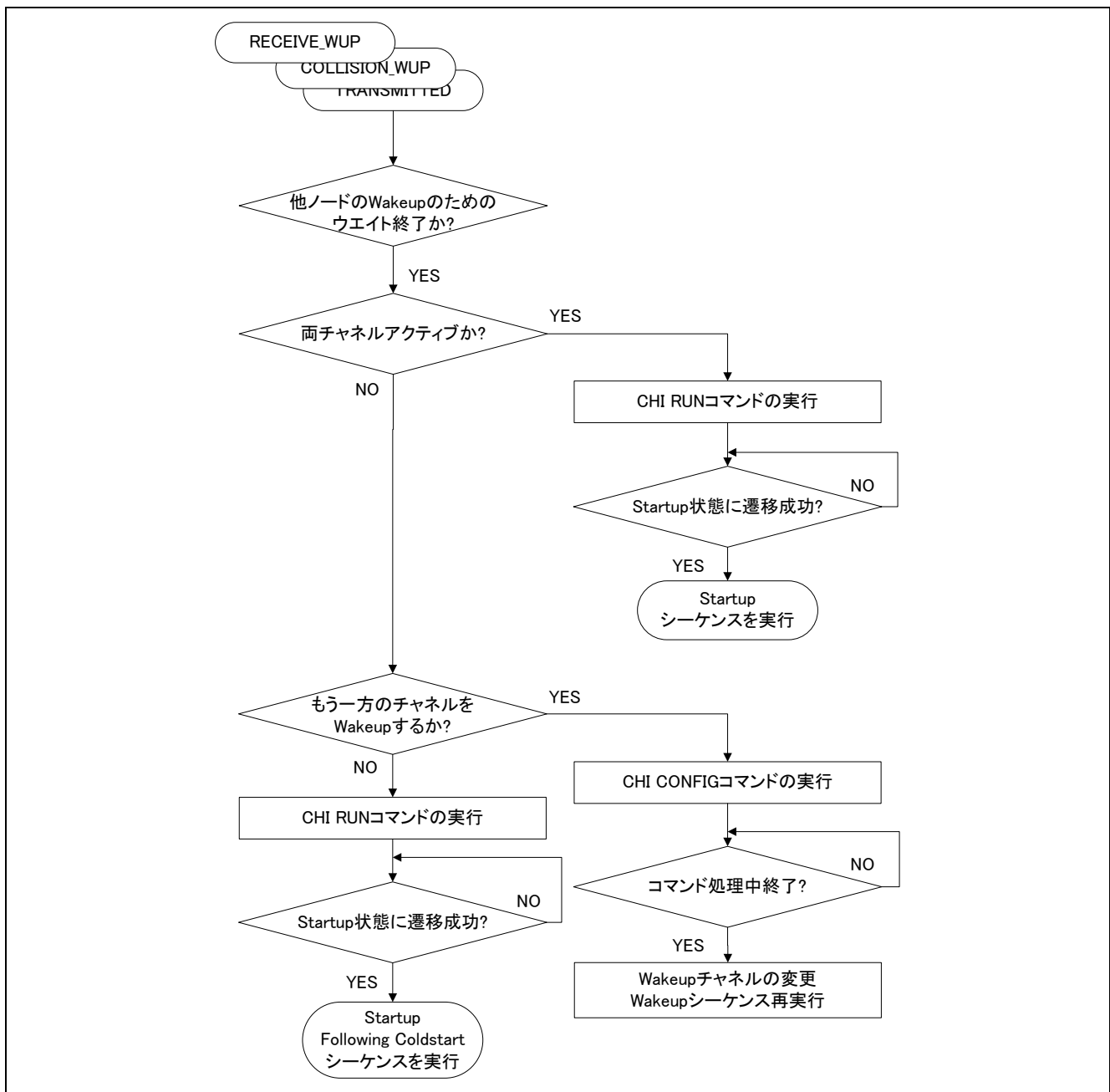


図 3-9 設定手順例

3.3.3 COLLISION_UNKNOWN 時の処理例

COLLISION_UNKNOWN (FLXAnFRCCSV レジスタの WSV ビット = “101”) 状態時、有効な Wakeup パタンや有効なフレームのヘッダ部を受信することなく内部の wakeup タイマが規定値に達した後、WAKEUP_DETECT 状態を抜けることで Wakeup 処理を停止します。

通信コントローラがこのような状態にある場合、ノイズの影響が大きいか、または通信に異常があると推測できます。

Wakeup 処理を再度実行する場合は、他ノードの内部処理のために、一定時間ウェイトします。

【注】 Listen Timeout Noise タイマ値は FLXAnFRSUCC2 レジスタの LTN0~LTN3 で設定できます。

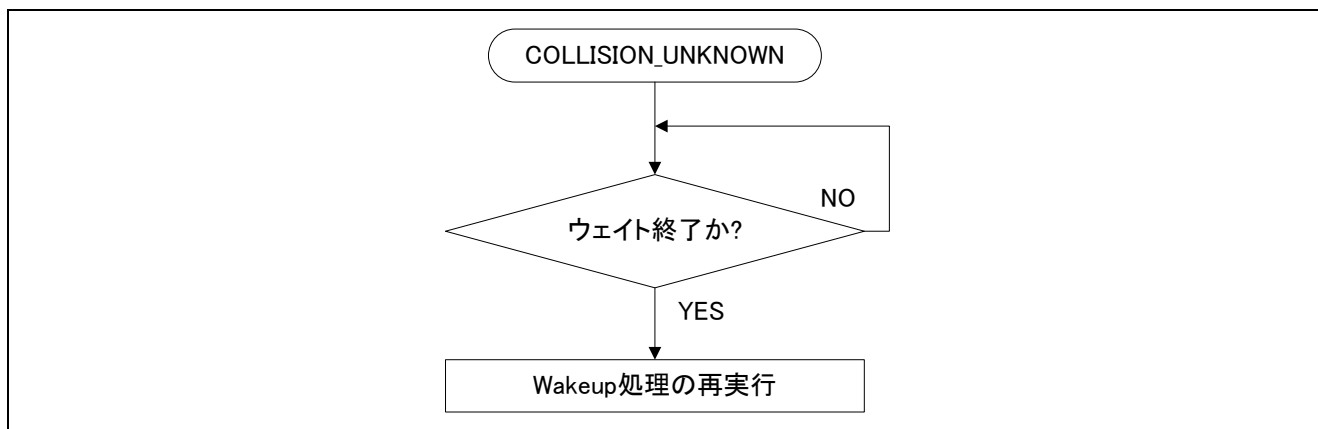


図 3-10 制御手順例

3.4 Startup 状態

FlexRay では、通信を開始する前に、クラスタ全体のクロックを同期させる必要があります。各ノードは Startup 処理を行い、自ノードのクロックを Cold_start ノードのクロックに統合します。

クラスタを Startup するために少なくとも 2 つ以上の Cold_start ノードが必要です。

Startup ノード、Sync ノードとして動作するための設定については、『1. 初期設定』を参照してください。

図 3-11 に Startup 中の状態遷移図を示します。

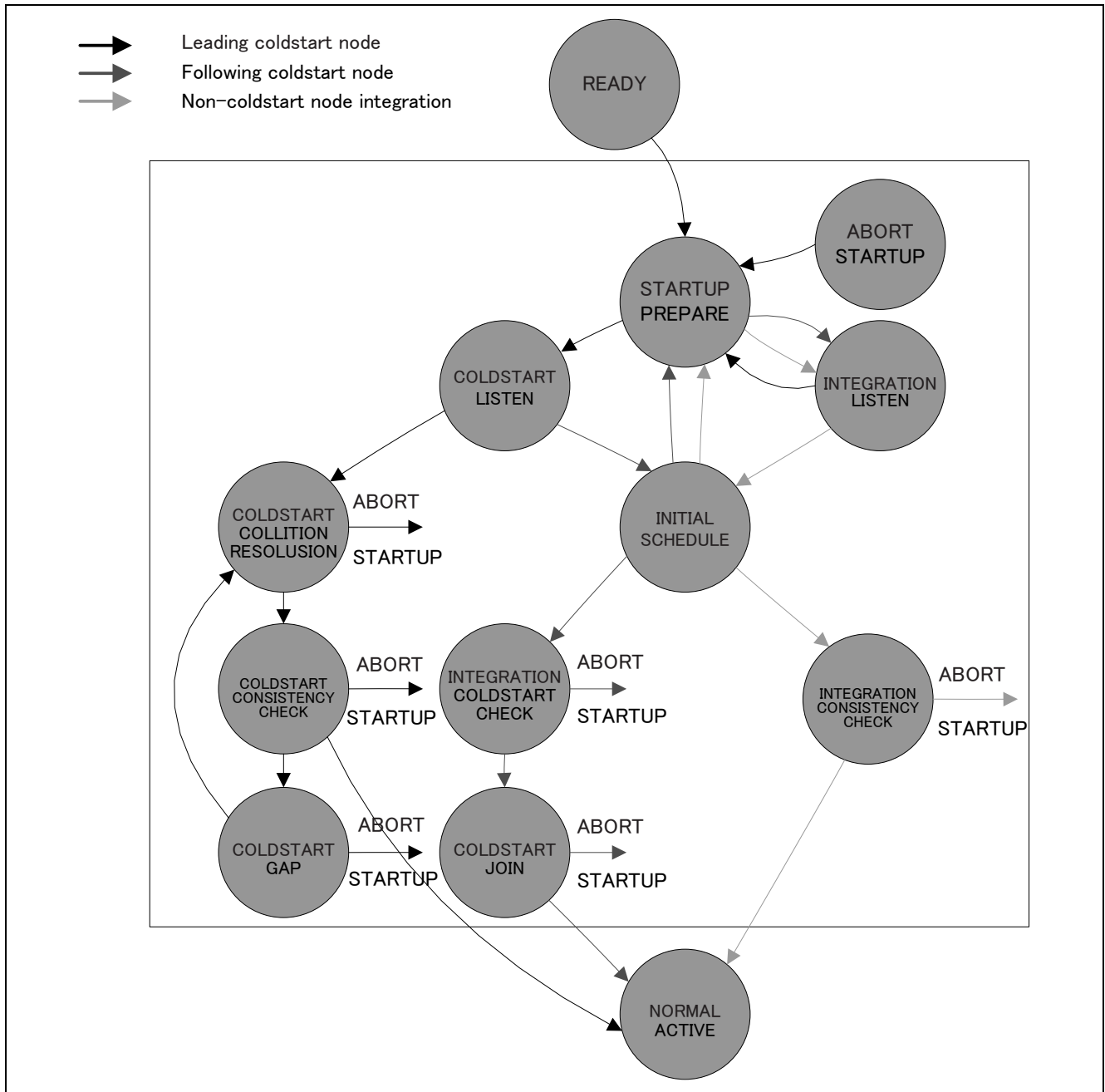


図 3-11 Startup 中の状態遷移

3.4.1 Cold_start ノードの動作

Ready 状態 (FLXAnFRCCSV レジスタの POCS ビット = "000001") で、CHI RUN コマンド (FLXAnFRSUCC1 レジスタの CMD ビット = "0100") を実行することで、Startup 状態に遷移します。

Cold_start の試行回数は、FLXAnFRSUCC1 レジスタの CSA0~CSA4 ビットで設定します。Cold_start ノードの Cold_start 試行回数は 2 以上とし、クラスタ内の全 Cold_start ノードで同じにしてください。

Startup ノードは FLXAnFRCCSV レジスタの CSI ビットが "0" のとき、Cold_start ノードとして Startup を開始することができます。このビットは、CHI ALLOW_COLDSTART (FLXAnFRSUCC1 レジスタの CMD ビット = "1001") を実行することで "0" になります。

しかし、継続している通信がある場合、Cold_start ノードが Startup 状態へ遷移すると同時に Startup を開始すると、通信を妨害する可能性があります。これを回避するために、Cold_start ノードは、CSI ビットを "1" にした状態で (Startup 禁止)、Startup 状態に遷移しなければなりません。

この場合、Cold_start ノードは、CHI RUN コマンド実行後 (Startup 状態へ遷移)、INTEGRATION_LISTEN 状態へ入ります。

通信状態の検出がなければ、CHI ALLOW_COLDSTART コマンド (FLXAnFRSUCC1 レジスタの CMD ビット = "1001") を実行し、CSI ビットをクリアします。これにより、COLDSTART_LISTEN 状態になります。

【注】 Cold_start が失敗した場合、ステータスレジスタとエラーレジスタを読み出してください。Startup を再実行する場合は、実行中の通信を妨害しないように、FLXAnFRCCSV レジスタの CSI ビットにより Cold_start を禁止しておくなどの考慮をしてください。

Cold_start ノードの制御手順例を図 3-12 に示します。

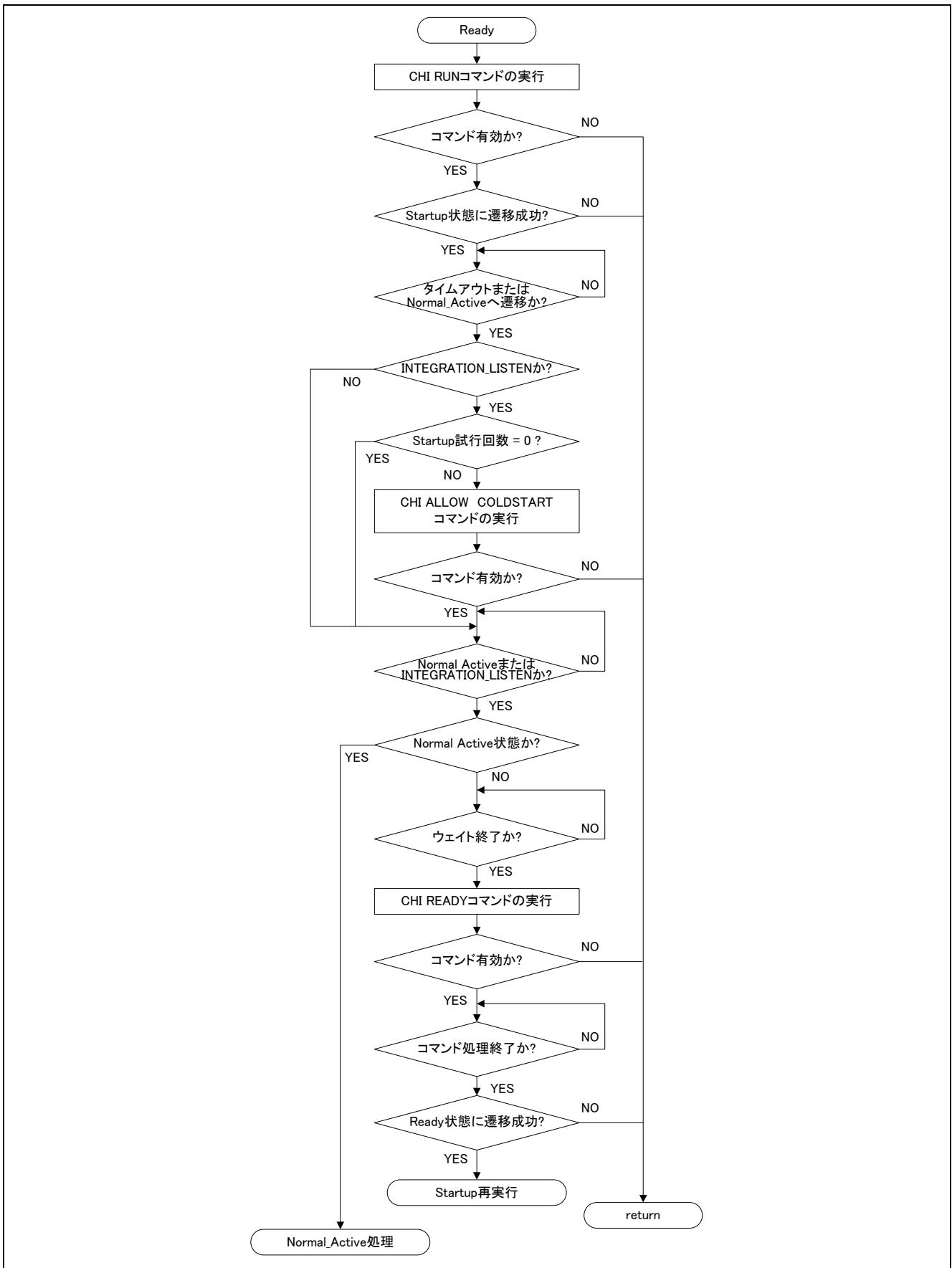


図 3-12 Cold_start ノードの制御手順例

■Leading Cold_start ノードのパス

通信が検出されない場合、ノードは COLDSTART_COLLISION_RESOLUTION 状態に遷移し、CAS シンボルを送信します。CAS シンボルの送信後、サイクル 0 が開始します。

複数の Cold_start ノードが同時に Startup を開始する場合がありますが、FlexRay では、最初の 4 サイクルでこの状況は回避されます。

Cold_start ノードは最初の 4 サイクル（サイクル 0～サイクル 3）の間に CAS シンボルまたはフレームヘッダを受信した場合、すぐに COLDSTART_LISTEN 状態に戻ります。このため、1 つのノードのみが Leading Cold_start ノードシーケンスに残ることになります。

サイクル 4 で Leading Cold_start ノード以外の Cold_start ノードがフレームの送信を開始します。

Leading Cold_start ノードは、COLDSTART_COLLISION_RESOLUTION の後、4 サイクルで COLDSTART_CONSISTENCY_CHECK 状態に遷移します。サイクル 4 およびサイクル 5 で、他の Cold_start ノードからの Startup フレームを受信することで、自ノードのクロックを補正します。

クロック補正に成功し、少なくとも自ノード以外の 1 つの Cold_start ノードから 2 サイクル Startup フレームの受信を成功した場合、Normal_Active 状態に遷移します。

Cold_start ノードが実行できる Cold Startup のリトライ回数は FLXAnFRSUCC1 レジスタの CSA0～CSA4 で設定します。リトライ回数の残数は FLXAnFRCCSV レジスタの RCA0～RCA4 で参照できます。RCA0～RCA4 の値は、1 回リトライするたびに 1 ずつ減少します。

この値が"0"より大きい場合にのみ COLDSTART_COLLISION_RESOLUTION 状態に、"1"より大きい場合にのみ、COLDSTART_LISTEN 状態に遷移可能です。リトライ回数を"1"と設定した場合、自ノードが Cold Start を開始することは禁止ですが、他ノードへの統合は可能です。

【注】 CSA0～CSA4 ビットへの設定値は"2"以上でかつクラスタで同じ値を設定してください。

■Following Cold_start ノードのパス

Following Cold_start ノードは COLDSTART_LISTEN 状態に遷移し、Leading Cold_start ノードから有効な Startup フレームを 2 サイクル受信し、スケジュール情報とクロック補正情報を取得します。

まず、有効な Startup フレームを 1 つ受信するとすぐに INITIALIZE_SCHEDULE 状態に遷移します。

2 つめの有効な Startup フレームの受信に成功し、クロック同期、スケジュール情報の取得に成功すると INTEGRATION_COLDSTART_CHECK 状態に遷移します。

INTEGRATION_COLDSTART_CHECK 状態の 2 サイクル間で、他ノードからのすべての Sync フレームを受信し、クロック補正を行います。エラーなくクロック補正を実行し、Leading Cold_start ノードが有効な状態であることを確認した後、COLDSTART_JOIN 状態に遷移します。

COLDSTART_JOIN 状態で、Startup フレームの送信を開始します。

この状態で、Leading Cold_start ノードと Following Cold_start ノードは相互にスケジュールをチェックします。クロック補正エラーが検出された場合、Startup 処理はアボードします。

すべての偶数サイクルで少なくとも 1 つの Startup フレームを受信し、すべての連続する 2 サイクルで少なくとも 1 ペアの Startup フレームを受信した場合、ノードは Normal_active 状態に遷移します。

その結果、Following Cold_start ノードは、Leading Cold_start ノードの少なくとも 1 サイクル後に Startup 状態を抜けます。

3.4.2 Non-Cold_start ノードの動作

Cold_start ノード以外のノードは、Non-Cold_start ノードとして Cold_start ノードのクロックに統合します。

Ready 状態（FLXAnFRCCSV レジスタの POCS ビット = “000001”）で、CHI RUN コマンド（FLXAnFRSUCC1 レジスタの CMD ビット = “0100”）を実行すると Startup 状態になります。

図 3-13 に Non-Cold_start ノードの制御手順例を示します。

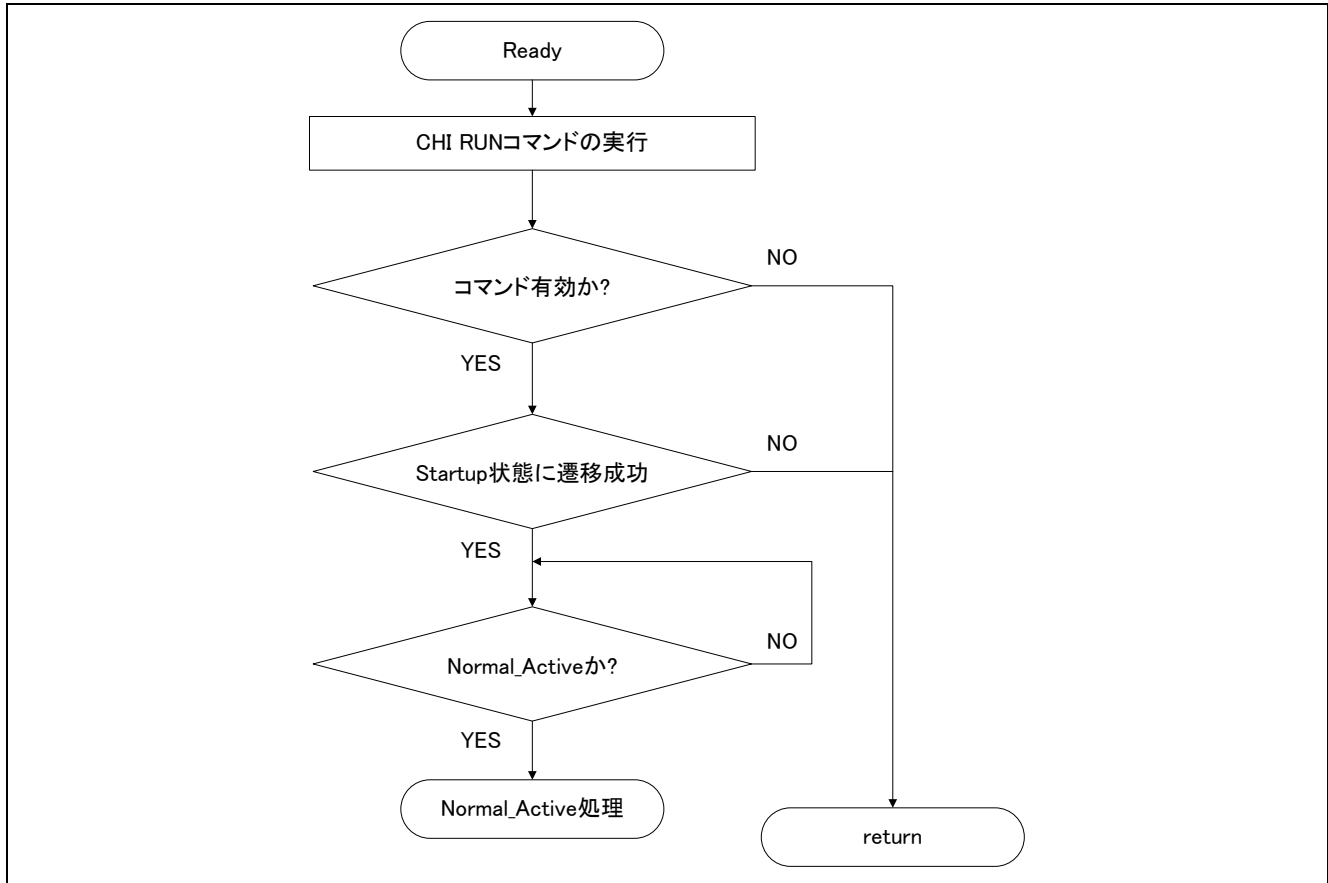


図 3-13 Non-Cold_start ノードの操作手順

Non-Cold_start ノードは、CHI RUN コマンド（FLXAnFRSUCC1 レジスタの CMD ビット = “0100”）実行後、INTEGRATION_LISTEN 状態に遷移して接続チャネルの状態を一定時間確認します。

有効な Startup フレームを受信すると、INITIALIZE_SCHEDULE 状態に遷移します。2 つめの有効な Startup フレームへのクロック同期に成功し、スケジュール情報を取得すると INTEGRATION_CONSISTENCY_CHECK 状態に遷移します。

INTEGRATION_CONSISTENCY_CHECK 状態で、ノードはクロック同期が正確に実行でき、（少なくとも 2 つの）Cold_start ノードが、自己と同じスケジュール情報を含んだ Startup フレームを送信するのを確認します。もし、エラーが検出された場合、アボートします。

Non-Cold_start ノードは、この状態の偶数サイクルで、2 つの有効な Startup フレームまたは、このノードが統合したノードの Startup フレームを受信しなければなりません。受信に失敗した場合、ノードは統合を中止し、アボートします。

また、最初の 2 サイクルで 2 つの有効な Startup フレームのペアまたは、このノードが統合したノードの Startup フレームのペアを受信しなければなりません。受信に失敗した場合、ノードは統合を中止し、アボートします。

最初の 2 サイクル後の、偶数サイクル内に 2 つ以上の Startup フレームを受信できなかった、または 2 サイクル内に 2 つの有効な Startup フレームペアを受信できなかった場合、ノードは統合を中止し、アポートします。

ノードはこの状態で、連続した 2 つの偶数・奇数のサイクルペアで 2 つの有効な Startup フレームを受信し、Normal_active 状態へ遷移します。

その結果、ノードは Leading Cold_start ノードの少なくとも 2 サイクル後の奇数サイクルの終わりに Startup 状態を抜けることになります。

4. MessageRAM へのデータ設定と MessageRAM からのデータ読み出し

4.1 送信フレームデータの構成

FLXAnFRWRHS1~FLXAnFRWRHS3 レジスタでメッセージバッファのヘッダ部（送受信フィルタリング条件）を設定します。

FLXAnFRWRDS1~FLXAnFRWRDS64 レジスタにてデータ部を設定します。

FLXAnFRWRHS1~FLXAnFRWRHS3 レジスタおよび FLXAnFRWRDS1~FLXAnFRWRDS64 レジスタで設定したデータは、IBF（入力バッファ）を経由して、MessageRAM（メッセージバッファ）へ転送します。

ヘッダ部分の設定例を表 4-1、制御手順例を図 4-1 に示します。

表 4-1 ヘッダ部分の設定例

名称	設定値	備考
フレーム ID	1	
サイクルフィルタリング	毎コミュニケーションサイクル	
送受信チャンネル	Ch. A/B 両方	
送受信設定	送信	
ネットワーク管理フラグ	使用しない	
送信モード	連続送信モード	
送受信完了割り込み	使用しない	
Header CRC	0xFFFF	
データ長	16 バイト	
データポインタ	データ部先頭アドレス H'400	

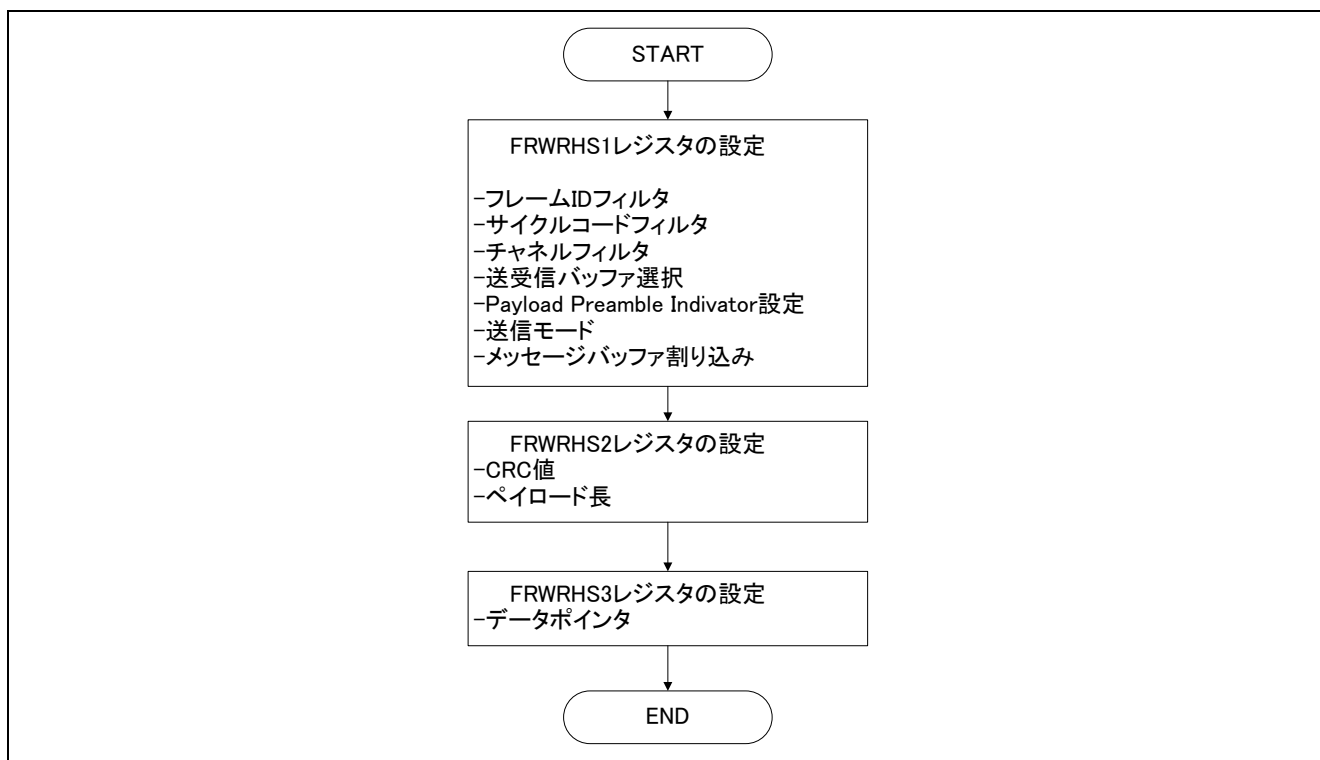


図 4-1 制御手順例

■Header CRC

FlexRay では、ヘッダ部でのデータ化けを検出するために HeaderCRC を導入しています。HeaderCRC 値はヘッダ部の SyncFrameIndicator・StartupFrameIndicator・FrameID・PayloadLength の値から計算する必要があります。

【注】 FlexRay 仕様書の規定により、通信コントローラはフレーム送信時に Header CRC の計算を行いません。Header CRC は、コンフィグレーションツール・Host MCU などにより、あらかじめ計算してヘッダ部に設定しておく必要があります。

4.2 IBF（インプットバッファ）：MessageRAM への転送

FlexRay モジュールでは、送信データの一貫性を保証するために、Message RAM への書き込みは IBF 経由で行います。また、データ転送パフォーマンス向上のために IBF 部分にホストサイド（CPU 側）／シャドウサイド（Message RAM 側）の 2 面構成を導入しています。

4.2.1 転送方法

(1) データ部およびヘッダ部を IBF（ホストサイド）に設定します。

（FLXAnFRWRDS1～FLXAnFRWRDS64 レジスタ、FLXAnFRWRHS1～FLXAnFRWRHS3 レジスタ、FLXAnFRIBCM レジスタ）

(2) FLXAnFRIBCR レジスタの IBRH ビットに転送先のバッファ番号を設定。

(3) (2)の書き込みが転送要求となり、IBF ホストと IBF シャドウが切り替わり、シャドウサイドの IBF と MessageRAM の転送が開始します。

このとき、IBSYS ビットが"1"となり、ビジー状態であることが示されます。

(4) 次に転送するデータがある場合、再度、データ部およびヘッダ部を IBF（ホストサイド）に設定します。

このとき、IBSYS ビットが"1"の場合、IBSYH ビットが"1"となります。

(5) 転送が完了すると、IBF ホストと IBF シャドウが切り替わり、転送待ちとなっていた次の転送が開始します。

このとき、IBSYS ビットは"1"を保持し、IBSYH ビットが"0"となります。

【注】 IBSYS ビット、IBSYH ビットがともに"1"の状態、IBRH ビットに次の転送要求のための設定を実行した場合、エラーとなり、FLXAnFREIR レジスタの IIBA ビットが"1"となります。この場合、IBF の値は変わりません。

入力バッファのダブルバッファ構造を図 4-2 に示します。

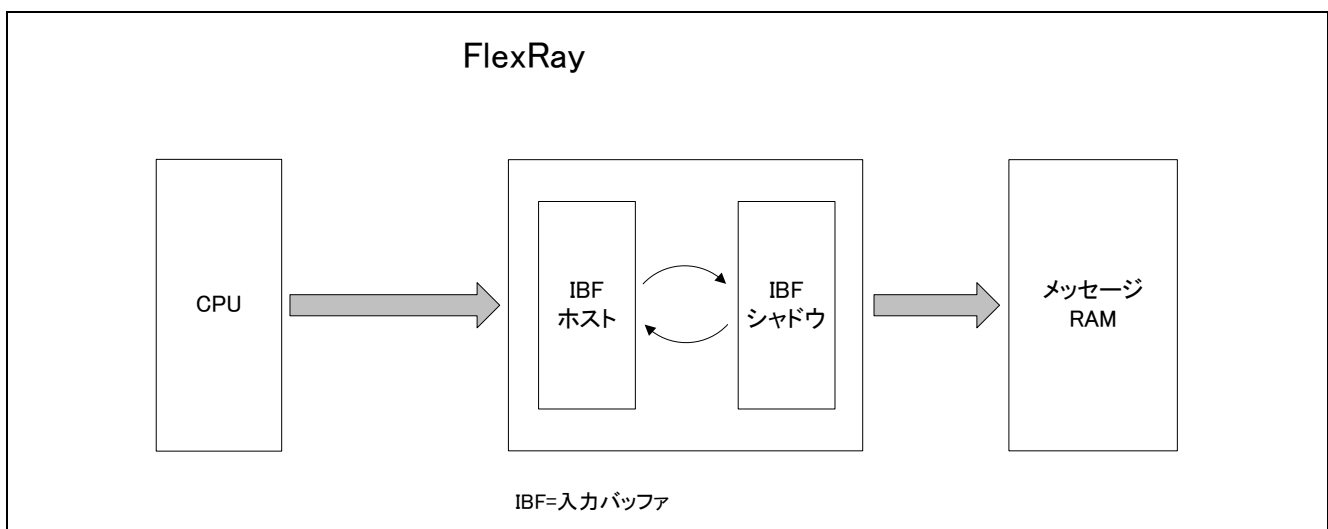


図 4-2 入力バッファのダブルバッファ構造

送信フレームのデータを Message RAM のメッセージバッファ 1 に転送する場合の設定例を表 4-2、制御手順例を図 4-3 に示します。

表 4-2 送信フレームデータの設定例

名称	設定値	備考
ヘッダ部	転送する	
データ部	転送する	
送信要求	送信要求 ON	
転送先バッファ番号	1	バッファ番号は"0"起算なので、2番目のバッファに転送

【注】 1. FLXAnFRMRC レジスタの SEC0~SEC1 ビットによりロックされたバッファに対しては、ヘッダ部は転送しないでください。

2. FIFO バッファに対しては、受信専用です。データ部は転送しないでください。

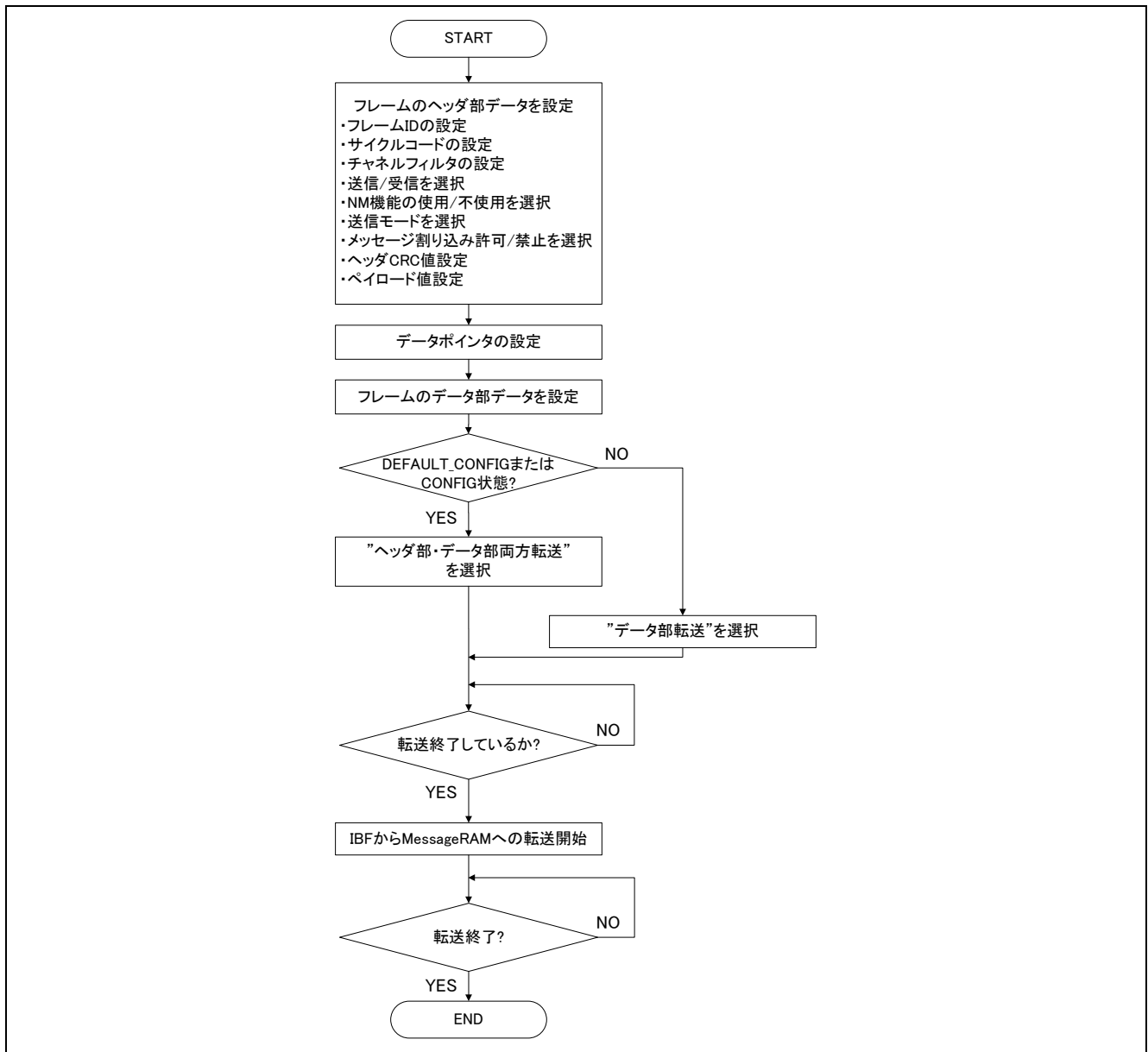


図 4-3 制御手順例

4.3 OBF（アウトプットバッファ）：MessageRAM からの読み出し

FlexRay モジュールでは、受信データの一貫性を保証するために、Message RAM からの読み込みは OBF 経由で行います。ホストサイド（CPU 側）／シャドウサイド（Message RAM 側）の 2 面構成を導入しています。

MessageRAM から読み出されたメッセージバッファのヘッダ部は FLXAnFRRDHS1～FLXAnFRRDHS3 レジスタ、データ部は FLXAnFRRDDS1～FLXAnFRRDDS64 レジスタに格納されます。

4.3.1 転送方法

- (1) FLXAnFROBCR レジスタの OBRS ビットに MessageRAM のバッファ番号を設定します。
- (2) OBSYS ビットが"0"であることを確認後、REQ ビットを"1"に設定します。
この設定により、メッセージバッファから OBF への転送が開始します。
- (3) 転送が完了すると、OBSYS ビットは自動的に"0"になります。
- (4) OBSYS ビットが"0"であることを確認後、VIEW ビットに"1"を設定します。
この設定により、OBF バッファの OBF ホストと OBF シャドウが切り替わります。
- (5) OBF バッファからデータを読み出します。

- 【注】
1. OBSYS ビットが"1"の状態、REQ ビットを"1"にした場合、エラーが発生し、FLXAnFREIR レジスタの IOBA ビットが"1"になります。
 2. FIFO バッファからデータを転送する場合は、FIFO バッファの先頭番号を OBRS ビットに設定してください。

出力バッファのダブルバッファ構造を図 4-4 に示します。

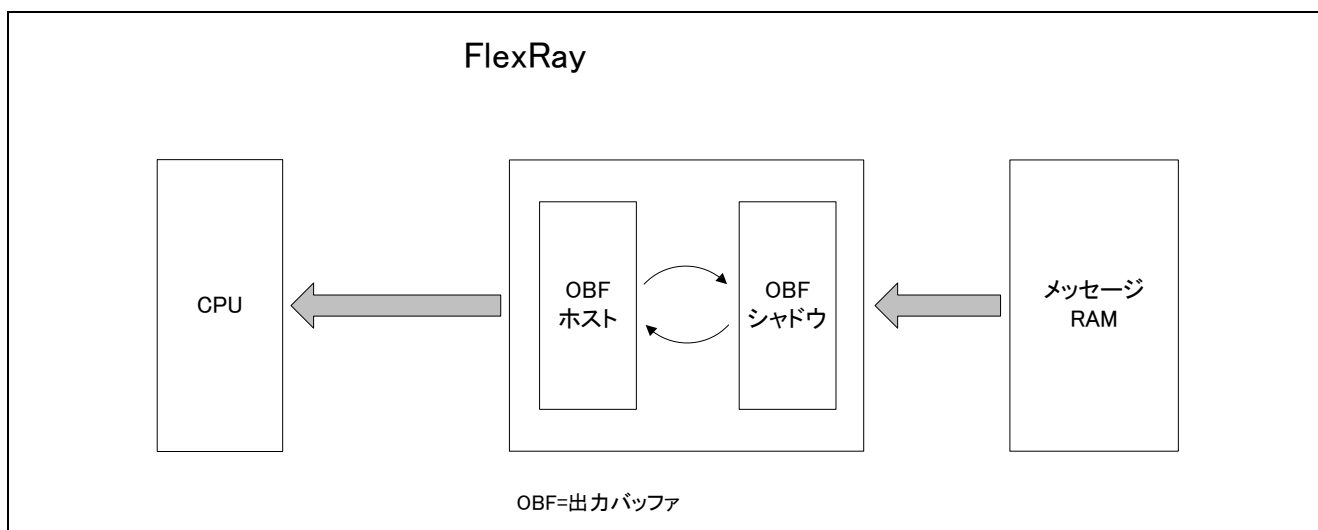


図 4-4 出力バッファのダブルバッファ構造

Message RAM のバッファ番号 1 で受信したフレームを OBF に読み出す場合の設定例を表 4-3、制御手順例を図 4-5 に示します。

表 4-3 受信フレーム読み出し時の設定例

名称	設定値	備考
ヘッダ部	転送する	
データ部	転送する	
転送元バッファ番号	1	

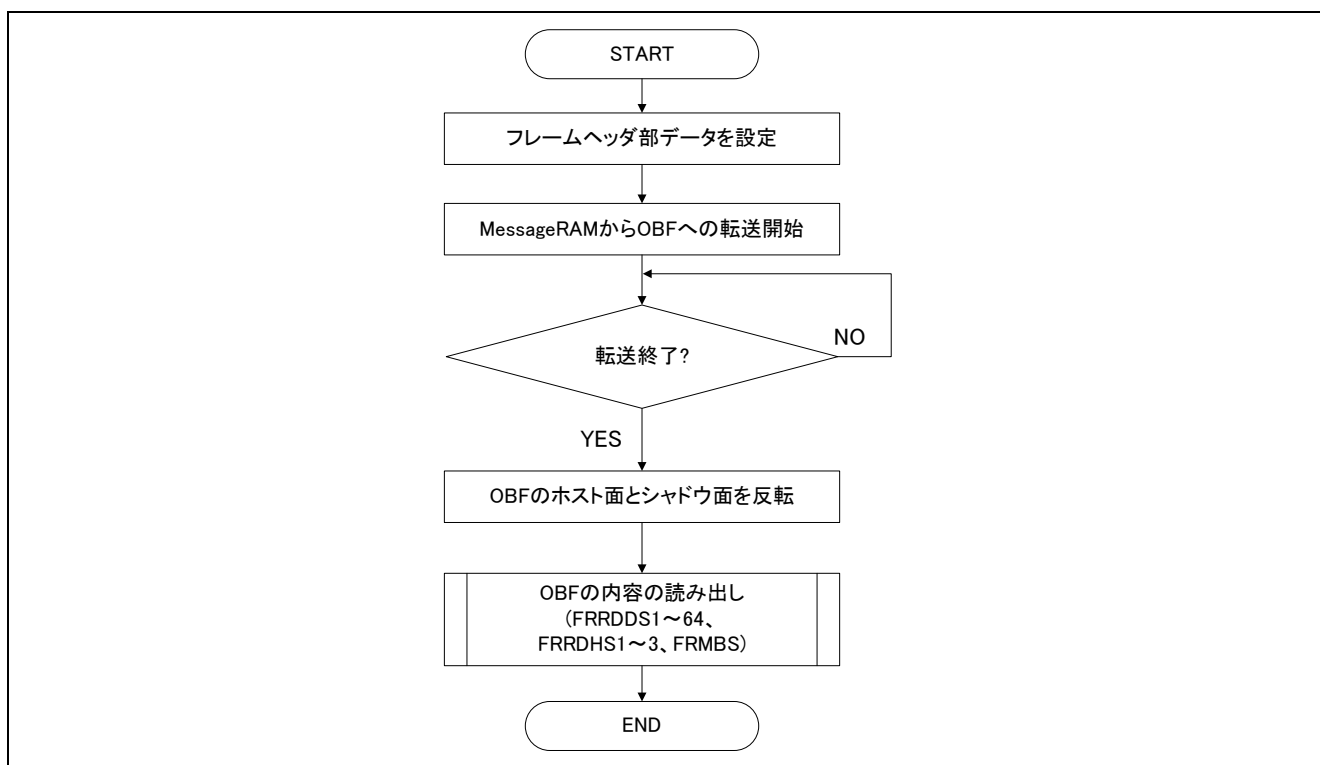


図 4-5 制御手順例

5. フレーム送受信

5.1 フレーム送信

FlexRay では、TDMA 方式に基づいて通信を行います。このため、フレーム送信タイミング詳細についてユーザが設定する必要はありません。あらかじめ設定した通信スケジュールに従って、通信コントローラがフレーム送信タイミングを制御します。

送信データ準備～フレーム送信 の流れを以下に示します。

- (1) 送信したいフレームを IBF に設定する。（このとき FLXAnFRWRHS1 レジスタの CFG ビット = "1" に設定する）
- (2) IBF⇒Message RAM に送信フレームデータを転送する。
ヘッダ部の STXRH ビットが"1"の場合、該当するメッセージバッファの送信要求フラグ（FLXAnFRTXRQ1～4 レジスタの対応するフラグ）が"1"になり、送信待ち状態になる。
- (3) 該当スロットの ActionPoint より、フレーム送信が開始される。
- (4) 設定時に送信完了割り込みを許可（FLXAnFRWRHS1 レジスタの MBI ビット = "1"）に設定している場合には、フレーム送信完了時に割り込み要求（FLXAnFRSIR レジスタの TXI ビット = "1"）が発生する。

このとき、送信成功したメッセージバッファ番号が FLXAnFRMHDS レジスタの MBT ビットに格納される。

【注】 送信フレームのデータは、フレームを送信する Slot 開始までに IBF バッファから Message RAM への転送を開始しておく必要があります。（例えば、スロット'5'でフレームを送信するには、スロット'5'開始までに IBF バッファから Message RAM への転送を開始しておく必要があります。）

■連続送信モードとシングルショットモード

FLXAnFRWRHS1 レジスタの TXM ビットで送信モードを選択します。

シングルショットモードを選択した場合、TXR フラグは送信完了時に"0"になります。

連続送信モードを選択した場合、TXR フラグは送信完了時に"0"になりません。

■パディング機能

Static セグメントでは、FLXAnFRMHDC レジスタの SFDL ビットに設定した値より、メッセージバッファのヘッダ部に設定したペイロード長が短い場合、パディングデータが付加されます。パディング値は"0"です。

5.2 フレーム受信

FlexRay では、TDMA 方式に基づいて通信を行います。このため通信コントローラは「受信フレームのフォーマットは正常か」だけでなく、「フレームを受信したタイミングは適切か」「フレーム内の各種フラグは正常か」などについても確認を行います。

バス上でのデータ検出～受信フレーム格納までの通信コントローラ内部動作の流れを以下に示します。

- (1) フレームフォーマットが正常か確認する。
(CRC エラーなどを検出した場合には、Syntax Error とする)
- (2) ヘッダ内容が正常か確認する。
(サイクルカウンタ値 異常などを検出した場合には、Content Error とする)
- (3) 受信バッファへの格納条件を満たすかどうか確認する。
- (4) 受信 FIFO への格納条件を満たすかどうか確認する。

プロトコルで規定されている各種受信条件を満たしたフレームのうち、受信バッファ/受信 FIFO で設定しているフィルタリング条件 (Frame ID, Channel ID, Cycle counter) を満たすフレームのみ受信バッファに格納します。

受信バッファへのフレーム格納により、FLXAnFRNDAT1～4 レジスタ、FLXAnFRMBSC1～4 レジスタの対応するフラグが“1”になります。

このとき、FLXAnFRMHDS レジスタの MBU ビットに、受信成功したメッセージバッファ番号が格納されます。

ヘッダ設定時に受信完了割り込みを許可 (FLXAnFRWRHS1 レジスタの MBI ビット = “1”) に設定している場合には、フレーム受信完了により割り込み要求 (FLXAnFRSIR レジスタの RXI ビット = “1”) が発生します。

【注】 FLXAnFRNDAT1～FLXAnFRNDAT4 レジスタの各フラグは、Message RAM から OBF へのデータ部の転送により自動的にクリアされます。フレームのヘッダ部のみを OBF に転送しても、FLXAnFRNDAT1～FLXAnFRNDAT4 レジスタの受信完了フラグはクリアされません。

■ 受信メッセージの読み出しタイミング

受信メッセージは、受信した次のスロットの先頭から、以下の計算式で算出したワースト時間を待って読み出すことにより、確実に読み出すことができます。

計算式：

AchとBchの並行処理分 受信したスロットと、次の次のスロット分

$$\text{待ち時間[秒]} = 2 \times \{17 + 2 \times \text{ceil}(\text{PLmax} / 2)\} \times (1 / \text{fbus})$$

ヘッダ部転送回数+セットアップクロック数 ペイロード部転送回数 周辺バスクロック周波数

PLmax:最大ペイロード長(word)
ceil(x):xを超える最小整数値

待ち時間の算出例を、以下に示します。

(1) ペイロードが 16word(32byte)、周辺バスクロックが 80MHz の時

$$\begin{aligned} \text{待ち時間} &= 2 \times \{17 + 2 \times \text{ceil}(16 / 2)\} \times \{1 / (80 \times 10^6)\} \\ &= 2 \times (17 + 2 \times 8) \times \{1 / (80 \times 10^6)\} \\ &= 82.5 \text{ [nsec]} \end{aligned}$$

(2) ペイロードが 128word(254byte)、周辺バスクロックが 80MHz の時

$$\begin{aligned} \text{待ち時間} &= 2 \times \{17 + 2 \times \text{ceil}(128 / 2)\} \times \{1 / (80 \times 10^6)\} \\ &= 2 \times (17 + 2 \times 64) \times \{1 / (80 \times 10^6)\} \\ &= 3.6 \text{ [usec]} \end{aligned}$$

6. 割り込み

6.1 割り込み制御

RH850/P1x の FlexRay 関連割り込みを表 6-1 に示します。割り込みについての詳細は、RH850/P1x ハードウェアマニュアル『第 6 章. 割り込み』を参照ください。

表 6-1 FlexRay 関連割り込み

割り込み要因	EIINT 割り込み チャンネル番号	オフセットアドレス (テーブル参照方式)
FlexRay 0 割り込み	509	+3FAH
FlexRay 1 割り込み	510	+3FCH
タイマ 0 割り込み	511	+3FEH
タイマ 1 割り込み	512	+400H
タイマ 2 割り込み	513	+402H
FIFO 転送割り込み	514	+404H
FIFO 転送警告割り込み	515	+406H
出力転送警告割り込み	516	+408H
出力転送終了割り込み	517	+40AH
入力キューフル割り込み	518	+40CH
入力キューエンpty割り込み	519	+40EH

6.2 FlexRay0 割り込み、FlexRay1 割り込み

FlexRay0 割り込み、FlexRay1 割り込みは以下のような条件で割り込みを発生させることができます。

- フレーム送受信完了
- 通信エラーを検出
- 各種ステータスの変化
- タイマカウンタが規定値に達した
- IBF/OBF ⇄ Message RAM 間の転送完了

各割り込みは、FLXAnFREIES (Set) / FLXAnFREIER (Reset) レジスタ、FLXAnFRSIES (Set) / FLXAnFRSIER (Reset) レジスタにより、許可/禁止の設定を行うことができます。Set レジスタの該当ビットに"1"を設定することで割り込みが許可になり、Reset レジスタの該当ビットに"1"を設定することにより割り込みが禁止になります ("0"書き込みは、レジスタ値に影響を与えません)。

各割り込みのステータスは、割り込みの許可/禁止に関係なく FLXAnFREIR レジスタ・FLXAnFRSIR レジスタに反映されます。ステータスレジスタの各ビットをクリアする場合は、該当ビットに"1"を設定してください ("0"書き込みは、レジスタ値に影響を与えません)。

各割り込みは、割り込み出力選択レジスタ (FLXAnFREILS / FLXAnFRSILS) により、FlexRay0 割り込み、FlexRay1 割り込みのどちらの割り込みを使用するかを選択できます。

割り込み設定例を表 6-2、チャンネル A エラー検出割り込みを使用する場合の制御手順例を図 6-1 に示します。

表 6-2 割り込みと要求フラグ、割り込みアサイン例

割り込み名称	要求フラグ	割り込みアサイン
受信割り込み	FLXAnFRSIR_RXI	FlexRay1 割り込み
POC エラーモード変更	FLXAnFREIR_PEMC	FlexRay0 割り込み
チャンネル A エラー検出	FLXAnFREIR_EDA	FlexRay0 割り込み
チャンネル B エラー検出	FLXAnFREIR_EDB	FlexRay0 割り込み

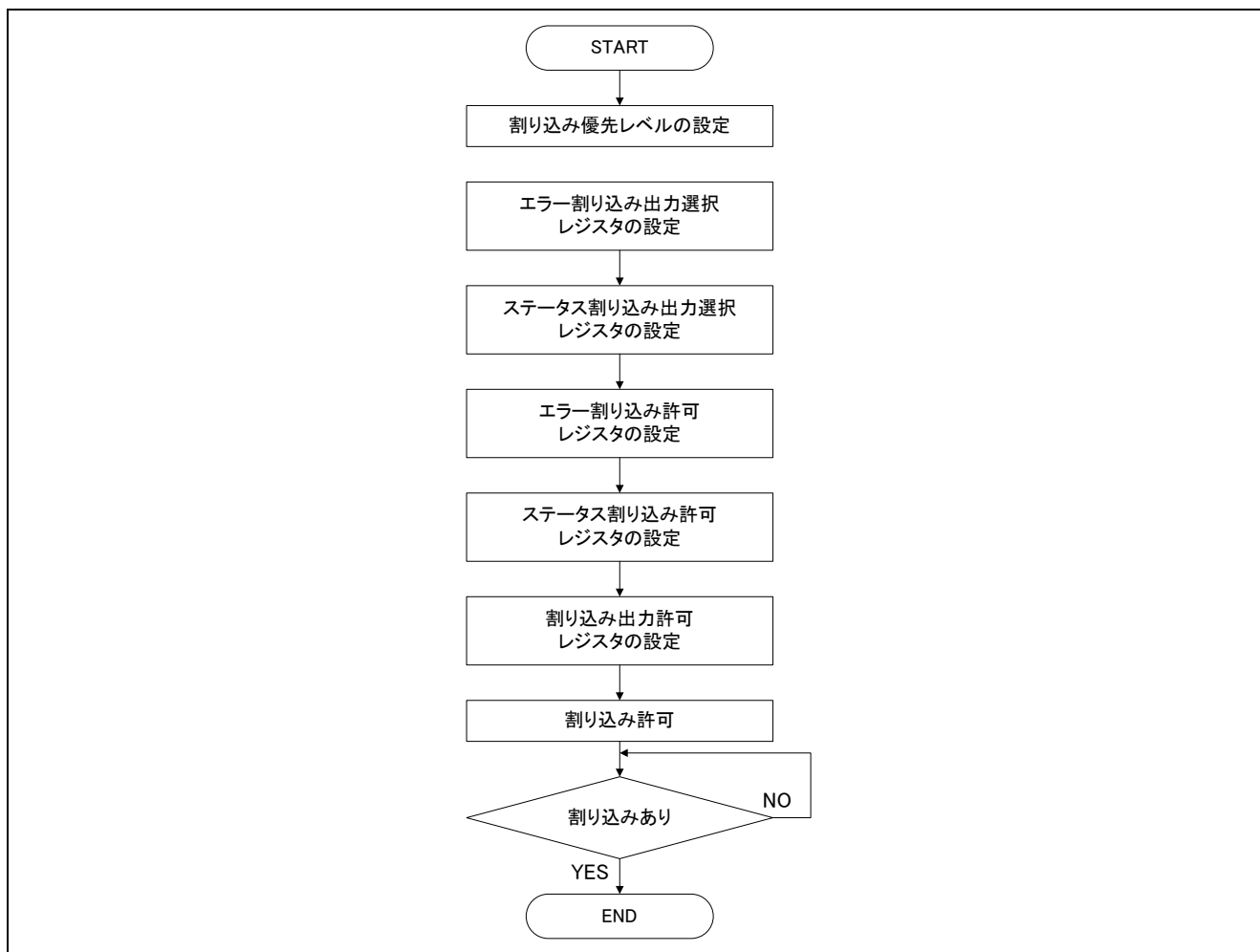


図 6-1 チャンネル A エラー検出割り込みの制御手順例

【注】 受信割り込みを使用する場合、該当するメッセージバッファの MBI ビットを“1”に設定してください。

6.3 割り込み要求クリアについて

FlexRay モジュールでは、割り込みステータスレジスタ（FLXAnFREIR、FLXAnFRSIR）の各ビットに“1”を書き込むことにより、割り込み要求のクリアを行います。

割り込み要求をクリアするには、該当ビットのみを“1”にした値（即値）をレジスタに書き込んでください。ビットフィールド構造体・論理演算を使用して割り込み要求ビットをクリアした場合、誤って他の割り込み要求をクリアする場合があります。

7. タイマ

FlexRay モジュールは、以下の 2 種類のタイマを内蔵しています。

- タイマ 0
- タイマ 1

7.1 タイマ 0

絶対値タイマです。割り込み発生タイミングを、サイクルカウント値とサイクル開始からのオフセット値 (MT 値) で設定します。タイマ 0 は FLXAnFRT0C レジスタの T0RC に"1"を設定することでスタートします。割り込み発生タイミングは FLXAnFRT0C レジスタの T0CC ビットおよび TOMO ビットで設定します。

- 【注】
1. FLXAnFRT0C レジスタへタイマ値を設定する場合は、あらかじめ T0RC ビットへ"0"を設定し、タイマ T0 を停止してください。
 2. タイマ 0 は、Normal_Active 状態でのみ動作可能です。他の状態へ遷移した場合、タイマ 0 は停止します。

タイマ 0 設定例を表 7-1、制御手順例を図 7-1 に示します。

表 7-1 タイマ 0 設定例

名称	設定	備考
サイクルカウンタ値	2 コミュニケーション サイクルごと	偶数コミュニケーションサイクルごとに割り込み発生
MT オフセット値	1,000 MT	コミュニケーションサイクル先頭から 1,000MT 後
タイマ動作モード	連続モード	設定条件を満たすたびに割り込み要求発生

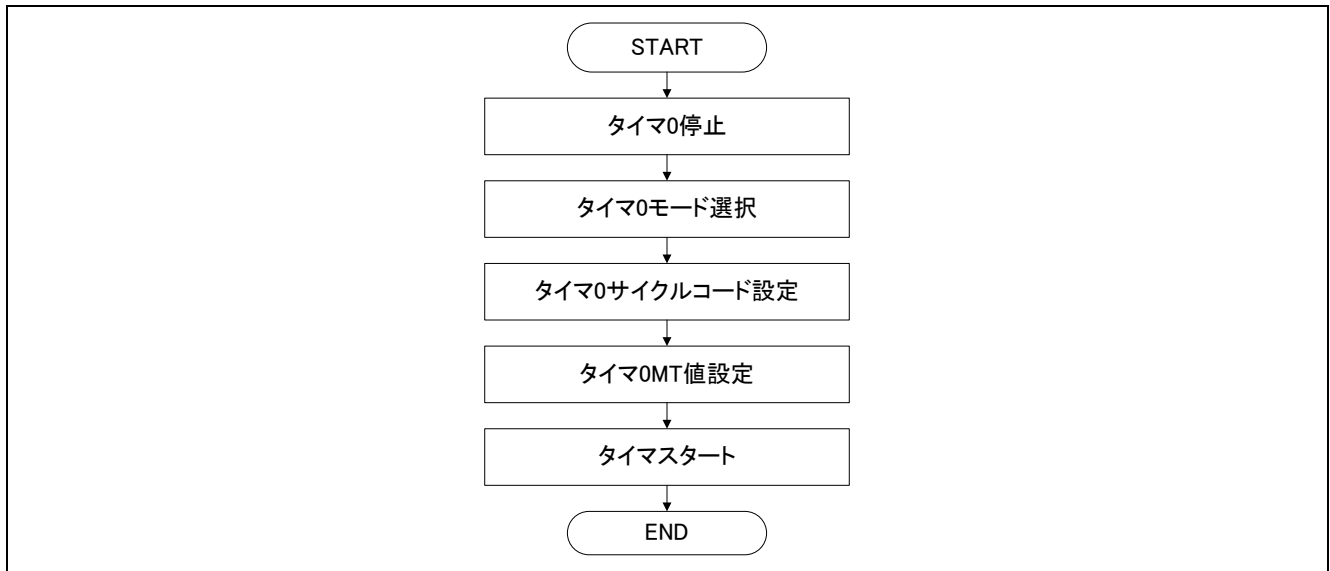


図 7-1 制御手順例

7.2 タイマ 1

相対値タイマです。割り込み発生タイミングを、タイマスタート時からの経過時間（MT 値）で設定します。タイマ 1 は FLXAnFRT1C レジスタの T1RC に”1”を設定することでスタートします。割り込み発生タイミングは FLXAnFRT1C レジスタの T1MC ビットで設定します。

- 【注】
1. FLXAnFRT1C レジスタへタイマ値を設定する場合は、あらかじめ T1RC ビットへ”0”を設定し、タイマ 1 を停止してください。
 2. タイマ 1 は、NORMAL_ACTIVE 状態でのみ動作可能です。他の状態へ遷移した場合、タイマ 1 は停止します。

タイマ 1 設定例を表 7-2、制御手順例を図 7-2 に示します。

表 7-2 タイマ 1 設定例

名称	設定	備考
MT カウント	100 MT 後	
タイマ動作モード	シングルショットモード	タイマ 1 割り込み発生により停止

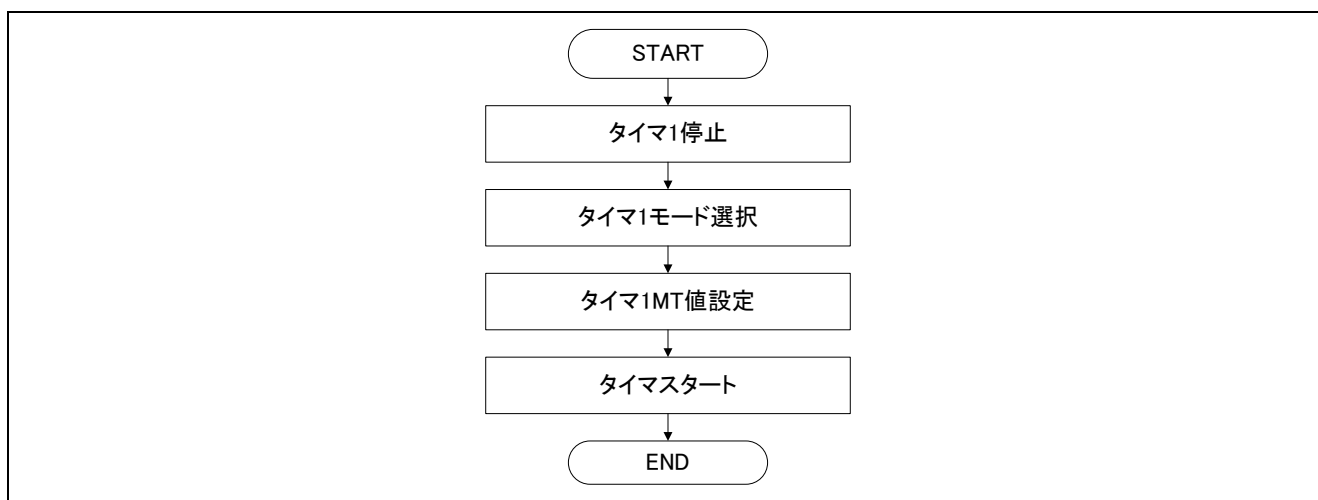


図 7-2 制御手順例

8. ストップウォッチタイマ

ストップウォッチタイマは、トリガイイベントが発生したときの、サイクルカウンタ値、スロットカウンタ値および MT 値をキャプチャし、下記のレジスタに格納します。

- サイクルカウンタ値 : FLXAnFRSTPW1 レジスタの SCCV0～SCCV5 ビット
- MT 値 : FLXAnFRSTPW1 レジスタの SMTV0～SMTV13 ビット
- チャンネル A スロットカウンタ値 : FLXAnFRSTPW2 レジスタの SSCVA0～SSCVA10 ビット
- チャンネル B スロットカウンタ値 : FLXAnFRSTPW2 レジスタの SSCVB0～SSCVB10 ビット

ストップウォッチタイマのトリガイイベントは下記のとおりです。

- FlexRay 0 割り込み要求の発生
- FlexRay 1 割り込み要求の発生
- ソフトウェアトリガ (FLXAnFRSTPW1 レジスタの SSWT ビット = “1”)

- 【注】
1. FlexRay0/1 割り込みトリガとソフトウェアトリガを両方許可にすることはできません。
 2. ストップウォッチトリガによる各値のキャプチャは、トリガ発生時とトリガが発生した次の MT 開始時です。

ストップウォッチトリガの設定例を表 8-1、制御手順例を図 8-1 に示します。

表 8-1 ストップウォッチトリガ設定例

名称	設定	備考
FlexRay0/1 割り込みトリガ	禁止	
ストップウォッチ動作モード	シングルショットモード	ストップウォッチ割り込み発生により停止
ソフトウェアトリガ	許可	

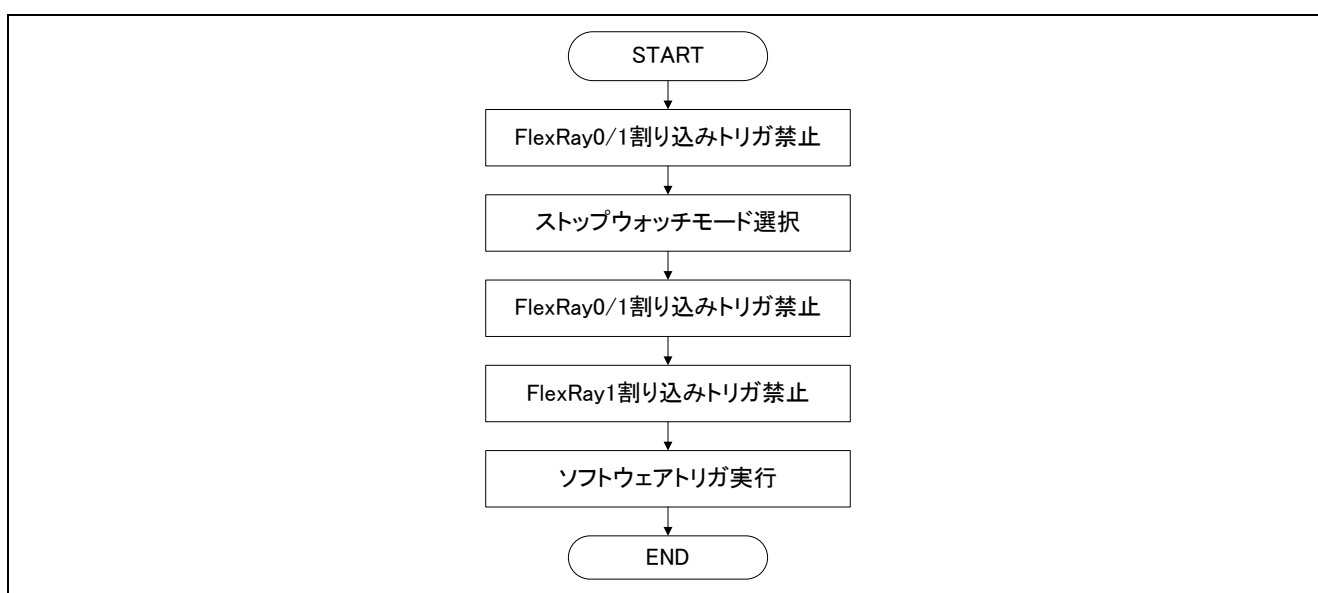


図 8-1 制御手順例

9. ネットワーク管理機能

FlexRay モジュールには、Static セグメントで受信したフレームのうち、Payload preamble indicator (PPI) フラグが'1'にセットされているフレームの先頭部分を「NM ベクタ」として抽出する、ネットワーク管理機能を内蔵しています。

通信コントローラは、1 コミュニケーションサイクル中に受信した全ての NM ベクタのビットごとの OR (論理和) をとり、NM ベクタ情報として保存します。現在コミュニケーションサイクルでの NM ベクタ情報と前回コミュニケーションサイクルでの NM ベクタ情報を比較し値が異なっている場合、割り込み要求を発生させます。

【注】 NM ベクタ情報 (FLXAnFRNMV1~3 レジスタ) は、コミュニケーションサイクルの終わりに更新されます。

ネットワーク管理機能設定例を表 9-1 に示します。

表 9-1 ネットワーク管理機能設定例

名称	設定	備考
ネットワーク管理機能	使用する	サンプルプログラムでは未使用に設定しています
NM ベクタ長	4 バイト	0~12 バイトの範囲で、1 バイト単位で設定

必要に応じて Node_1.h ファイルのマクロ定義値 : NEMC_SET を変更してください。

10. 受信 FIFO

FlexRay モジュールは、受信 FIFO 機能を内蔵しています。

FIFO バッファを MessageRAM に構成する場合、FIFO バッファの開始バッファ番号は FRMRC レジスタの FFB0~FFB7 ビット、終了バッファ番号は FLXAnFRMRC レジスタの LCB0~LCB7 ビットで設定できません。

FIFO バッファは最大 128 個構成可能です。MessageRAM の構成方法については『1. 初期設定』を参照してください。

受信データが、Static バッファおよび Dynamic バッファのフィルタリング条件に合わず、FIFO バッファのフィルタリングに合致した場合、FIFO バッファに格納されます。この場合、該当するメッセージバッファの MBS ビットは、受信フレームのフレーム ID、Payload 長、受信サイクルカウンタ、ステータスで上書きされます。

10.1 FIFO フィルタリング

FIFO バッファのフィルタリングはチャンネルフィルタ、フレーム ID フィルタ、サイクルカウンタフィルタで構成されます。

10.1.1 FIFO リジェクションフィルタの構成

FLXAnFRFRF レジスタと FLXAnFRFRFM レジスタに、受信データを FIFO バッファに格納するかしないかのリジェクション条件を設定します。

FLXAnFRFRF レジスタの RSS ビットが"1"の場合（デフォルト）、Static セグメントでのすべての受信メッセージは FIFO によってリジェクションされます。

FLXAnFRFRF レジスタの RNF ビットが"1"の場合、受信された Null フレームは FIFO バッファに格納されません。

Null フレームが FIFO リジェクションフィルタによってリジェクションされなかった場合は、Null フレームは他のデータと同様に FIFO バッファに格納されます。この場合、NDAT レジスタの該当するビットが"1"になります。

FLXAnFRFRF レジスタの FID0~FID10 ビットにてフレーム ID のリジェクションフィルタリング条件を設定します。FLXAnFRFRFM レジスタの MFID0~MFID10 ビットにて、FLXAnFRFRF レジスタの FID0~FID10 ビットで設定した各ビットのフィルタリング条件を適応するか・しないかを設定し、実際に使用するリジェクションフレーム ID を決定します。MFID0~MFID10 ビットに"1"を設定した場合、対応する MFID0~MFID10 ビットの設定は無視されます。

表 10-1~表 10-5 にフレーム ID リジェクトフィルタ設定例を示します。

表 10-1 リジェクトフィルタ設定例 1

ビット名	設定値
FLXAnFRFRF レジスタ FID ビット	000 0000 0011b
FLXAnFRFRFM レジスタ MFID ビット	000 0000 0000b
リジェクトフレーム ID	000 0000 0011b

『説明』

フレーム ID = 3 がリジェクトされます。

表 10-2 リジェクトフィルタ設定例 2

ビット名	設定値
FRF レジスタ FID ビット	000 0000 0011b
FLXAnFRFRFM レジスタ MFID ビット	000 0000 0001b
リジェクトフレーム ID	000 0000 001Xb (X は任意)

『説明』

フレーム ID2 とフレーム ID3 がリジェクトされます。

表 10-3 リジェクトフィルタ設定例 3

ビット名	設定値
FLXAnFRFRF レジスタ FID ビット	000 0000 1000b
FLXAnFRFRFM レジスタ MFID ビット	000 0000 0111b
リジェクトフレーム ID	000 0000 1XXXb (X は任意)

『説明』

フレーム ID8~フレーム ID15 がリジェクトされます。

表 10-4 リジェクトフィルタ設定例 4

ビット名	設定値
FLXAnFRFRF レジスタ FID ビット	000 0000 1000b
FLXAnFRFRFM レジスタ MFID ビット	000 0000 0100b
リジェクトフレーム ID	000 0000 1X00b (X は任意)

『説明』

フレーム ID8~フレーム ID12 がリジェクトされます。

表 10-5 リジェクトフィルタ設定例 5

ビット名	設定値
FLXAnFRFRF レジスタ FID ビット	000 0000 1000b
FLXAnFRFRFM レジスタ MFID ビット	000 0001 1111b
リジェクトフレーム ID	000 000X XXXXb (X は任意)

『説明』

フレーム ID1~フレーム ID31 がリジェクトされます。

10.2 FIFO バッファの読み出し

受信 FIFO に格納されているフレームを読み出すには、OBF バッファに対して「FIFO 先頭バッファ内容読み出し処理」を行います。

DEFAULT_CONFIG または CONFIG 状態以外で FIFO バッファから OBF バッファへの転送トリガを開始するために、FLXAnFROBCR レジスタの OBRS0~OBRS6 ビットへ FIFO バッファの最初のバッファ番号を設定します。この書き込みが転送要求となります。

FIFO バッファの最初のバッファ番号は、FLXAnFRMRC レジスタの FFB0~FFB7 ビットで参照できません。

FIFO バッファからデータを読み出すためには以下のシーケンスを行います。詳細は、『3.3 OBF（アウトプットバッファ）：MessageRAM からの読み出し』を参照してください。

FIFO バッファからの読み出し設定例を表 10-6、制御手順例を図 10-1 に示します。

表 10-6 設定例

名称	設定	備考
ヘッダ部転送	する	
データ部転送	する	
FIFO バッファ先頭番号	8	

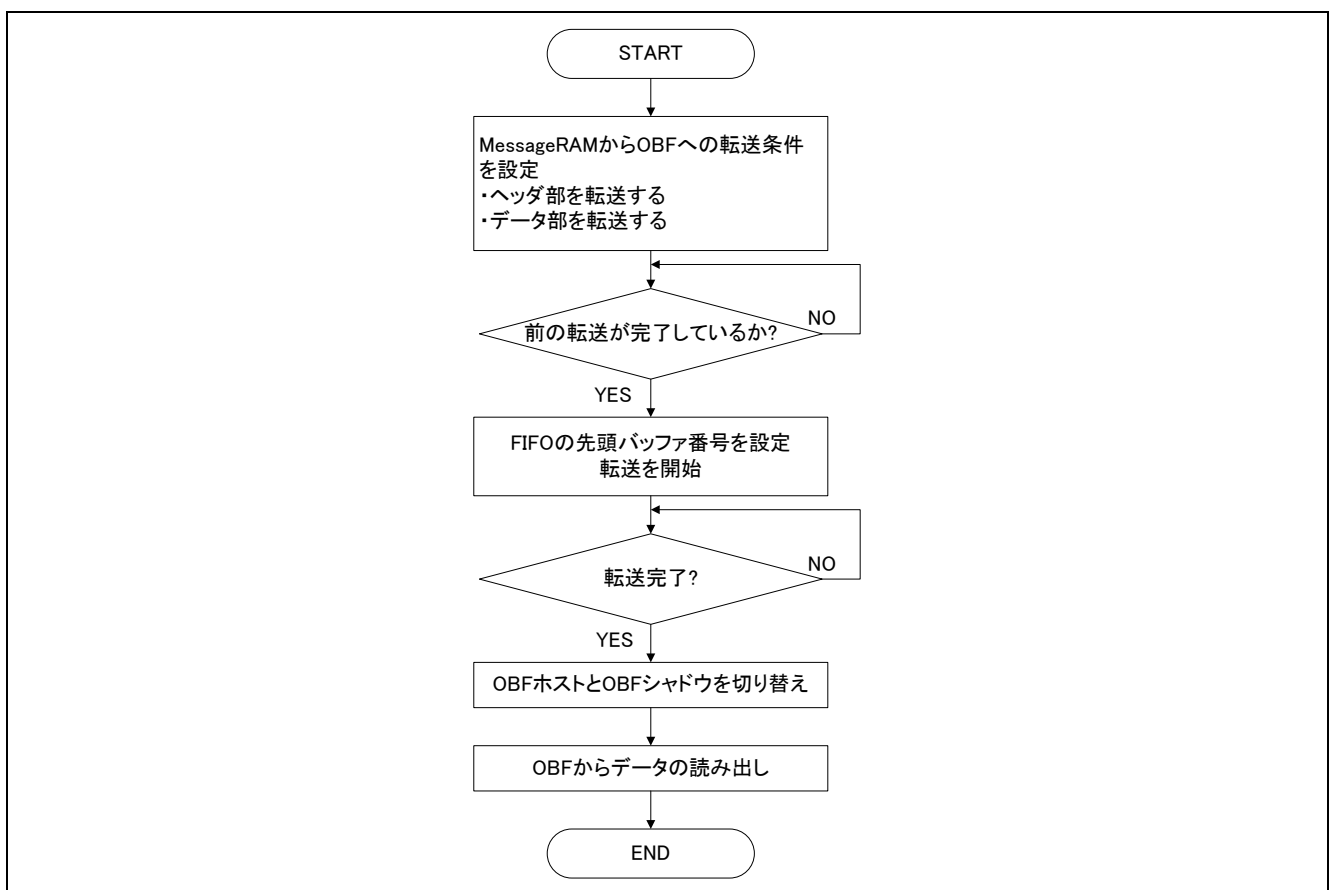


図 10-1 制御手順例

- 【注】
1. FIFO バッファの「FIFO 先頭バッファ」以外に対して読み出し処理を行わないでください。
 2. FIFO バッファの空読みは行わないでください。
空読みを行った場合、FLXAnFREIR レジスタの EFA（Empty FIFOAccess）フラグがセットされます。

読み出し処理は、FLXAnFRSIR レジスタ RFNE (Receive FIFO Not Empty) などにより、FIFO にデータが格納されていることを確認した上で行ってください。

3. FIFO バッファからデータを読み出した場合、FLXAnFRRDHS1 レジスタは受信フレームのフレーム ID (FID ビット) のみを保持します。それ以外のビットは、読み出し時、すべて"0"となります。
4. FIFO バッファから OBF バッファへデータ部の転送を行った場合 (FLXAnFROBCM レジスタの RDSS ビット = "1")、設定されたペイロード長分 (FLXAnFRWRHS2 レジスタの PLC0~PLC6 ビット、FLXAnFRRDHS2 レジスタの PLC0~PLC6 ビット) のデータが転送されます。受信したフレームのペイロード長 (FLXAnFRRDHS2 レジスタの PLR0~PLR6) は転送されません。
メッセージが FIFO バッファへ格納されたとき、受信したペイロード (FLXAnFRRDHS2 レジスタの PLR0~PLR6) と設定したペイロード長 (FLXAnFRWRHS2 レジスタの PLC0~PLC6 ビット、FLXAnFRRDHS2 レジスタの PLC0~PLC6 ビット) は次のようになります。
 - $PLR[6:0] > PLC[6:0]$:
メッセージバッファへ格納された Payload データは、設定された Payload 長に切られます。PLC+1 の長さになります。
 - $PLR[6:0] \leq PLC[6:0]$:
受信した Payload データはメッセージバッファのデータ部へ格納されます。データ部のデータは PLC によって示されるバイト数まで不定値によって満たされます。
 - $PLR[6:0] = "0"$:
メッセージバッファのデータ部は不定値によって満たされます。
 - $PLC[6:0] = "0"$:
メッセージバッファはデータ部を構成しません。メッセージバッファのデータ部にはデータは格納されません。

10.3 FIFO バッファへの書き込み

FIFO バッファへの FLXAnFRWRHS1~FLXAnFRWRHS3 レジスタの書き込みに関しては、『3. MessageRAM へのデータ設定と MessageRAM からのデータ読み出し』と同様の制御となります。

メッセージバッファを FIFO バッファとして使用する場合、FLXAnFRWRHS1 レジスタの受信条件は無視され、FLXAnFRFRF レジスタと FLXAnFRFRFM レジスタの受信条件が使用されます。

メッセージバッファ 2 に配置されている FIFO バッファへの書き込み制御手順例を図 10-2 に示します。

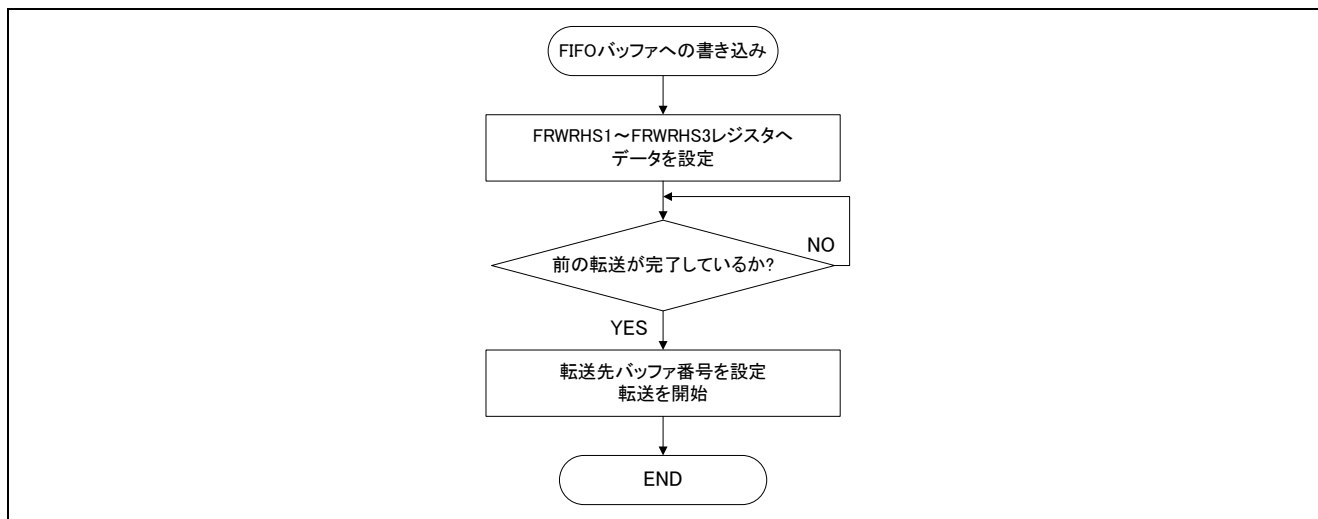


図 10-2 制御手順例

【注】 FIFO バッファは同一データ長にしてください（FLXAnFRWRHS2 レジスタの PLC0~PLC6 ビット）。

12. 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2025.12.24	全頁	新規作成

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しないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

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

5. クロックについて

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

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

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

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、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まれます。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も、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）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いたします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

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

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

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

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。