

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日  
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。
5. 輸出に際しては、「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。本資料に記載されている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の目的で使用しないでください。また、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に使用することができません。
6. 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、誤りが無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。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されてい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生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
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  
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、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社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。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、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おりません。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、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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、当社は、一切その責任を負いません。
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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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。

# コントローラエリアネットワーク (RCAN-ET)

## 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、RCAN-ET を使用した通信動作例から構成されており、ユーザにてソフトウェア設計およびハードウェア設計の際、ご参考として役立てていただけるようにまとめたものです。

なお、本アプリケーションノートに掲載されているタスク例およびアプリケーション例は動作確認しておりますが、実際にご使用になる場合には、必ず動作確認の上ご使用くださいますようお願いいたします。

### 動作確認デバイス

H8SX/1544

### 目次

1. RCAN-ET の概要.....	2
1.1 RCAN-ET の特長.....	2
1.2 入出力端子 .....	2
1.3 レジスタ機能.....	3
1.4 割り込み要因.....	4
1.5 ビットレートの設定.....	5
2. RCAN-ET の応用例.....	9
2.1 初期設定.....	9
2.2 データフレーム送受信 (スタンダード ID) .....	19
2.3 データフレーム送受信 (エクステンデッド ID) .....	31
2.4 リモートフレーム送受信 (ATX 使用) .....	47
2.5 自動再送信無効機能 (DART) .....	64
2.6 ニューメッセージコントロール (NMC) .....	73
2.7 メッセージフィルタリング (LAFM) によるグループ受信.....	92
2.8 メッセージフィルタリング (LAFM) の IDE ビットマスク.....	122
2.9 CAN wake-up 機能.....	140
2.10 DMAC を使用した受信メッセージの転送.....	158
2.11 メールボックスの再設定.....	183
2.12 CAN 経由 FLASH 書き込み.....	205

## 1. RCAN-ET の概要

### 1.1 RCAN-ET の特長

#### 1.1.1 特長

RCAN-ET の主な特長を以下に示します。

- CAN 規格 2.0B および SO-11898 規格に対応
- 16 個のメールボックス (送受信 15 個、受信 1 個)
- 全メールボックスに対して受信フィルタマスク対応 (LAFM)
- テスト機能内蔵
- HCAN2 と類似したレジスタ構成

#### 1.1.2 追加機能

RCAN-ET で新たに追加された機能について以下に示します。

- ID 並べ替え機能 (HCAN2 と互換性のある ID,RTR,IDE の順序に設定可能)
- バスオフ時ホルト (バスオフ状態に遷移後、直ちにホルトモードに遷移可能)
- IDE ビットのフィルタマスクビット追加
- メールボックスを RAM とレジスタで構成

### 1.2 入出力端子

H8SX/1544 の RCAN-ET 端子構成を表 1.2.1 に示します。H8SX/1544 には 2 チャンネルの RCAN-ET が内蔵されています。

表 1.2.1 端子構成 (H8SX/1544)

チャンネル	名称	端子名	入出力	機能
0	送信データ端子	CTx_0	出力	CAN バス送信用端子
	受信データ端子	CRx_0	入力	CAN バス受信用端子
1	送信データ端子	CTx_1	出力	CAN バス送信用端子
	受信データ端子	CRx_1	入力	CAN バス受信用端子

### 1.3 レジスタ機能

表 1.3.1 に RCAN-ET のレジスタ機能について示します。

表 1.3.1 レジスタ機能

レジスタ名	略称	機能
メッセージコントロール 0	MB[x].CONTROL0	スタンダードフォーマット/エクステンデッドフォーマットの選択、データフレーム/リモートフレームの選択、スタンダード ID、エクステンデッド ID の設定を行います。
ローカルアクセプタンスフィルタ	MB[x].LAFM	IDE ビット、スタンダード ID、エクステンデッド ID のフィルタマスク設定を行います。
メッセージデータ	MB[x].MSG_DATA[n] (n=0~7)	送受信データの設定を行います。
メッセージコントロール 1	MB[x].CONTROL1	ニューメッセージコントロール (NMC)、データフレーム自動送信 (ATX)、自動再送信無効 (DART)、メールボックス構成 (MBC)、データ長 (DLC) の設定を行います。
マスタコントロールレジスタ	MCR	RCAN-ET の動作を制御します。
ジェネラルスレータスレジスタ	GSR	RCAN-ET の状態を示します。
ビットタイミング コンフィギュレーションレジスタ	BCR1/BCR0	CAN のビットタイミングパラメータとボーレートプリスケアラを設定します。
インタラプトリクエストレジスタ	IRR	各種割り込み要因のステータスフラグです。
インタラプトマスクレジスタ	IMR	IRR の各ビットに対応する割り込み要求をマスクします。
送信エラーカウンタ 受信エラーカウンタ	TEC/REC	送信/受信メッセージエラー数を示すカウンタです。
送信待ちレジスタ	TXPR	メールボックスに格納したメッセージを送信待ち状態にします。
送信キャンセルレジスタ	TXCR	送信待ちメッセージの送信リクエストをキャンセルします。
送信アクノレッジレジスタ	TXACK	メールボックスの送信メッセージが正常に送信されたことを示すステータスレジスタです。
アポータアクノレッジレジスタ	ABACK	メールボックスの送信メッセージが正常に取り消されたことを示すステータスレジスタです。
データフレーム受信完了レジスタ	RXPR	メールボックスがデータフレームを正常に受信したことを示すステータスレジスタです。
リモートフレーム受信完了 レジスタ	RFPR	メールボックスがリモートフレームを正常に受信したことを示すステータスレジスタです。
メールボックスインタラプト マスクレジスタ	MBIMR	各メールボックスの割り込み要求をイネーブルにします。
未読メッセージステータス レジスタ	UMSR	未読メッセージのあるメールボックスに新たなメッセージを受信し、オーバライト/オーバランが発生したことを示すステータスフラグです。
RCAN-ET モニタレジスタ	RCANMON*	RCAN-ET の送受信端子の許可/禁止、送信端子の送信停止制御、送受信端子の端子状態をモニタします。

【注】 1. x : メールボックスの番号 (0~15)

2. RCAN-ET のレジスタはアクセスサイズに制限があります。各レジスタのアクセスサイズについてはハードウェアマニュアルを参照ください。

\* RCAN-ET モニタレジスタ (RCANMON) は、内蔵されていないデバイスもあります。

## 1.4 割り込み要因

表 1.4.1 に、RCAN-ET の割り込み要因を示します。これらの要因は、パワーオンリセットによるリセット処理割り込み (IRR0) を除き、マスクすることが可能です。マスクには、メールボックスインタラプトマスクレジスタ (MBIMR) およびインタラプトマスクレジスタ (IMR) を使用します。各割り込み要求の割り込みベクタについては、「H8SX/1544 ハードウェアマニュアル 第 5 章 割り込みコントローラ」を参照してください。

表 1.4.1 RCAN-ET の割り込み要因

名称	説明	割り込みフラグ	DMAC の起動
ERS_n	エラーパッシブ (TEC 128 または REC 128)	IRR5	不可
	バスオフ (TEC 256) / バスオフからの復帰	IRR6	
	エラーワーニング (TEC 96)	IRR3	
	エラーワーニング (REC 96)	IRR4	
OVR_n	メッセージエラー検出 (テストモードのみ有効)	IRR13	不可
	リセット / ホルト / CAN スリープ遷移	IRR0	
	オーバーロードフレーム送信	IRR7	
	未読メッセージのオーバーライト / オーバラン	IRR9	
	CAN スリープ中 CAN バス動作の検出	IRR12	
SLE_n	メッセージの送信 / 送信取り消し (メールボックスエンプティ)	IRR8	不可
RM1_n	メールボックス 1~15 のリモートフレーム受信	IRR2	
	メールボックス 1~15 のデータフレーム受信	IRR1	
RM0_n	メールボックス 0 のリモートフレーム受信	IRR2	可
	メールボックス 0 のデータフレーム受信	IRR1	

【注】 n : RCAN のチャンネル番号 (0,1)

## 1.5 ビットレートの設定

RCAN-ET のビットレートは、ビットタイミングコンフィギュレーションレジスタ (BCR0、BCR1) で設定します。ビットレートは、下記の計算式により算出されます。

$$\text{ビットレート} = \text{fclk} / 2 \times (\text{BRP} + 1) \times (\text{TSEG1} + \text{TSEG2} + 1)$$

BRP : ポーレートプリスケアラ、TSEG1、TSEG2 : タイムセグメント 1,2、fclk : 周辺クロック(Pφ)

本項では、BCR 一覧表を使用した、BCR レジスタ設定方法について説明します。BCR 一覧表を使用することで、上記の計算式を使用せずに、CAN 仕様で規定された範囲内の BCR レジスタ設定が可能です。

### < BCR 一覧表の使用方法 >

- 1 . ビットレートを 1Mbps,500kpbs,250kpbs\* より選択します。
- 2 . 周辺クロック ( Pφ ) 設定値を 16MHz, 18MHz, 20MHz\* より選択します。
- 3 . SJW (再同期ジャンプ幅)、BSP (ビットサンプルポイント) の設定値を選択します。  
設定方法は、表 1.5.1 を参照してください。
- 4 . BCR 一覧表 (表 1.5.2 ~ 1.5.4) で、対応する BCR レジスタ値を設定してください。  
また、SJW の設定値が TSEG2 SJW を満たしていることを確認してください。  
 ビットレート 1Mbps    表 1.5.2 参照  
 ビットレート 500kpbs    表 1.5.3 参照  
 ビットレート 250kpbs    表 1.5.4 参照

【注】 \* 1Mbps, 500kpbs, 250kpbs 以外のビットレートを設定する場合、または、Pφを 16, 18, 20MHz 以外の周波数に設定する場合には、上記の計算式を使用して BCR 設定値を算出してください。設定方法の詳細は、ハードウェアマニュアルを参照してください。

< 使用例 > ビットレート 500kpbs、Pφ = 16MHz、サンプルポイント 75% 時の BCR レジスタ設定値

Pφ (MHz)	SP*1 (%)	1bit の TQ 数	SYNC_SEG	TSEG1	TSEG2	BCR1*2	BCR0
16	68.8	16	1	10	5	H'94xx	H'0000
	75.0	8	1	5	2	H'41xx	H'0001
		16	1	11	4	H'A3xx	H'0000

【注】 \*1 SP : サンプルポイント = (1+TSEG1) の TQ 数 / 1bit の TQ 数    TQ : タイムクォンタム

\*2 BCR1 の下位 8bit には、SJW (再同期ジャンプ幅)、BSP (ビットサンプルポイント) を設定します。

上記の表より、SJW=0、BSP=0 とすると、BCR レジスタ設定値は

**BCR1 = H'4100、BCR0 = H'0001 または BCR1 = H'A300、BCR0 = H'0000**

となります。

表 1.5.1 に、SJW,BSP 設定方法を、表 1.5.2～表 1.5.4 に、ビットレート 1Mbps, 500kbps,250kbps 時の BCR レジスタ設定表を示します。

表 1.5.1 SJW,BSP 設定方法

ビット名	ビット	機能	設定値	設定内容
SJW	BCR1 の bit5,4	再同期ジャンプ幅	0	ジャンプ幅 = 1 タイムクオンタム
			1	ジャンプ幅 = 2 タイムクオンタム
			2	ジャンプ幅 = 3 タイムクオンタム
			3	ジャンプ幅 = 4 タイムクオンタム
BSP	BCR1 の bit0	ビットサンプルポイント	0	1 箇所でのビットサンプリング (TSEG1 の最後)
			1	3 箇所でのビットサンプリング (TSEG1 の最終 3 クロックサイクルの立ち上がりエッジ)

- 【注】 1. BCR1 の bit7,6,3,2,1 はリザーブビットです。書き込む値は常に 0 にしてください。  
 2. SJW は、CAN 仕様で 1 SJW 4TQ (タイムクオンタム)、TSEG1 > TSEG2 SJW と規定されています。

表 1.5.2 BCR 一覧表 (1Mbps)

P <sub>φ</sub> (MHz)	SP*1 (%)	1bit の TQ 数	SYNC_SEG	TSEG1	TSEG2	BCR1*2	BCR0
16	62.5	8	1	4	3	H'32xx	H'0000
	75.0	8	1	5	2	H'41xx	H'0000
18	66.7	9	1	5	3	H'42xx	H'0000
	77.8	9	1	6	2	H'51xx	H'0000
20	60.0	10	1	5	4	H'43xx	H'0000
	70.0	10	1	6	3	H'52xx	H'0000
	80.0	10	1	7	2	H'61xx	H'0000

- 【注】 \*1 SP : サンプルポイント = (1+TSEG1) の TQ 数 / 1bit の TQ 数 TQ : タイムクオンタム  
 \*2 BCR1 の下位 8bit には、SJW (再同期ジャンプ幅)、BSP (ビットサンプルポイント) を設定します。設定方法は、表 1.5.1 を参照してください。



表 1.5.3 BCR 一覧表 (500kbps)

P $\phi$ (MHz)	SP <sup>*1</sup> (%)	1bit の TQ 数	SYNC_SEG	TSEG1	TSEG2	BCR1 <sup>*2</sup>	BCR0
16	56.3	16	1	8	7	H'76xx	H'0000
	62.5	8	1	4	3	H'32xx	H'0001
		16	1	9	6	H'85xx	H'0000
	68.8	16	1	10	5	H'94xx	H'0000
	75.0	8	1	5	2	H'41xx	H'0001
		16	1	11	4	H'A3xx	H'0000
	81.3	16	1	12	3	H'B2xx	H'0000
87.5	16	1	13	2	H'C1xx	H'0000	
18	55.6	18	1	9	8	H'87xx	H'0000
	61.1	18	1	10	7	H'96xx	H'0000
	66.7	9	1	5	3	H'42xx	H'0001
		18	1	11	6	H'A5xx	H'0000
	72.2	18	1	12	5	H'B4xx	H'0000
	77.8	9	1	6	2	H'51xx	H'0001
		18	1	13	4	H'C3xx	H'0000
	83.3	18	1	14	3	H'D2xx	H'0000
88.9	18	1	15	2	H'E1xx	H'0000	
20	60.0	10	1	5	4	H'43xx	H'0001
		20	1	11	8	H'A7xx	H'0000
	65.0	20	1	12	7	H'B6xx	H'0000
	70.0	10	1	6	3	H'52xx	H'0001
		20	1	13	6	H'C5xx	H'0000
	75.0	20	1	14	5	H'D4xx	H'0000
	80.0	10	1	7	2	H'61xx	H'0001
		20	1	15	4	H'E3xx	H'0000
85.0	20	1	16	3	H'F2xx	H'0000	

【注】 \*1 SP : サンプルポイント = (1+TSEG1) の TQ 数 / 1bit の TQ 数    TQ : タイムクォンタム

\*2 BCR1 の下位 8bit には、SJW (再同期ジャンプ幅)、BSP (ビットサンプルポイント) を設定します。設定方法は、表 1.5.1 を参照してください。

表 1.5.4 BCR 一覧表 (250kbps)

P $\phi$ (MHz)	SP <sup>*1</sup> (%)	1bit の TQ 数	SYNC_SEG	TSEG1	TSEG2	BCR1 <sup>*2</sup>	BCR0
16	56.3	16	1	8	7	H'76xx	H'0001
	62.5	8	1	4	3	H'32xx	H'0003
		16	1	9	6	H'85xx	H'0001
	68.8	16	1	10	5	H'94xx	H'0001
	75.0	8	1	5	2	H'41xx	H'0003
		16	1	11	4	H'A3xx	H'0001
	81.3	16	1	12	3	H'B2xx	H'0001
87.5	16	1	13	2	H'C1xx	H'0001	
18	55.6	18	1	9	8	H'87xx	H'0001
	61.1	18	1	10	7	H'96xx	H'0001
	66.7	9	1	5	3	H'42xx	H'0003
		18	1	11	6	H'A5xx	H'0001
	72.2	18	1	12	5	H'B4xx	H'0001
	77.8	9	1	6	2	H'51xx	H'0003
		18	1	13	4	H'C3xx	H'0001
	83.3	18	1	14	3	H'D2xx	H'0001
88.9	18	1	15	2	H'E1xx	H'0001	
20	60.0	10	1	5	4	H'43xx	H'0003
		20	1	11	8	H'A7xx	H'0001
	65.0	20	1	12	7	H'B6xx	H'0001
	70.0	10	1	6	3	H'52xx	H'0003
		20	1	13	6	H'C5xx	H'0001
	75.0	20	1	14	5	H'D4xx	H'0001
	80.0	10	1	7	2	H'61xx	H'0003
		20	1	15	4	H'E3xx	H'0001
85.0	20	1	16	3	H'F2xx	H'0001	

【注】 \*1 SP : サンプルポイント = (1+TSEG1) の TQ 数 / 1bit の TQ 数    TQ : タイムクォンタム

\*2 BCR1 の下位 8bit には、SJW (再同期ジャンプ幅)、BSP (ビットサンプルポイント) を設定します。設定方法は、表 1.5.1 を参照してください。

## 2. RCAN-ET の応用例

### 2.1 初期設定

本項では、RCAN-ET の初期設定方法について説明します。

#### 2.1.1 仕様

図 2.1.1 に示すように、スタンダード ID : H'555 の CAN メッセージを送信するための RCAN-ET 初期設定を行います。

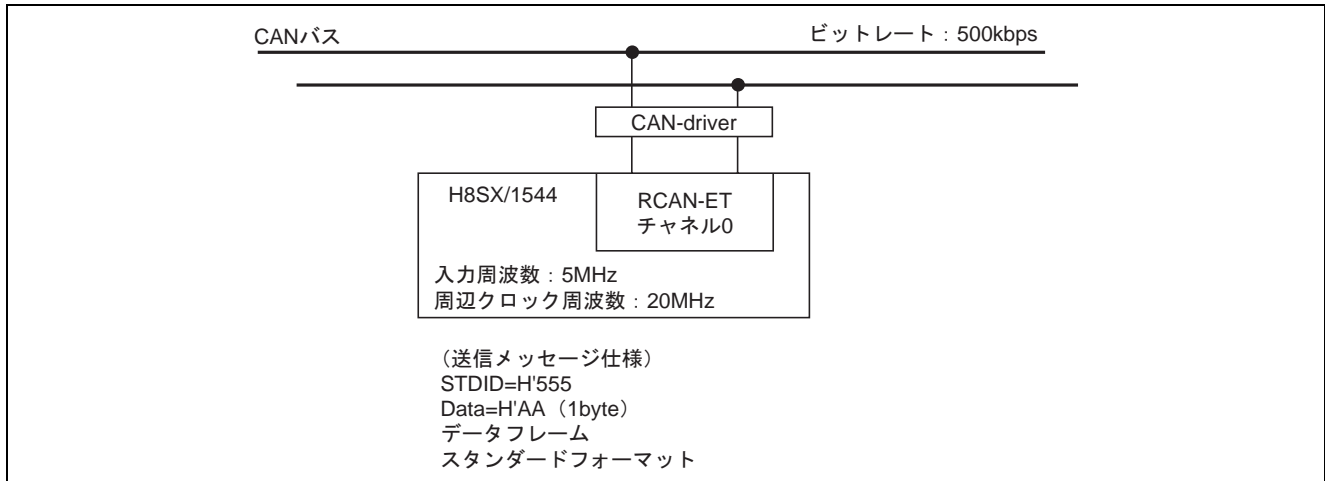


図 2.1.1 初期設定仕様

### 2.1.2 ソフトウェア説明

#### (1) モジュール説明

表 2.1.1 に、本アプリケーションノートで使用する共通の初期設定モジュールを示します。

表 2.1.1 初期設定モジュール

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	H8SX/1544 のクロック設定、送受信端子の設定。 RCAN-ET のモジュールストップモード解除。	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init	H8SX/1544 の RCAN-ET 割り込み優先レベル設定。	
RCAN 初期設定	set_RCAN0_init	RCAN-ET のリセットリクエストの設定、IRR0 クリア。	RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init	送受信に使用するメールボックスの設定。	
RCAN スタート	set_RCAN0_start	RCAN-ET のリセット解除。	

#### (2) 使用レジスタ説明

表 2.1.2 ~ 2.1.3 に、各初期設定モジュールのレジスタ設定について示します。表中の設定値の変更可能レジスタについては、システム仕様に応じて設定値の変更が可能です。

表 2.1.2 使用レジスタ説明 (デバイス個別)

モジュール名	レジスタ名	設定値	機能	設定値の 変更 <sup>*1</sup>
H8SX/1544 初期設定	SCKCR.BIT.ICK	0	システムクロックの周波数を 40MHz (入力クロック:5MHz×8) に設定。	可能
	SCKCR.BIT.BCK	1	周辺モジュールクロックの周波数を 20MHz (入力クロック:5MHz×4) に設定。	
	SCKCR.BIT.PCK	1	外部バスクロックの周波数を 20MHz (入力クロック:5MHz×4) に設定。	
	SBYCR.BIT.SSBY	1	Sleep 命令実行後、ソフトウェアスタンバイ モードに遷移するように設定。	
	MSTP.CRC.BIT._RCAN01	0	RCAN-ET のモジュールストップモードを 解除。	不可 <sup>*2</sup>
	RCANET0.RCANMON.BYTE	0x20	RCAN-ET の送受信端子をイネーブルに 設定。	
	P6.ICR.BIT.B4	1	P64 (67Pin) を RCAN-ET チャンネル 0 の 受信端子 (CRx_0) に設定。	
H8SX/1544 割り込み設定	INTC.INTCR.BIT.INTM	2	割り込みコントローラの割り込み制御モード を、割り込み制御モード 2 に設定。	可能
	INTC.IPRQ.BIT._RCAN01	7	RCAN-ET の割り込み優先レベルを設定。	

【注】 \*1 設定値が変更可能なレジスタについては、設定するレジスタの順番は変更しないでください。

\*2 RCAN-ET 送受信端子の設定方法はデバイスにより異なりますので、H8SX/1544 以外のデバイスを使用する場合は、設定方法を確認し、必要に応じて変更してください。

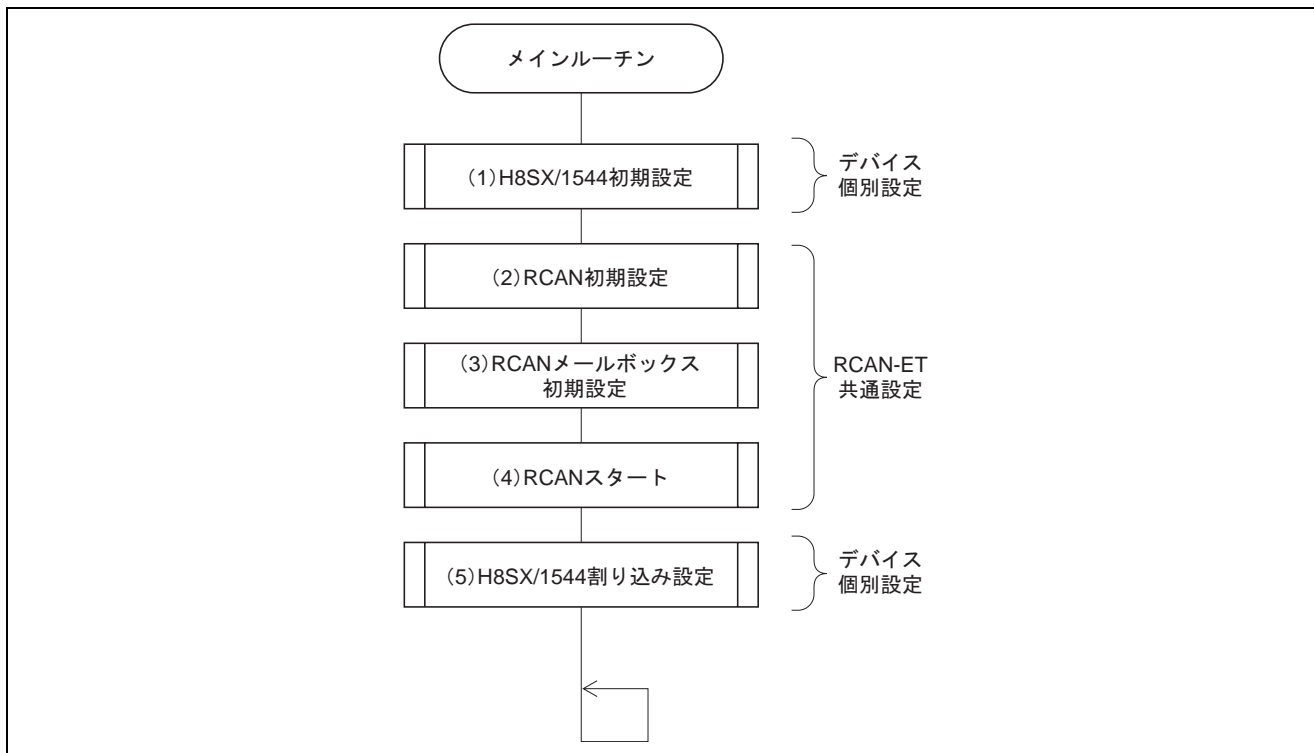
表 2.1.3 使用レジスタ説明 (RCAN - ET 共通)

モジュール名	レジスタ名	設定値	機能	設定値の変更*
RCAN 初期設定	RCANET0.MCR.WORD	0x0001	リセットリクエストビットをセット。(MCR0=1) ハードウェアリセット時のみ自動的にセット	不可
	RCANET0.GSR.WORD	-	RCAN-ET がリセット状態であることを確認。(GSR3=1を確認)	
	RCANET0.IRR.WORD	-	RCAN-ET がリセットモードに遷移したことを確認。(IRR0=1を確認)	
	RCANET0.IRR.WORD	0x0001	リセット/ホールド/スリープ割り込みフラグ(IRR0)をクリア。(クリア条件:1ライト)	
	RCANET0.MCR.WORD	0x8000	ID並びをHCAN2と異なる順序に設定。(MCR15=1)	
	RCANET0.MB[m].CTRL0.WORD.H RCANET0.MB[m].CTRL0.WORD.L RCANET0.MB[m].LAFM.WORD.H RCANET0.MB[m].LAFM.WORD.L RCANET0.MB[m].MSG_DATA[n] (m = 0 ~ 15, n = 0 ~ 7)	0	メールボックス(RAMエリア)を初期化。	
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンpty割り込み(IRR8)を許可に設定。(IMR8=0)	可能
	RCANET0.MBIMR0.WORD	0xFFFFD	メールボックス1を割り込み許可に設定。	
	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス1をスタンダードフォーマット、データフレームに設定。また、ID(H'555)を設定。	
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス1を送信用に設定。	
	RCANET0.MB[1].CTRL1.BYTE.L	0x01	データ長(1byte)を設定。	
	RCANET0.MB[1].MSG_DATA[0]	0xAA	送信データを設定。	
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz時500kbpsに設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)	
	RCANET0.BCR0.WORD	0x0001		
RCANスタート	RCANET0.MCR.WORD	0xFFFFE	リセットリクエストビットをクリア。(MCR0=0)	不可
	RCANET0.GSR.WORD	-	RCAN-ETのリセットが解除されたことを確認。(GSR3=0を確認)	

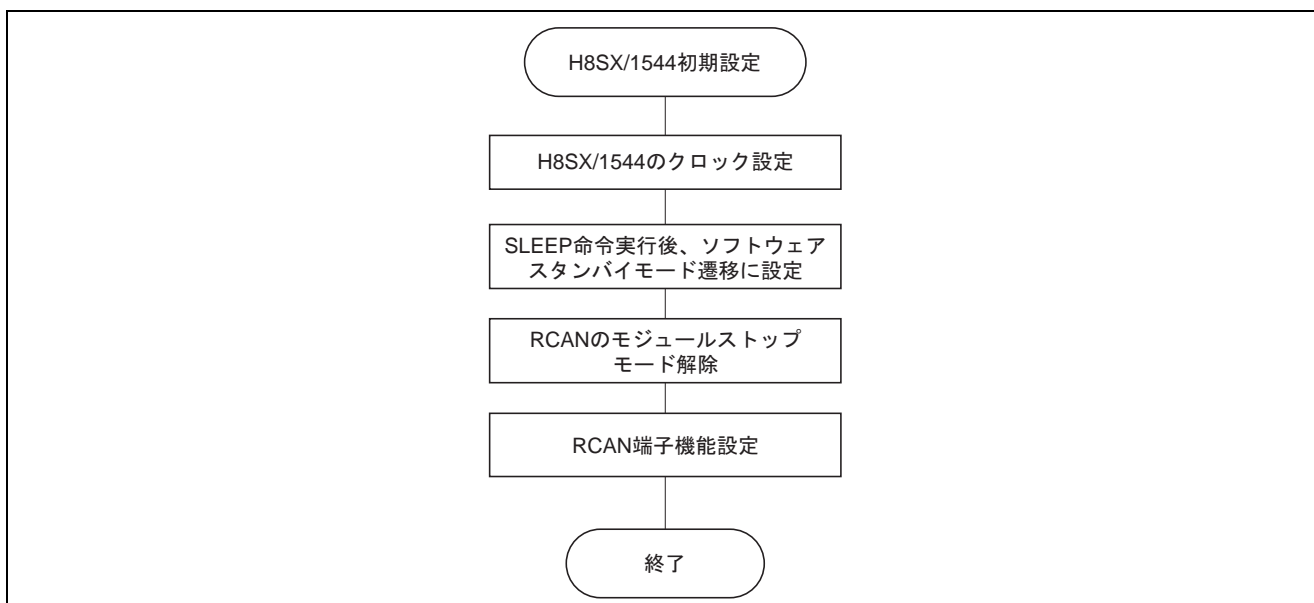
【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

\* 設定値が変更可能なレジスタについては、設定するレジスタの順番は変更しないでください。

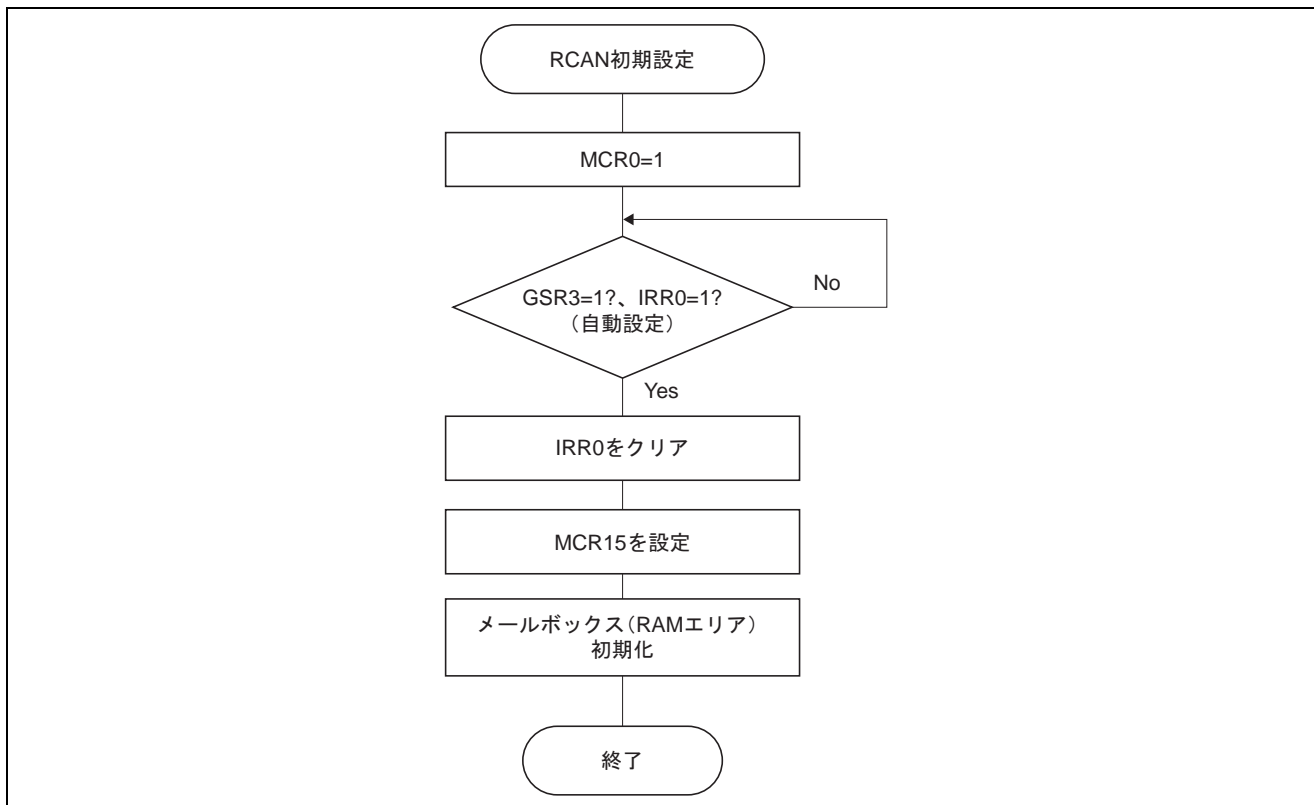
2.1.3 フローチャート  
メインルーチン



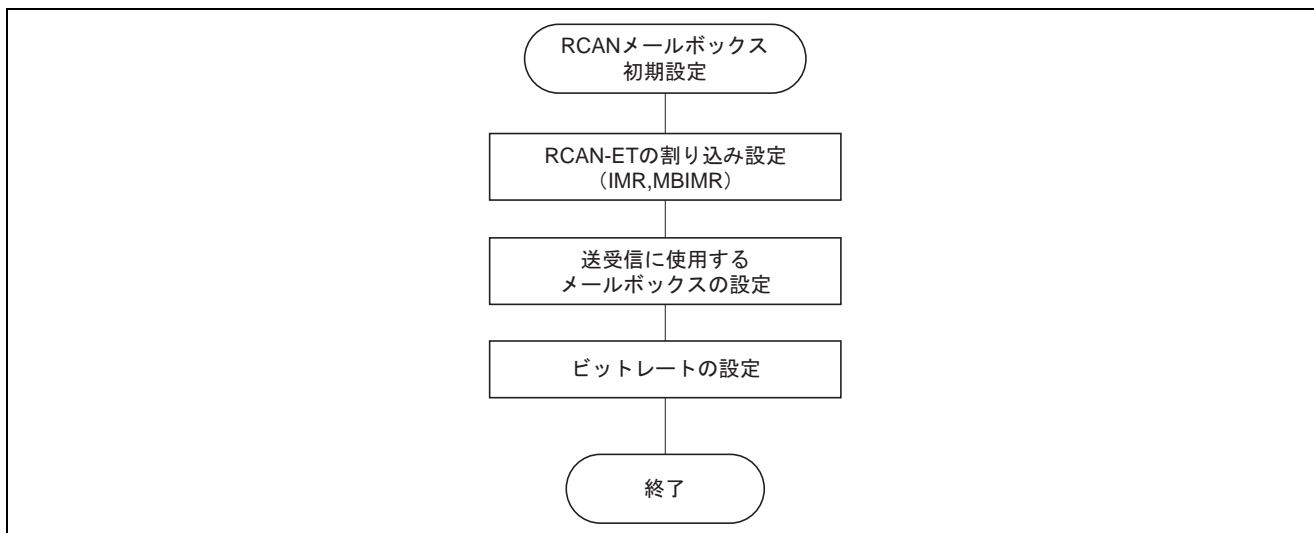
(1) H8SX/1544 初期設定ルーチン



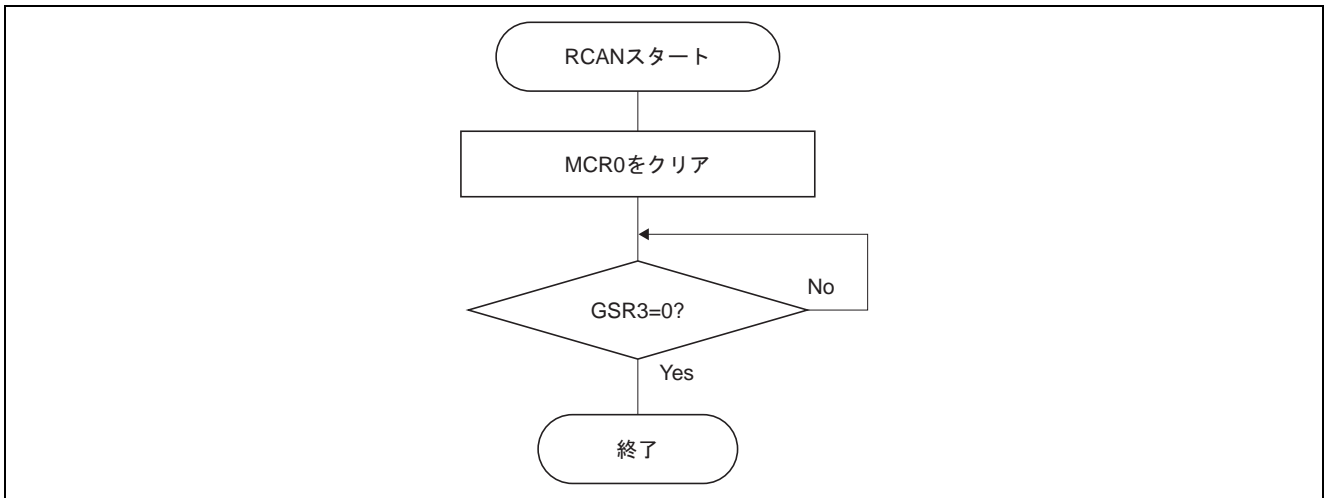
(2) RCAN 初期設定ルーチン



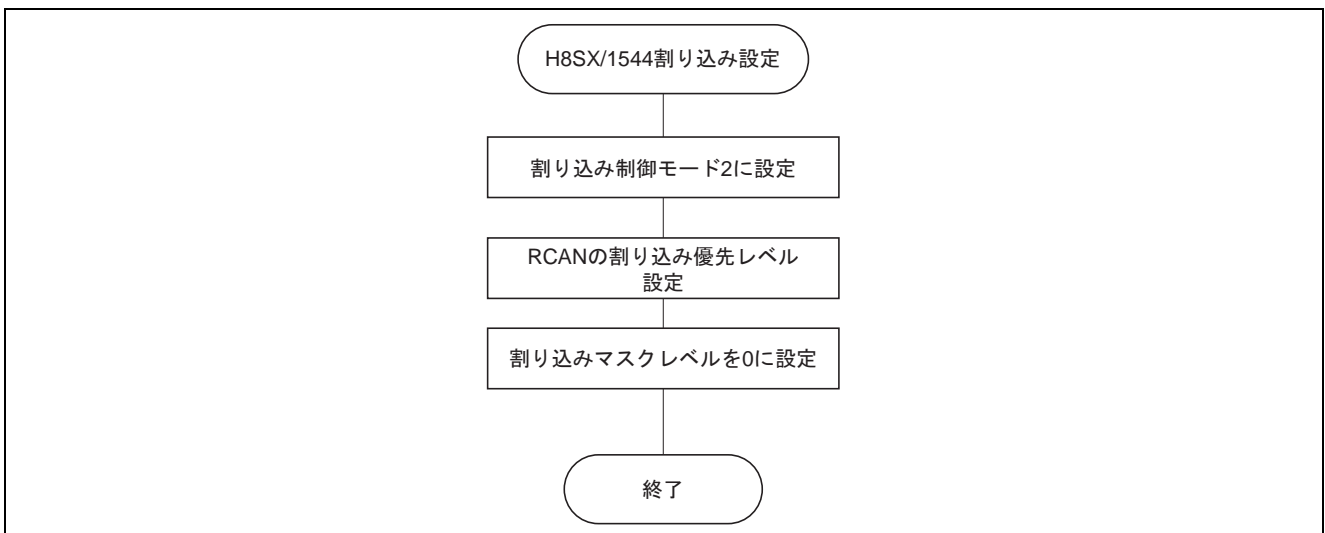
(3) RCAN メールボックス初期設定ルーチン



(4) RCAN スタートルーチン



(5) H8SX/1544 割り込み設定ルーチン





### 2.1.4 プログラムリスト

#### (1) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose      :   for H8SX/1544  RCAN-ET               */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();          /* H8SX/1544 初期化          */
    set_RCAN0_init();        /* RCAN 初期化              */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定  */

    while(1);

}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;      /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;     /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;     /* 外部ハッククロック(5MHz×4=20MHz)   */
    SBYCR.BIT.SSBY = 1;    /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定     */

}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

### (2) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include "iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件: 1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え: MCR15=1(初期値) に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックスインプティ割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554; /* ID:H'555,スタンダードフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;   /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x01;   /* データ長:1byte */
    RCANET0.MB[1].MSG_DATA[0] = 0xAA;    /* 送信データ:0xAA */
}

```

```

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /*TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/* RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}
/*****

```

## 2.2 データフレーム送受信 (スタンダード ID)

### 2.2.1 仕様

図 2.2.1 に示すように、ノード A からスタンダード ID : H'555 のデータフレームを送信し、ノード B で受信します (割り込みは使用しません)。

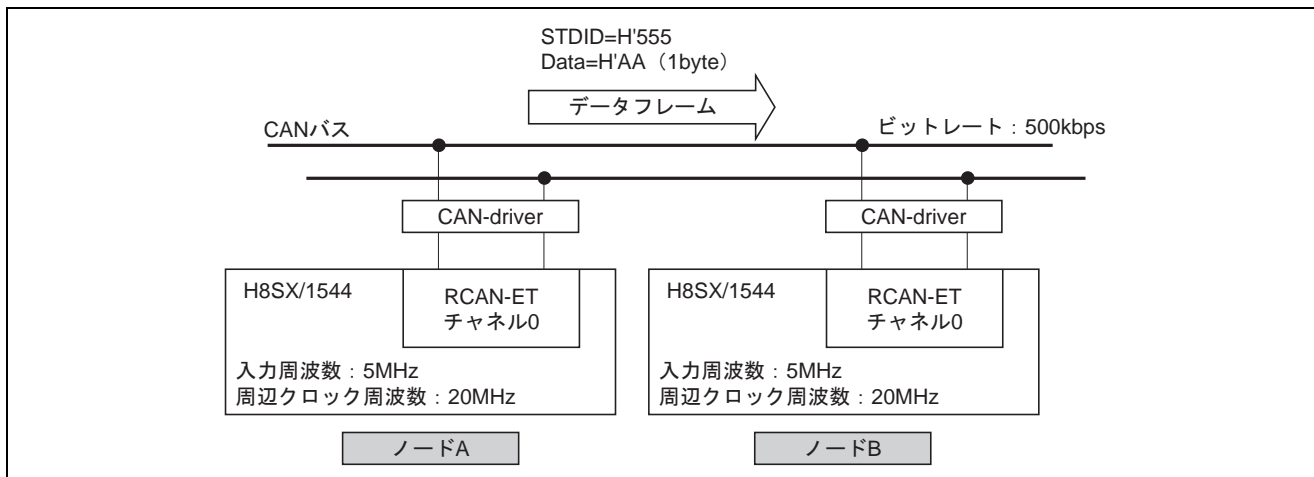


図 2.2.1 通信仕様

## 2.2.2 ソフトウェア説明

## (1) モジュール説明

表 2.2.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス個別
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信、送信完了フラグクリア。	
CAN メッセージ受信	RCAN0_Rx	CAN メッセージの受信、受信完了フラグクリア。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	-

## (2) 使用レジスタ説明

## (a) 送信側

表 2.2.2 使用レジスタ説明（送信側）

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、ID (H'555) を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x01	データ長 (1byte) を設定。
	RCANET0.MB[1].MSG_DATA[0]	0xAA	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。
	RCANET0.TXACK0.WORD	-	メールボックス 1 の送信完了待ち。 ( TXACK0 の bit1 が 1 にセットされるまでフラグポーリング )
	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

## (b) 受信側

表 2.2.3 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.MB[0].CTRL0.WORD.H	0x1554	メールボックス 0 をスタンダードフォーマット、データフレームに設定。また、ID (H'555) を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID, IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
CAN メッセージ 受信	RCANET0.RXPR0.WORD	-	メールボックス 0 の受信完了待ち。 (RXPR0 の bit0 が 1 にセットされるまでフラグポーリング)
	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

## (3) 使用 RAM 説明

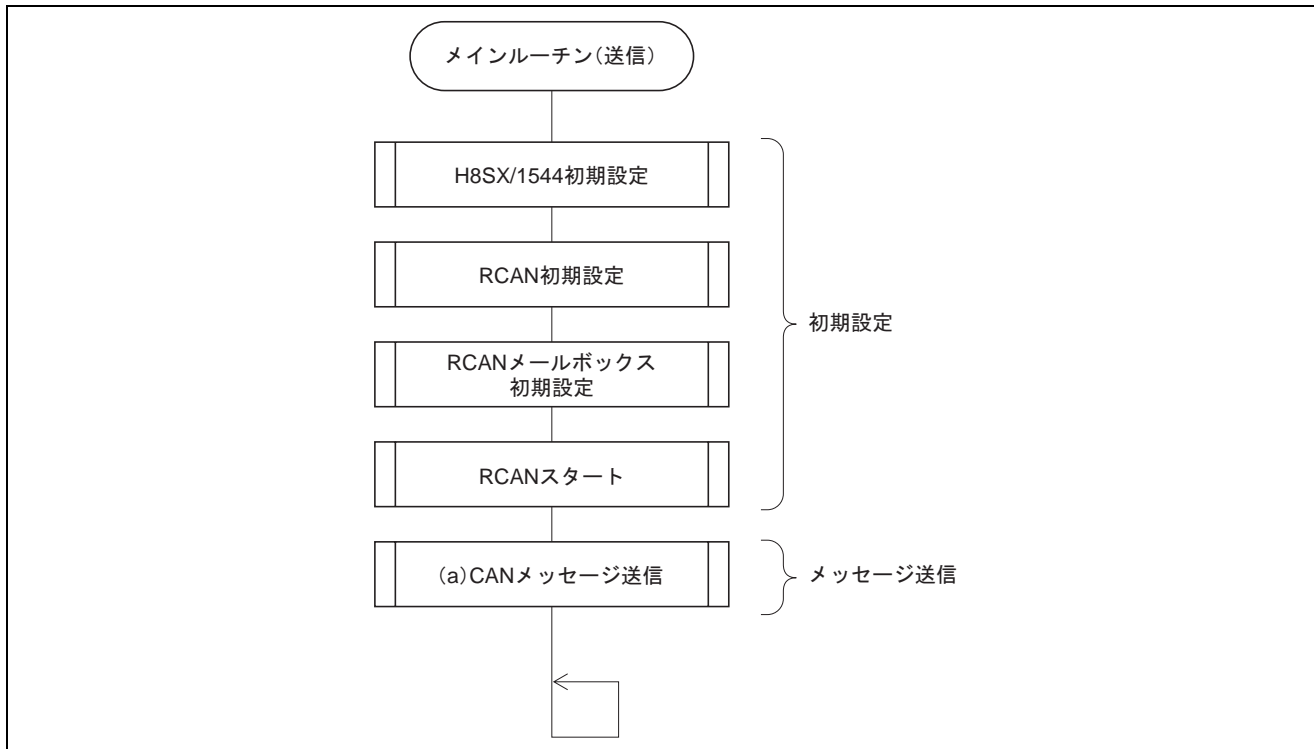
表 2.2.4 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化	MBbuff.ID.LONG	受信 ID を格納します。
CAN メッセージ受信	MBbuff.ID.WORD.H	
	MBbuff.ID.WORD.L	
	MBbuff.DATA.LONG[0] ~ [1]	受信データを格納します。
	MBbuff.DATA.WORD[0] ~ [3]	
MBbuff.DATA.BYTE[0] ~ [7]		

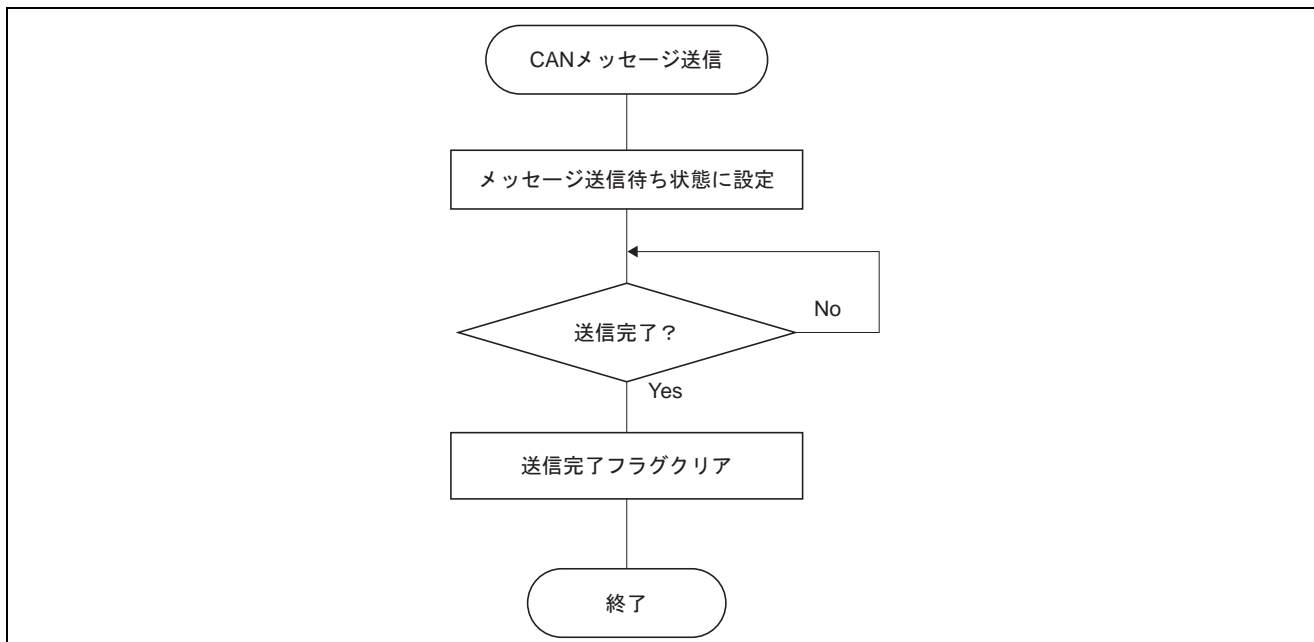
### 2.2.3 フローチャート

#### (1) 送信側フローチャート

##### メインルーチン



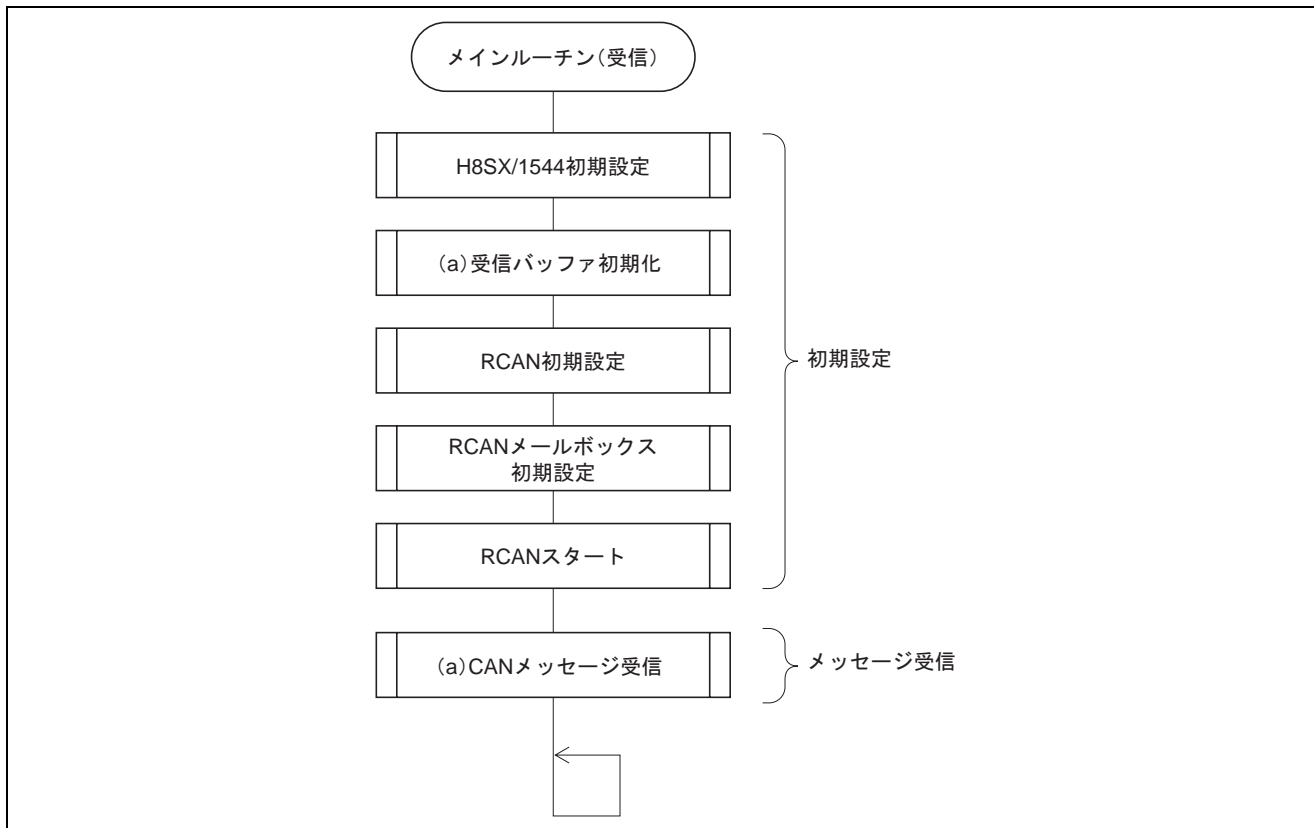
#### (a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



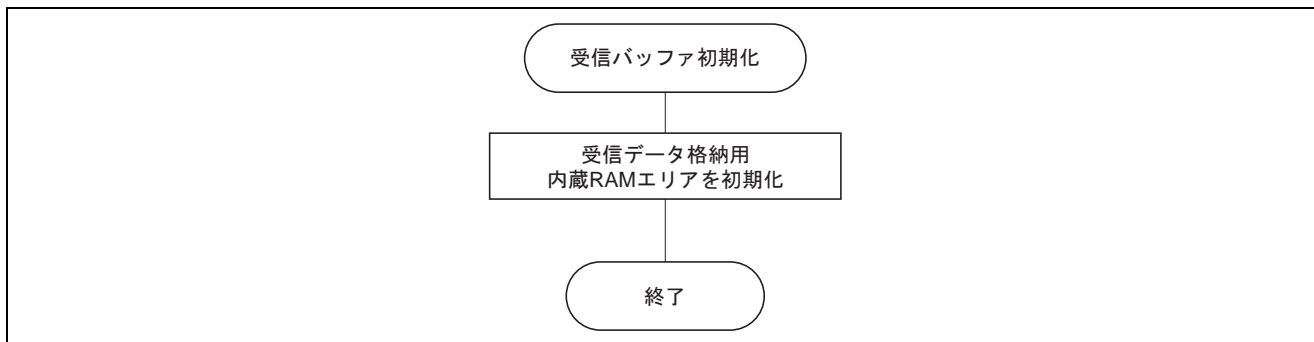


(2) 受信側フローチャート

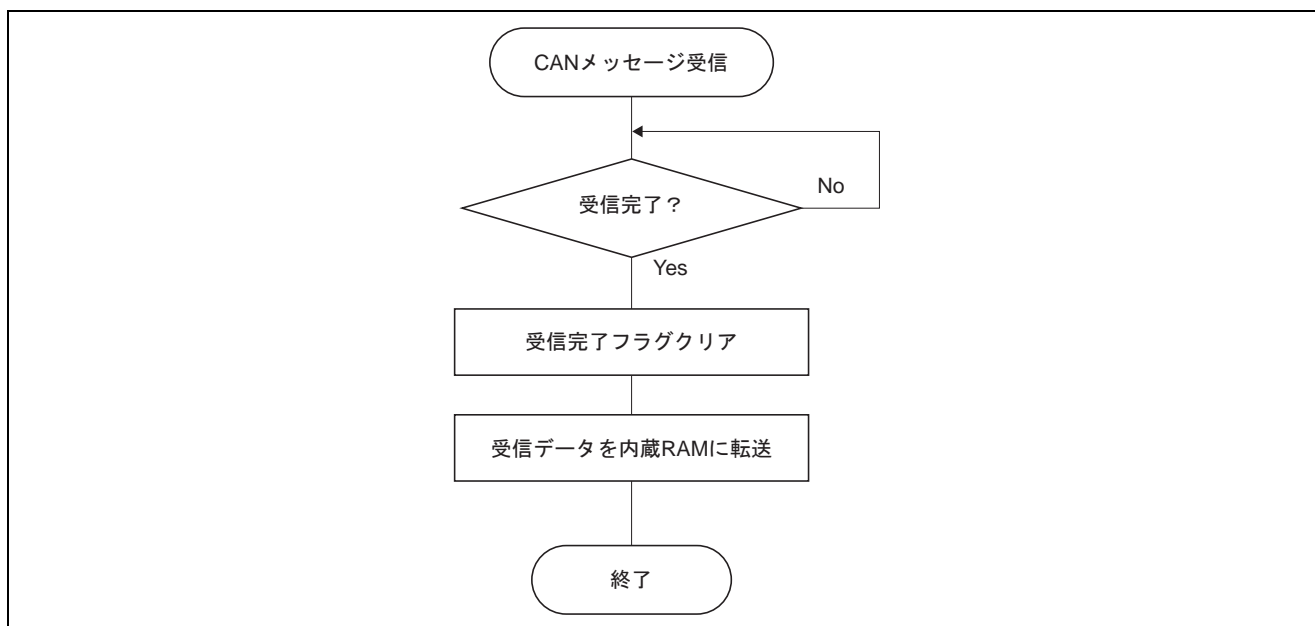
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)



(b) CAN メッセージ受信ルーチン (RCAN-ET 共通設定)



## 2.2.4 プログラムリスト

### (1) 送信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();          /* H8SX/1544 初期化          */
    set_RCAN0_init();        /* RCAN 初期化              */
    set_RCAN0MB_init();
    set_RCAN0_start();
    RCAN0_Tx();              /* CANメッセージ送信        */
    while(1);

}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;      /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;      /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;      /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;     /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定       */

}
/*****

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM I/O)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554; /* ID:H'555,スタンダートフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;    /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x01;    /* データ長:1byte */
    RCANET0.MB[1].MSG_DATA[0] = 0xAA;     /* 送信データ:0xAA */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /*TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
    RCANET0.BCR0.WORD = 0x0001;          /* BRP=1 */
}

```

```

/*****
/*      RCAN start routine
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE;          /* MCR0 クリア          */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?          */

}

/*****
/*      RCAN send message routine
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002;      /* MB1 を送信待ち状態に設定      */
    while ((RCANET0.TXACK0.WORD & 0x0002) != 0x0002); /* 送信完了待ち          */
    RCANET0.TXACK0.WORD = RCANET0.TXACK0.WORD; /* 送信完了フラグクリア(クリア条件:1ライト) */

}
/*****

```

## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c
/* Written       :   '06/06/01   REV.1.00
/* Purpose       :   for H8SX/1544   RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
extern void RCAN0_Rx(void);
/*****
/*      Main program
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMエリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    RCAN0_Rx();              /* CANメッセージ受信         */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定     */
}
/*****

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Rx(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    RCANET0.MB[0].CTRL0.WORD.H = 0x1554;      /* ID:H'555,スタンダードフォーマット,データレーム */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000;      /* STD_LAFM, IDE_LAFM の設定      */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;      /* メールボックス0を受信用に設定      */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001;      /* BRP=1      */

}

/*****
/*      RCAN start routine      */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;      /* MCR0 クリア      */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?      */

}

/*****
/*      RCAN receive message routine      */
/*****
void RCAN0_Rx(void){

    while ((RCANET0.RXPR0.WORD & 0x0001) != 0x0001); /* 受信完了待ち      */
    RCANET0.RXPR0.WORD = RCANET0.RXPR0.WORD; /* 受信完了フラグクリア(クリア条件:1バイト) */
    MBbuff.DATA.BYTE[0] = RCANET0.MB[0].MSG_DATA[0]; /* 受信データをRAMに格納      */

}

/*****
/*      RAM area Initialize routine      */
/*****
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;      /* データ格納用RAMクリア初期化      */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;

}
/*****

```



2.3 データフレーム送受信 (エクステンデッド ID)

2.3.1 仕様

図 2.3.1 に示すように、ノード A からスタンダード ID : H'555, エクステンデッド ID : H'2AAAA のデータフレームを送信し、ノード B で受信します (割り込みを使用します)。

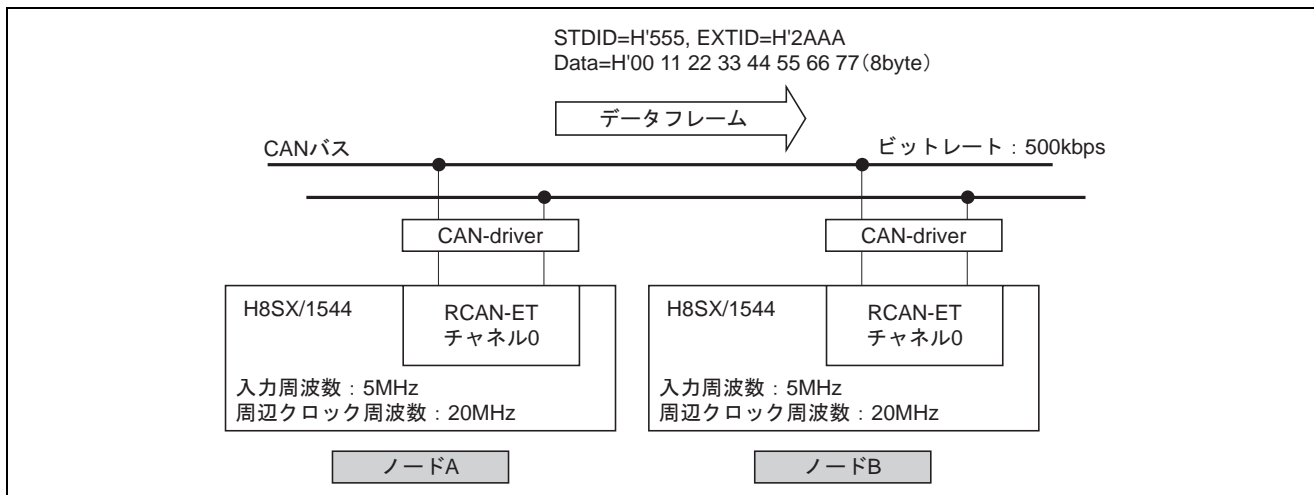


図 2.3.1 通信仕様

## 2.3.2 ソフトウェア説明

### (1) モジュール説明

表 2.3.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
RCAN-ET 共通			
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
データフレーム 受信割り込み	RM0_0	CAN メッセージの受信完了フラグクリア。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	-

### (2) 使用レジスタ説明

#### (a) 送信側

表 2.3.2 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み (IRR8) を許可に設定。 (IMR8=0)
	RCANET0.MBIMR0.WORD	0xFFFD	メールボックス 1 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x9556	メールボックス 1 をエクステンデッドフォーマット、データフレームに設定。 また、スタンダード ID (H'555) とエクステンデッド ID (H'2AAAA) を設定。
	RCANET0.MB[1].CTRL0.WORD.L	0xAAAA	
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0]	0x00	送信データを設定。
	RCANET0.MB[1].MSG_DATA[1]	0x11	
RCANET0.MB[1].MSG_DATA[2]	0x22		
RCANET0.MB[1].MSG_DATA[3]	0x33		

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[1].MSG_DATA[4]	0x44	送信データを設定。
	RCANET0.MB[1].MSG_DATA[5]	0x55	
	RCANET0.MB[1].MSG_DATA[6]	0x66	
	RCANET0.MB[1].MSG_DATA[7]	0x77	
	RCANET0.BCR1.WORD	0x4300	P $\phi$ =20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。
メールボックス エンプティ割り込み	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

(b) 受信側

表 2.3.3 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFFFFD	データフレーム受信割り込み (IRR1) を許可に設定。(IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFFE	メールボックス 0 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	0x9556	メールボックス 0 をエクステンデッドフォーマット、データフレームに設定。また、スタンダード ID (H'555) とエクステンデッド ID (H'2AAAA) を設定。
	RCANET0.MB[0].CTRL0.WORD.L	0xAAAA	
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID、エクステンデッド ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].LAFM.WORD.L	0x0000	
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	P $\phi$ =20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq),SJW= 0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
データフレーム 受信割り込み	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

## (3) 使用 RAM 説明

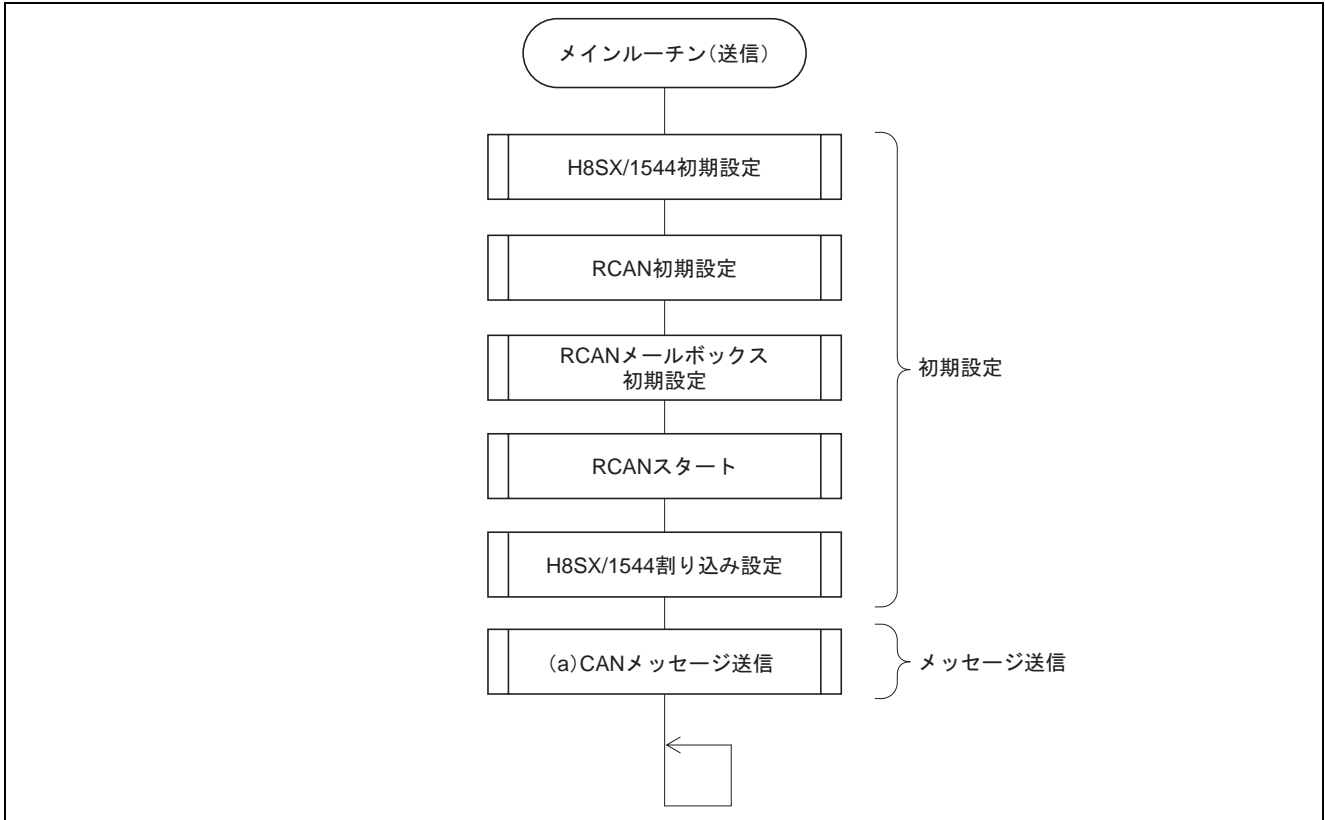
表 2.3.4 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化	MBbuff.ID.LONG MBbuff.ID.WORD.H MBbuff.ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuff.DATA.LONG[0] ~ [1] MBbuff.DATA.WORD[0] ~ [3] MBbuff.DATA.BYTE[0] ~ [7]	受信データを格納します。

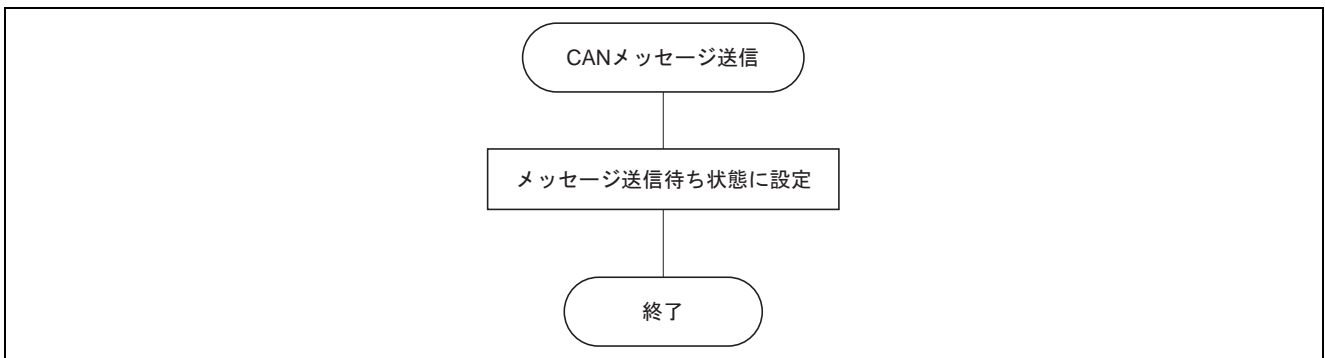
2.3.3 フローチャート

(1) 送信側フローチャート

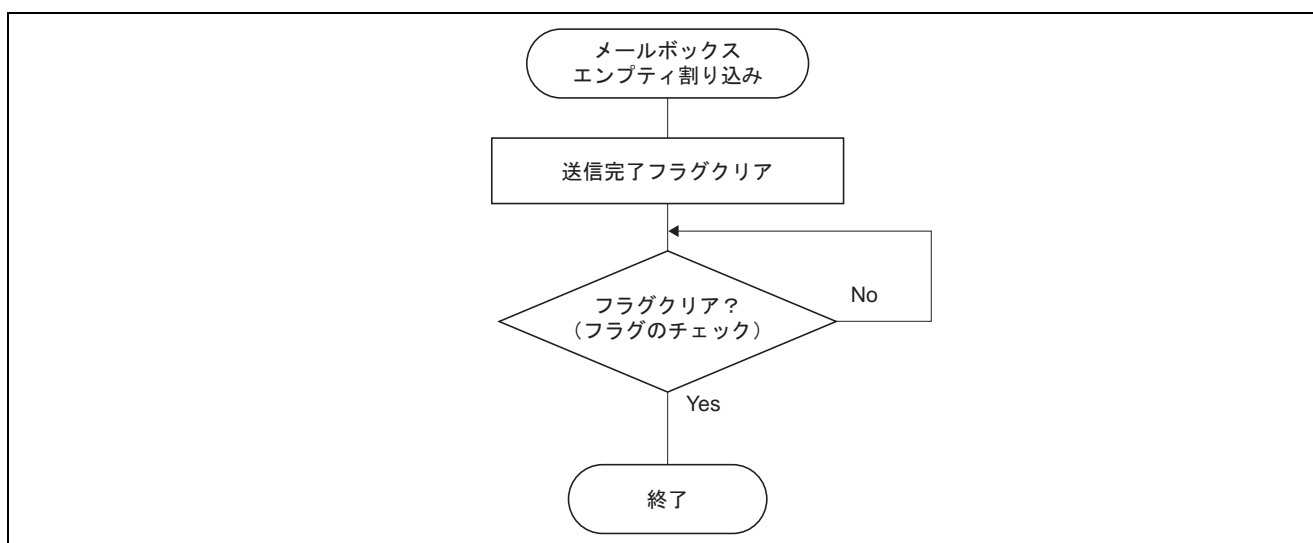
メインルーチン



(a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



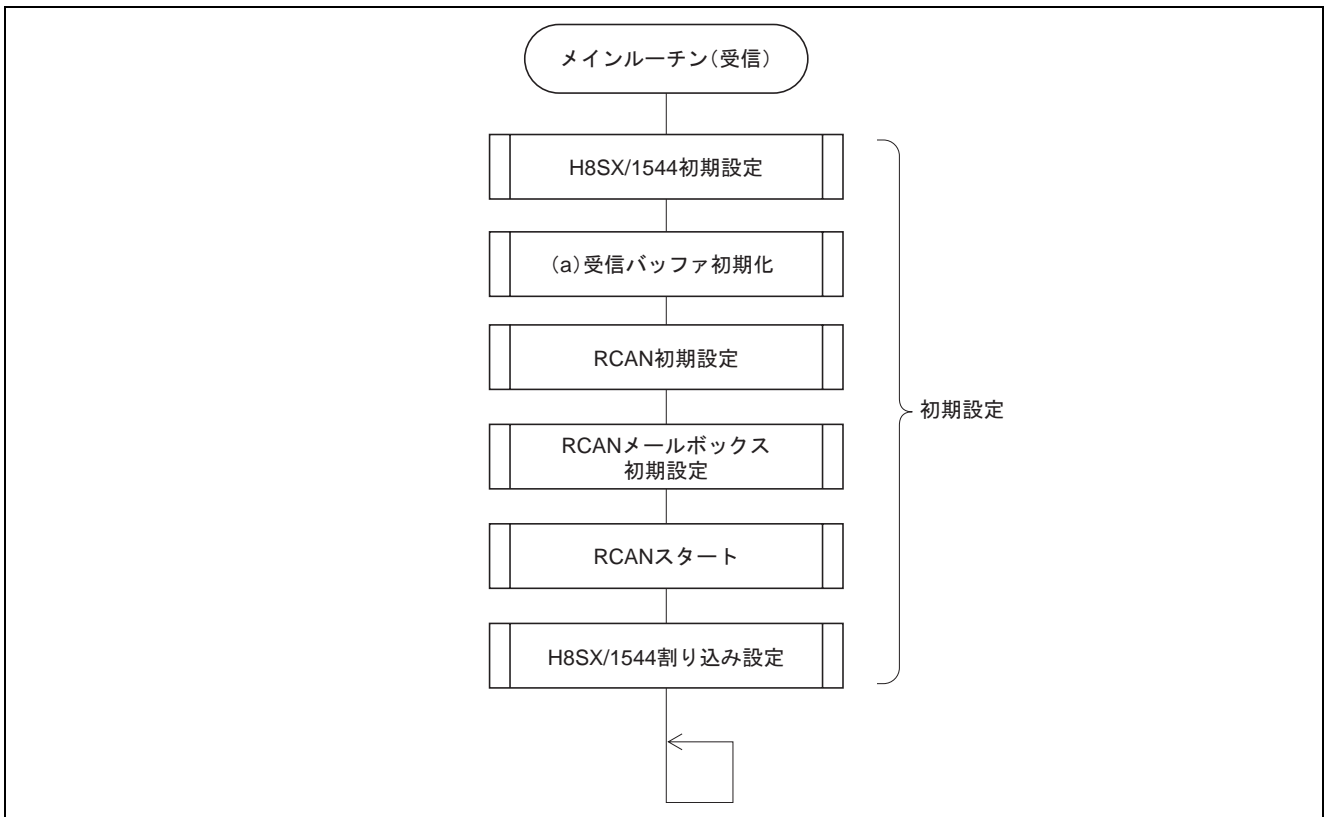
(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



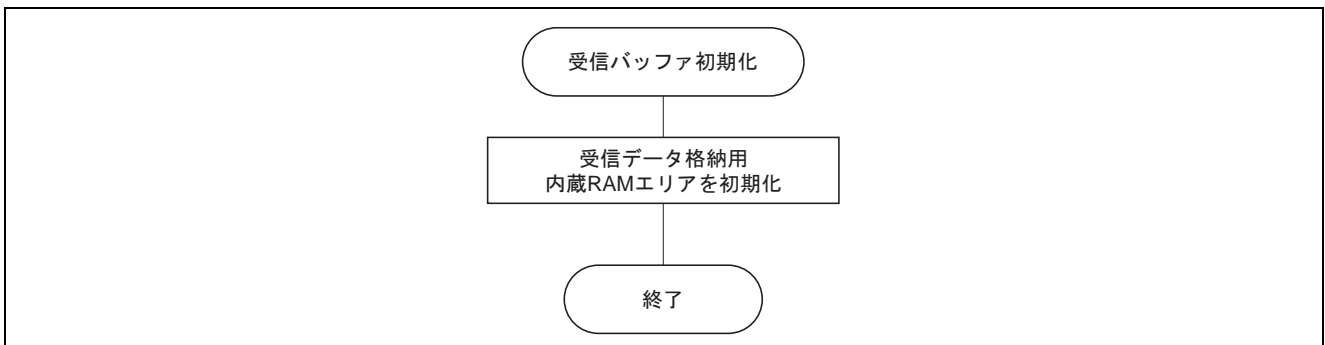
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

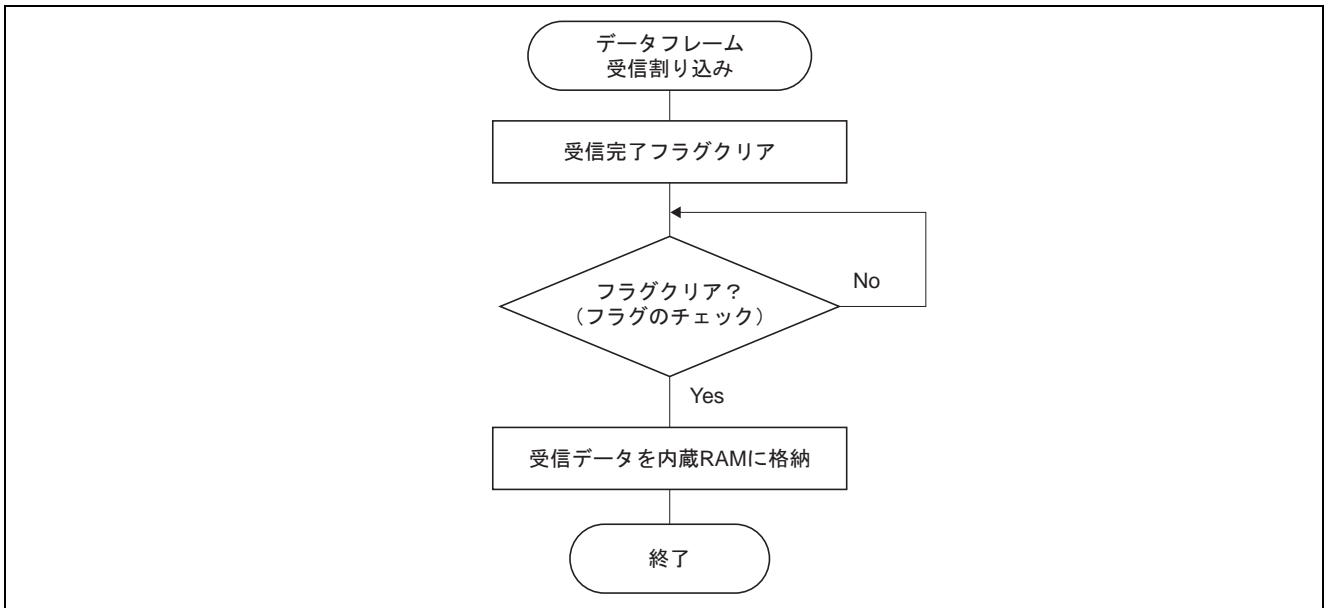
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)



(b) データフレーム受信割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。



## 2.3.4 プログラムリスト

### (1) 送信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定   */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);

}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;        /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;        /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;       /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */

}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;                /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;            /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);                     /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM I/O)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックスエンティティ割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

```

```

RCANET0.MB[1].CTRL0.WORD.H = 0x9556; /*STDID:H'555,拡張データフォーマット,データレム*/
RCANET0.MB[1].CTRL0.WORD.L = 0xAAAA; /* EXTID:H'2AAAA */
RCANET0.MB[1].CTRL1.BYTE.H = 0x00; /* メールボックス1を送信用に設定 */
RCANET0.MB[1].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
RCANET0.MB[1].MSG_DATA[0] = 0x00; /* 送信データ:0x00 */
RCANET0.MB[1].MSG_DATA[1] = 0x11; /* 送信データ:0x11 */
RCANET0.MB[1].MSG_DATA[2] = 0x22; /* 送信データ:0x22 */
RCANET0.MB[1].MSG_DATA[3] = 0x33; /* 送信データ:0x33 */
RCANET0.MB[1].MSG_DATA[4] = 0x44; /* 送信データ:0x44 */
RCANET0.MB[1].MSG_DATA[5] = 0x55; /* 送信データ:0x55 */
RCANET0.MB[1].MSG_DATA[6] = 0x66; /* 送信データ:0x66 */
RCANET0.MB[1].MSG_DATA[7] = 0x77; /* 送信データ:0x77 */

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/* RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}
/*****
/* RCAN send message routine */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002; /* MB1を送信待ち状態に設定 */

}

/*****
/* Mailbox Empty Interrupt routine */
/*****
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{

    RCANET0.TXACK0.WORD = 0x0002; /* 送信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.TXACK0.WORD & 0x0002); /* フラグのチェック */

}
/*****

```

## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMクリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    while(1);

}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */

}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/リト/スリプ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え: MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}

```

```

/*****/
/*      RCAN Mailbox Initialize routine      */
/*****/
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFFFD;          /* データレム受信割り込み許可      */
    RCANET0.MBIMR0.WORD &= 0xFFFE;      /* MB0 割り込み許可                */

    RCANET0.MB[0].CTRL0.WORD.H = 0x9556; /* STDID:H'555,イクステンツットフォーマット,データレム */
    RCANET0.MB[0].CTRL0.WORD.L = 0xAAAA; /* EXTID:H'2AAAA                */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000; /* STD_LAFM, IDE_LAFM の設定      */
    RCANET0.MB[0].LAFM.WORD.L = 0x0000; /* EXTID_LAFM の設定              */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02; /* マルチックス0を受信用に設定   */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1                          */
}
/*****/
/*      RCAN start routine                  */
/*****/
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0 クリア                    */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?                        */
}
/*****/
/*      RAM area Initialize routine        */
/*****/
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;                  /* データ格納用RAMクリア初期化    */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;
}
/*****/
/*      Data Frame Received Interrupt routine(MailBox0)      */
/*****/
#pragma interrupt(RM0_0)
void RM0_0(void)
{
    unsigned int i;

    RCANET0.RXPR0.WORD = 0x0001;        /* 受信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.RXPR0.WORD & 0x0001); /* フラグのチェック                */
    for(i = 0;i < 8;i++){
        MBbuff.DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i]; /* 受信データをRAMに格納 */
    }
}
/*****/

```



## 2.4 リモートフレーム送受信 (ATX 使用)

### 2.4.1 仕様

図 2.4.1 に示すように、ノード A からスタンダード ID : H'555 のリモートフレームを送信します。ノード B は、リモートフレームを受信後、データフレーム自動送信機能 (ATX) を使用して、スタンダード ID : H'555 のデータフレームを自動送信します。

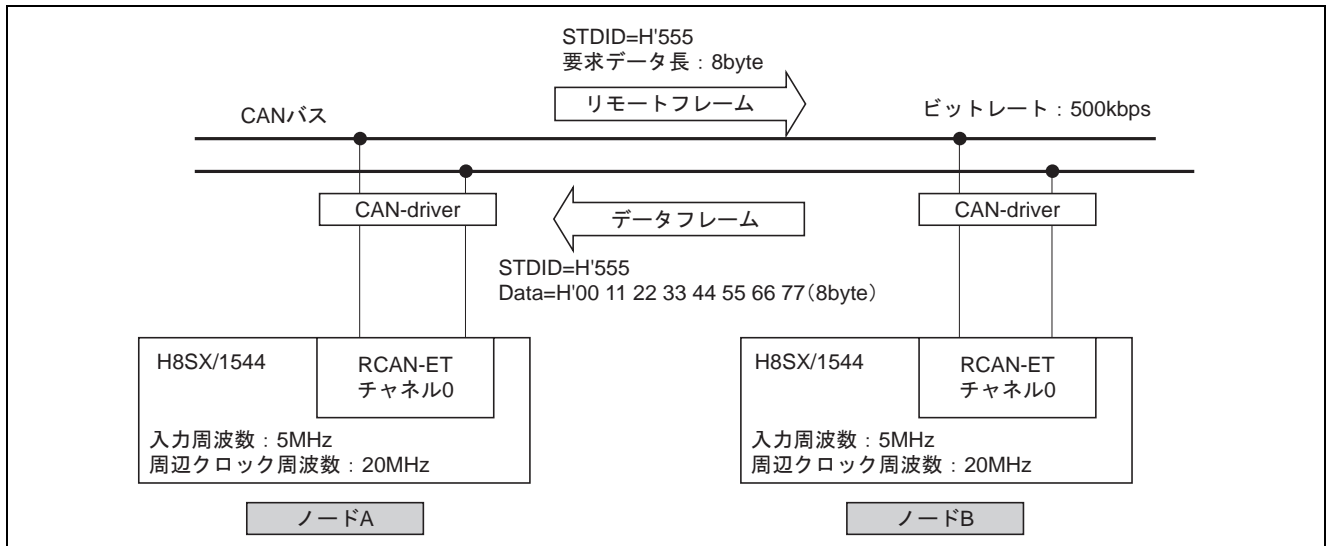


図 2.4.1 通信仕様

### 2.4.2 ソフトウェア説明

#### (1) モジュール説明

表 2.4.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信	RCAN0_Tx	リモートフレームの送信。	
メールボックス エンプティ割り込み	SLE0_0	リモートフレームの送信完了フラグクリア。	
データフレーム 受信割り込み	RM0_0	メールボックス0のデータフレームの受信完了フラグクリア。受信データの格納。	
リモートフレーム 受信割り込み	RM1_0	メールボックス1のリモートフレームの受信完了フラグクリア、および、データフレーム送信完了フラグクリア。	
受信バッファ初期化	Clear_MBbuff	CANメッセージ格納用内蔵RAMエリアの初期化。	-

【注】 H8SX/1544 は、メールボックス 0 以外のメールボックスは、すべての割り込み要因 (RM1,SLE,OV,ERS) に対して、共通の割り込みベクタアドレスを使用します。したがって、本動作例 (メールボックス 1 使用) では、リモートフレーム受信割り込み (IRR2) とデータフレーム送信完了割り込み (IRR8) を、同じ割り込み関数で処理します。

#### (2) 使用レジスタ説明

##### (a) 送信側

表 2.4.2 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFD	メールボックスエンプティ割り込み (IRR8) およびデータフレーム受信割り込み (IRR1) を許可に設定。(IMR8=0,IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFC	メールボックス0,1を割り込み許可に設定
	RCANET0.MB[0].CTRL0.WORD.H	0x1554	メールボックス0をスタンダードフォーマット、データフレーム (RTR=0)に設定。また、スタンダード ID (H'555) を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス0のスタンダード ID および IDE ビットのフィルタマスクを設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 をデータフレーム受信用に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x5554	メールボックス 1 をスタンダードフォーマット、リモートフレーム (RTR=1) に設定。また、スタンダード ID (H'555) を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 をリモートフレーム送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。
メールボックス エンプティ割り込み	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)
データフレーム 受信割り込み	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

(b) 受信側

表 2.4.3 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFB	メールボックスエンプティ割り込み (IRR8) およびリモートフレーム受信割り込み (IRR2) を許可に設定。 (IMR8=0, IMR2=0)
	RCANET0.MBIMR0.WORD	0xFFFFD	メールボックス 1 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス 0 をスタンダードフォーマット、データフレーム (RTR=0) に設定。 また、スタンダード ID (H'555) を設定。 データフレーム自動送信機能 (ATX) 使用時は、リモートフレーム受信後も RTR はセットされません。
	RCANET0.MB[1].LAFM.WORD.H	0x0000	メールボックス 1 のスタンダード ID、IDE ビットのフィルタマスクを設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[1].CTRL1.BYTE.H	0x11	メールボックス1をリモートフレーム受信可、データフレーム送信可に設定。また、データフレーム自動送信機能(ATX)を設定。(ATX=1,MBC=b'001)
	RCANET0.MB[1].MSG_DATA[0]	0x00	送信データを設定。
	RCANET0.MB[1].MSG_DATA[1]	0x11	
	RCANET0.MB[1].MSG_DATA[2]	0x22	
	RCANET0.MB[1].MSG_DATA[3]	0x33	
	RCANET0.MB[1].MSG_DATA[4]	0x44	
	RCANET0.MB[1].MSG_DATA[5]	0x55	
	RCANET0.MB[1].MSG_DATA[6]	0x66	
	RCANET0.MB[1].MSG_DATA[7]	0x77	
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz時500kbpsに設定。 (TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1)
RCANET0.BCR0.WORD	0x0001		
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
リモートフレーム 受信割り込み	RCANET0.RFPR0.WORD	0x0002	メールボックス1のリモートフレーム受信完了フラグのクリア。 (クリア条件:1ライト)
	RCANET0.TXACK0.WORD	0x0002	メールボックス1の送信完了フラグのクリア。(クリア条件:1ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

### (3) 使用 RAM 説明

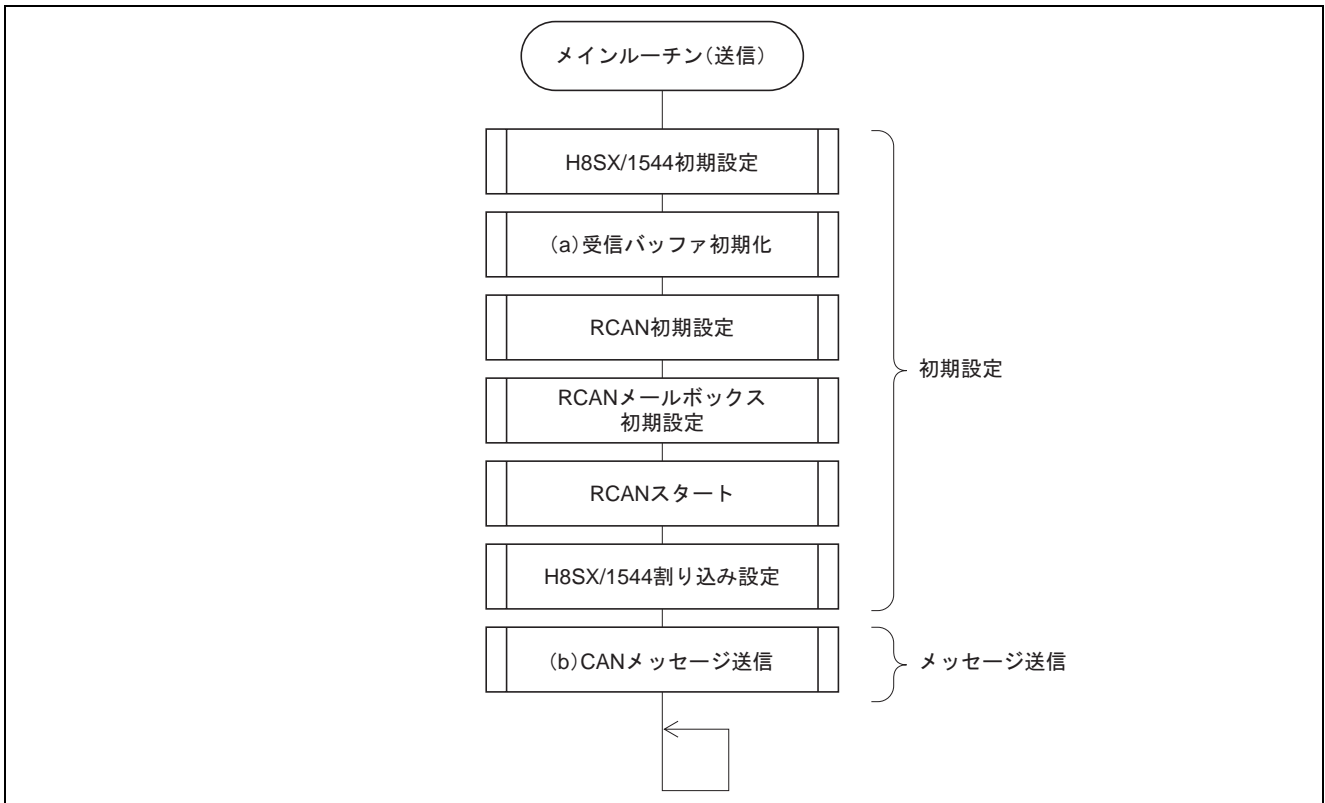
表 2.4.4 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化	MBbuff.ID.LONG MBbuff.ID.WORD.H MBbuff.ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuff.DATA.LONG[0] ~ [1] MBbuff.DATA.WORD[0] ~ [3] MBbuff.DATA.BYTE[0] ~ [7]	受信データを格納します。

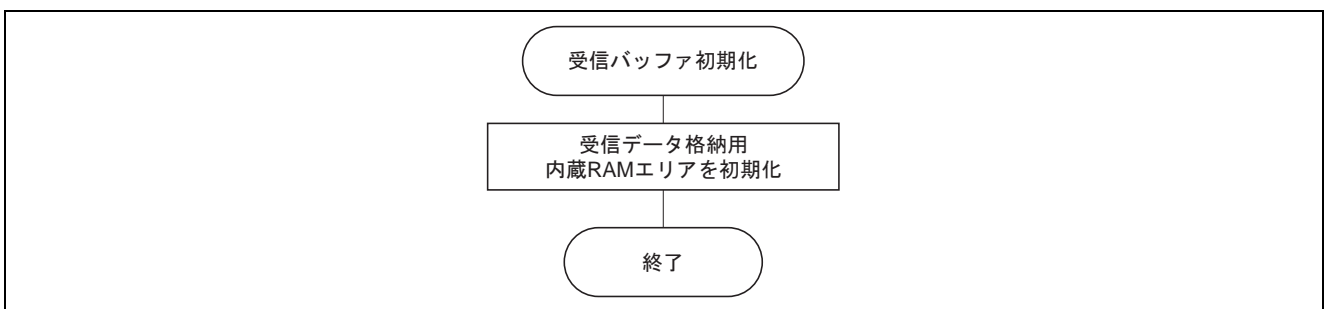
### 2.4.3 フローチャート

#### (1) 送信側フローチャート

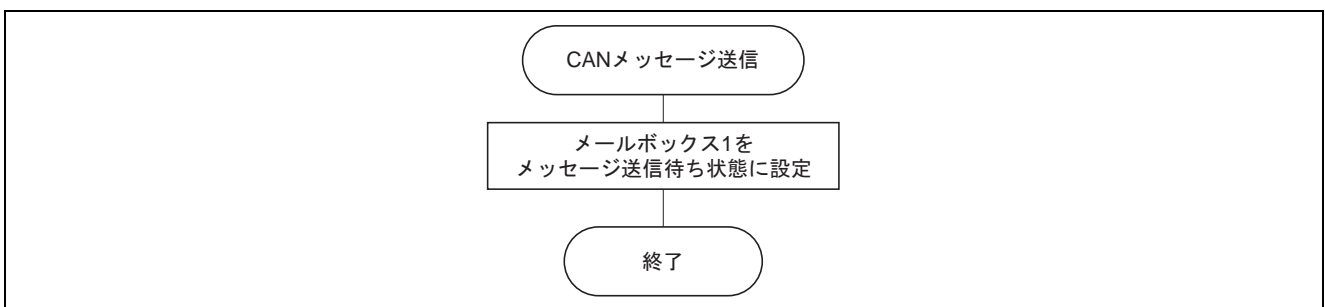
##### メインルーチン



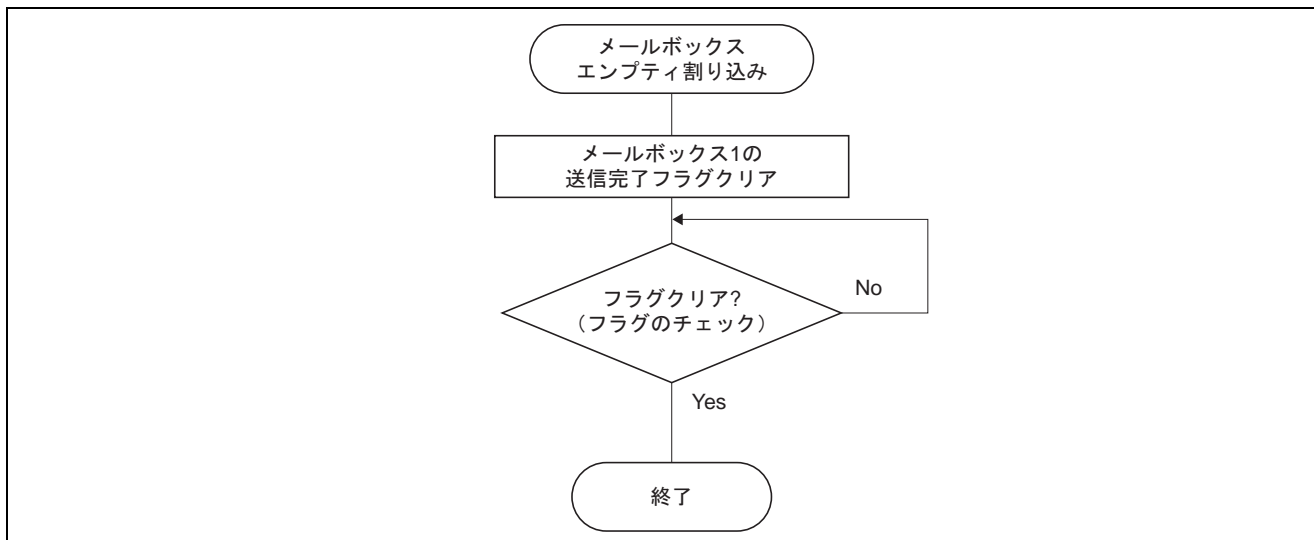
#### (a) 受信バッファ初期化ルーチン (デバイス個別設定)



#### (b) CANメッセージ送信ルーチン (RCAN-ET 共通設定)

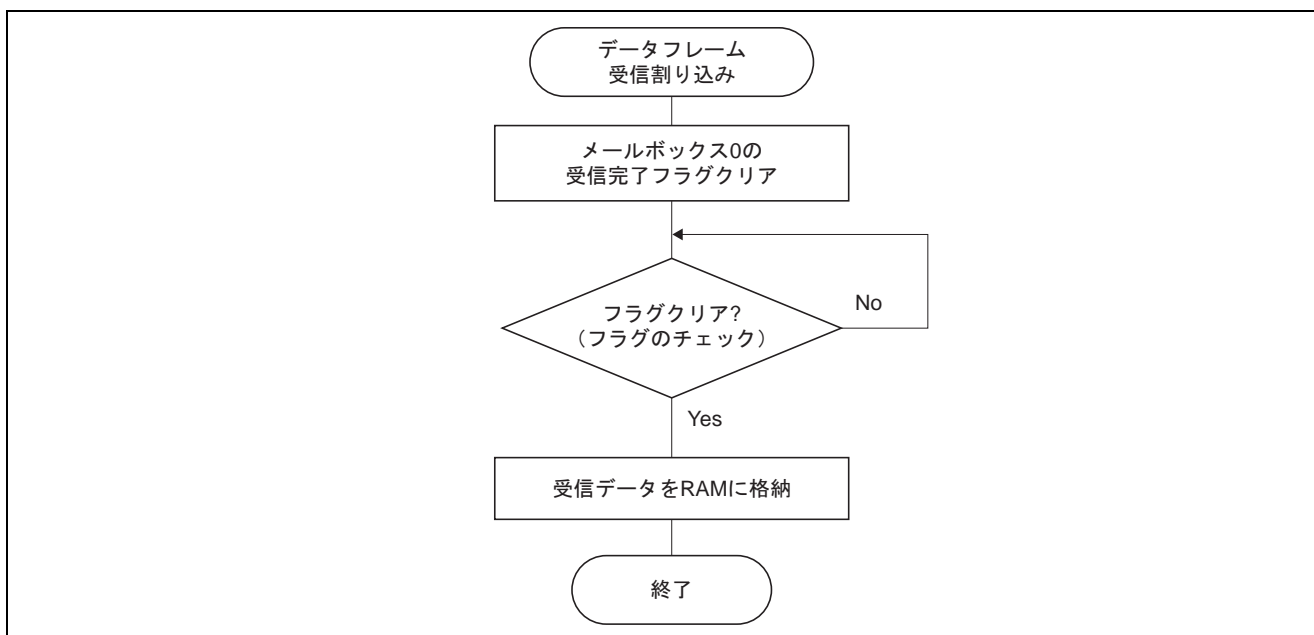


### (c) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

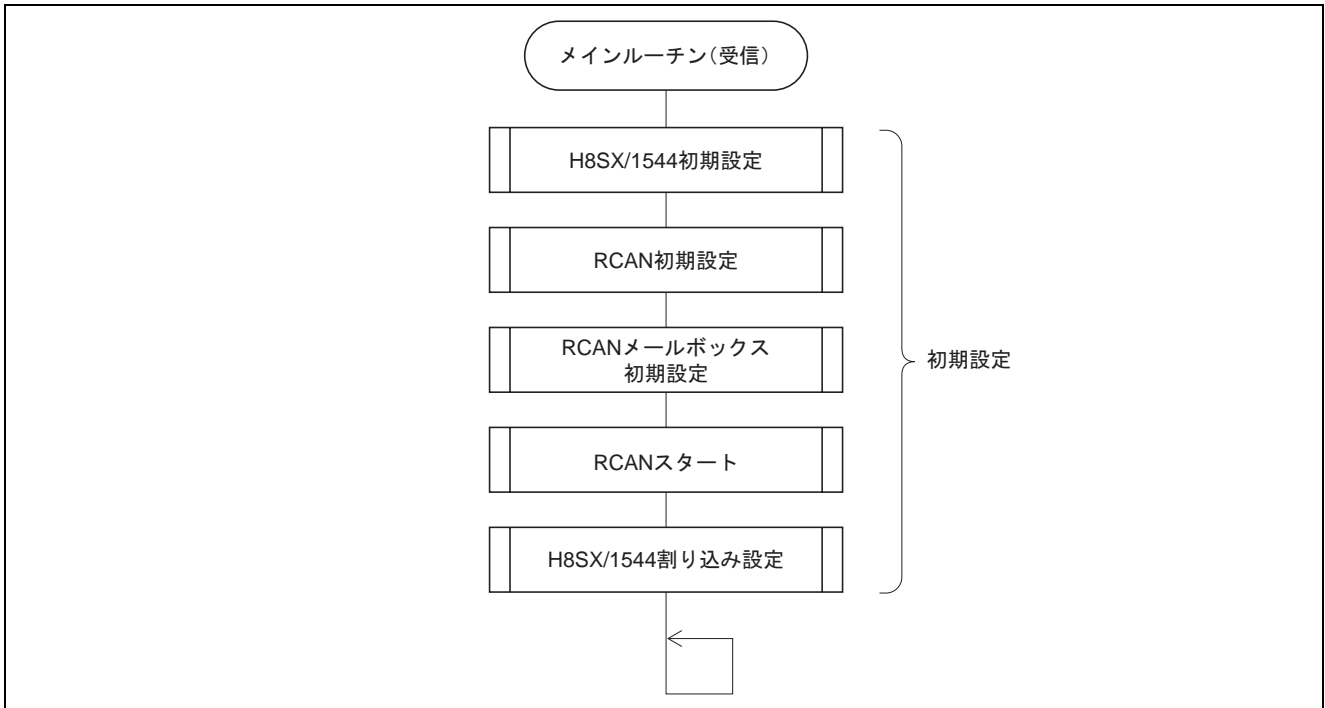
### (d) データフレーム受信割り込みルーチン (RCAN-ET 共通設定)



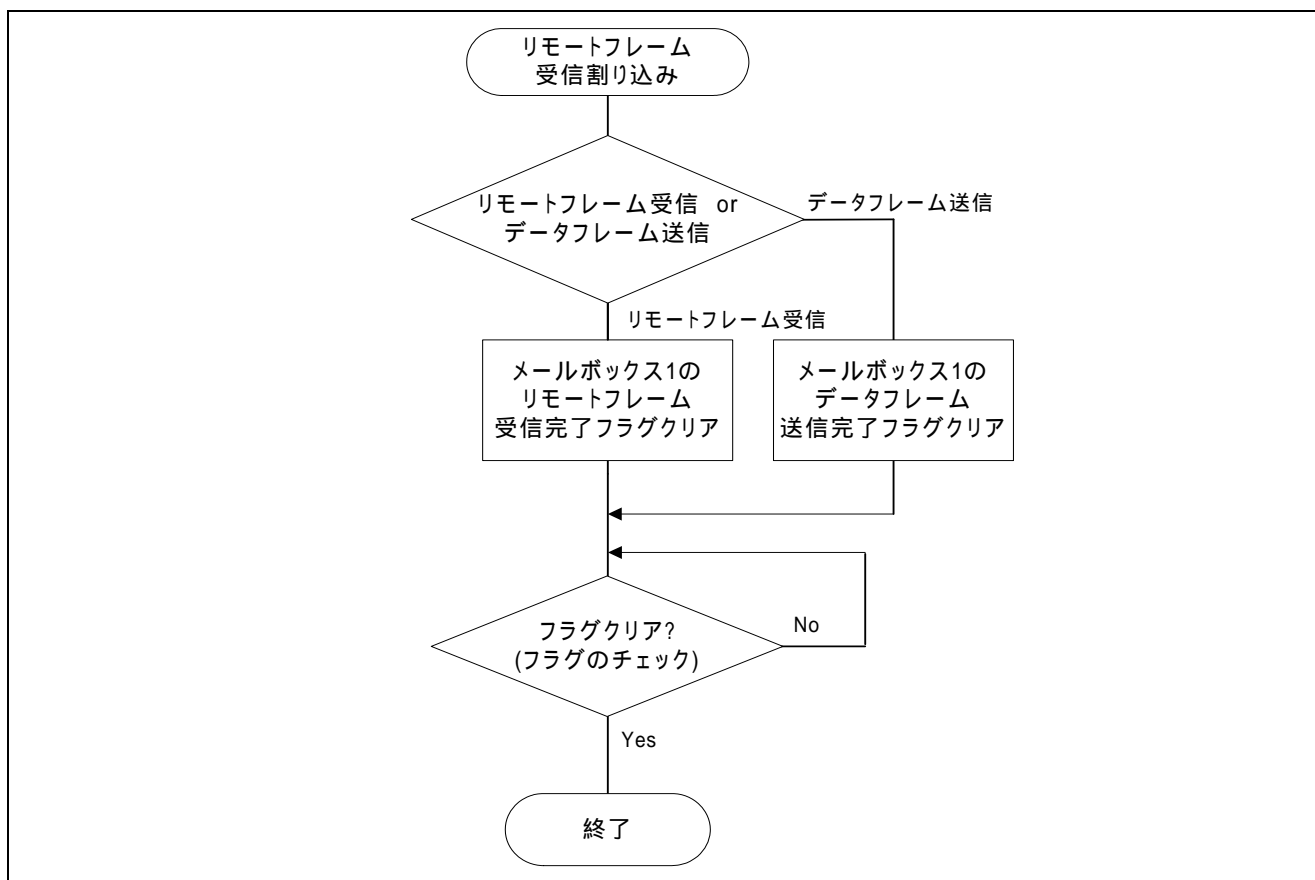
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

メインルーチン



(a) リモートフレーム受信割り込みルーチン (RCAN-ET 共通設定)



- 【注】
- H8SX/1544 は、メールボックス0以外のメールボックスは 全ての割り込み要因 (RM1, SLE, OVR, ERS) に対して、共通の割り込みベクタアドレスを使用します。したがって、本動作例 (メールボックス1 使用) では リモートフレーム受信割り込み (IRR2) とデータフレーム送信完了割り込み (IRR8) を、同じ割り込み関数で処理します。
  - クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。



## 2.4.4 プログラムリスト

### (1) 送信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMエリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);
}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定     */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFEFD;          /* メールボックスインテリゲント受信割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFC;      /* MB0-1 割り込み許可 */

    RCANET0.MB[0].CTRL0.WORD.H = 0x1554; /* STDID:H'555,スタンダードフォーマット,データフレーム(RTR=0) */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000; /* STD_LAFM, IDE_LAFM の設定 */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02; /* メールボックス0をデータフレーム受信用に設定 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x5554; /* STDID:H'555,スタンダードフォーマット,レトフレーム(RTR=1) */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00; /* メールボックス1をレトフレーム送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08; /* データ長:8byte */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/*      RCAN start routine      */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RCAN send message routine      */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002;      /* MB1 を送信待ち状態に設定 */

}

/*****
/*      RAM area Initialize routine      */
/*****
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;                  /* データ格納用 RAM 初期化 */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;

}

/*****
/*      Mailbox Empty Interrupt routine      */
/*****

```

```

/*****/
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{

    RCANET0.TXACK0.WORD = 0x0002;          /* 送信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.TXACK0.WORD & 0x0002);  /* フラグのチェック */

}

/*****/
/*      Data Frame Received Interrupt routine(MailBox0)      */
/*****/
#pragma interrupt(RM0_0)
void RM0_0(void)
{
    unsigned int i;

    RCANET0.RXPR0.WORD = 0x0001;          /* 受信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0001);  /* フラグのチェック */

    for(i = 0;i < 8;i++){
        MBbuff.DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i];/* 受信データをRAMに格納 */
    }

}
/*****/

```

## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose      :   for H8SX/1544  RCAN-ET                */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定   */
    while(1);

}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;        /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;        /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;       /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */

}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;                /* 割り込み制御モード 2      */
    INTC.IPRQ.BIT._RCAN01 = 7;             /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);                      /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メモリバス(RAM I/O)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFEFB;          /* メモリバスインプティ,リモートフレーム受信割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554; /* STDID:H'555,スタンダートフォーマット,RTR=0 */
    RCANET0.MB[1].LAFM.WORD.H = 0x0000; /* STD_LAFM,IDE_LAFM の設定 */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x11; /* ATX=1,MBC=B'001(リモートフレーム受信,データフレーム送信) */

```



```

RCANET0.MB[1].MSG_DATA[0] = 0x00; /* 送信データ:0x00 */
RCANET0.MB[1].MSG_DATA[1] = 0x11; /* 送信データ:0x11 */
RCANET0.MB[1].MSG_DATA[2] = 0x22; /* 送信データ:0x22 */
RCANET0.MB[1].MSG_DATA[3] = 0x33; /* 送信データ:0x33 */
RCANET0.MB[1].MSG_DATA[4] = 0x44; /* 送信データ:0x44 */
RCANET0.MB[1].MSG_DATA[5] = 0x55; /* 送信データ:0x55 */
RCANET0.MB[1].MSG_DATA[6] = 0x66; /* 送信データ:0x66 */
RCANET0.MB[1].MSG_DATA[7] = 0x77; /* 送信データ:0x77 */

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/* RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE; /* MCR0クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/* Remote Frame Received Interrupt routine(MailBox1~15) */
/*****
#pragma interrupt(RM1_0)
void RM1_0(void)
{
    unsigned int i;

    if(RCANET0.RFPR0.WORD){ /* IRR2セット,MB1受信 */
        RCANET0.RFPR0.WORD = 0x0002; /* リモートフレーム受信完了フラグクリア(クリア条件:1バイト) */
        while(RCANET0.RFPR0.WORD & 0x0002); /* フラグのチェック */
    }
    else if(RCANET0.TXACK0.WORD){ /* IRR8セット,MB1送信完了 */
        RCANET0.TXACK0.WORD = 0x0002; /* 送信完了フラグクリア(クリア条件:1バイト) */
        while(RCANET0.TXACK0.WORD & 0x0002); /* フラグのチェック */
    }

}

/*****

```

## 2.5 自動再送信無効機能 (DART)

### 2.5.1 仕様

図 2.5.1 に示すように、ノード A からノード B に対して、スタンダード ID : H'555 のデータフレームを送信します。ノード B が CAN バスに未接続の場合、ACK を送信するノードがないため、ノード A は ACK を受信するまで自動的にメッセージの再送信を繰り返します。ノード A 側で再送信無効機能 (DART) を設定した場合、メッセージは再送信されず、送信キャンセルされます。

【注】本動作例では、ノード A 側 (送信側) の動作例について説明します。

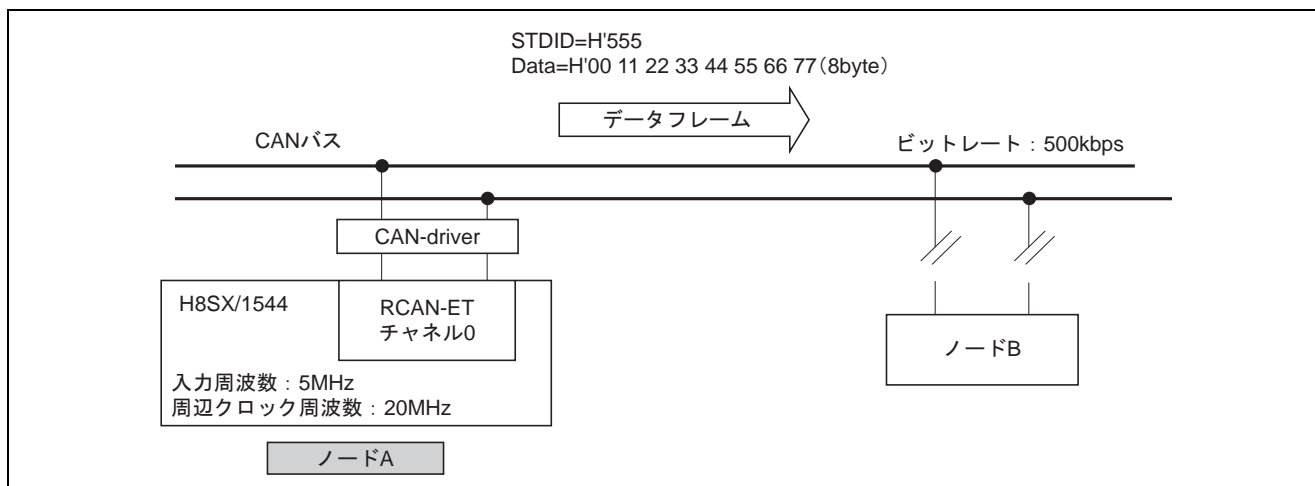


図 2.5.1 通信仕様

### 2.5.2 ソフトウェア説明

#### (1) モジュール説明

表 2.5.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信 メールボックス エンプティ割り込み	RCAN0_Tx SLE0_0		

#### (2) 使用レジスタ説明

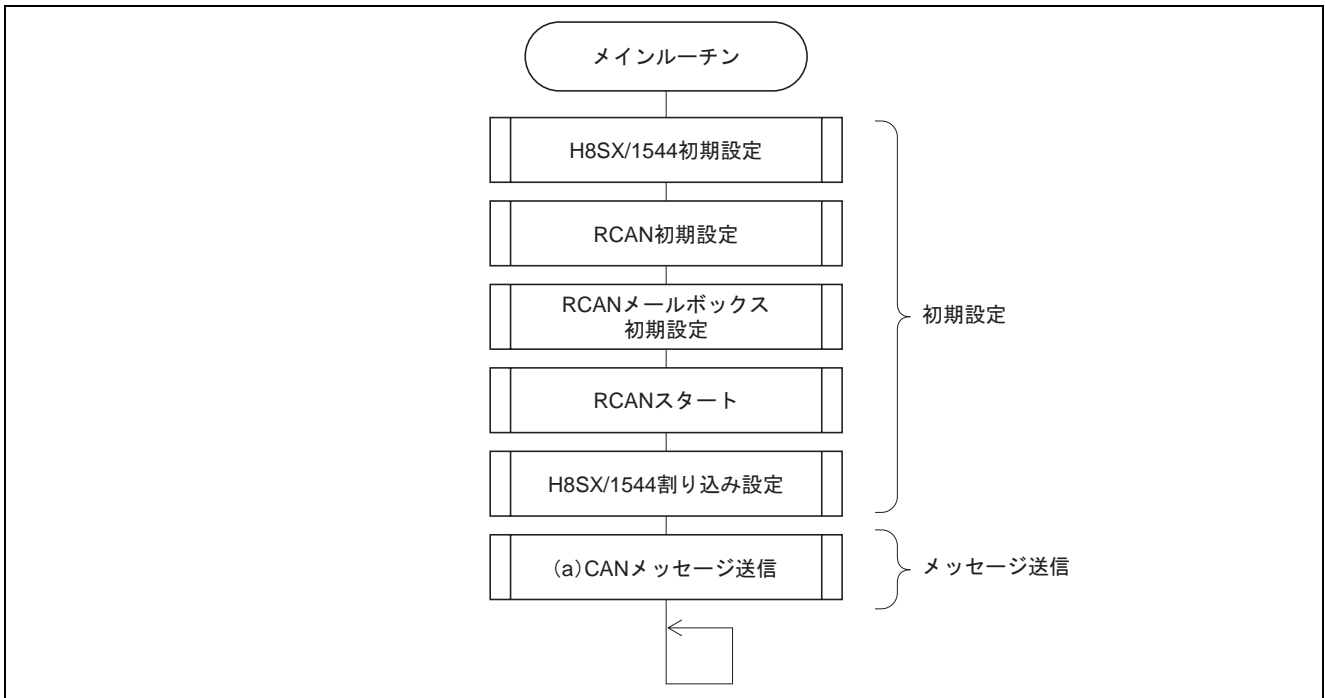
表 2.5.2 使用レジスタ説明

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み (IRR8) を許可に設定。(IMR8=0)
	RCANET0.MBIMR0.WORD	0xFFFD	メールボックス 1 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID (H'555) を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x08	メールボックス 1 を送信用に設定。また、自動再送信無効に設定 (DART=1, MBC=b'000)。DART=1 に設定した場合、対応する TXCR のビットが、送信の最初に自動的にセットされる。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0]	0x00	送信データを設定。
	RCANET0.MB[1].MSG_DATA[1]	0x11	
	RCANET0.MB[1].MSG_DATA[2]	0x22	
RCANET0.MB[1].MSG_DATA[3]	0x33		

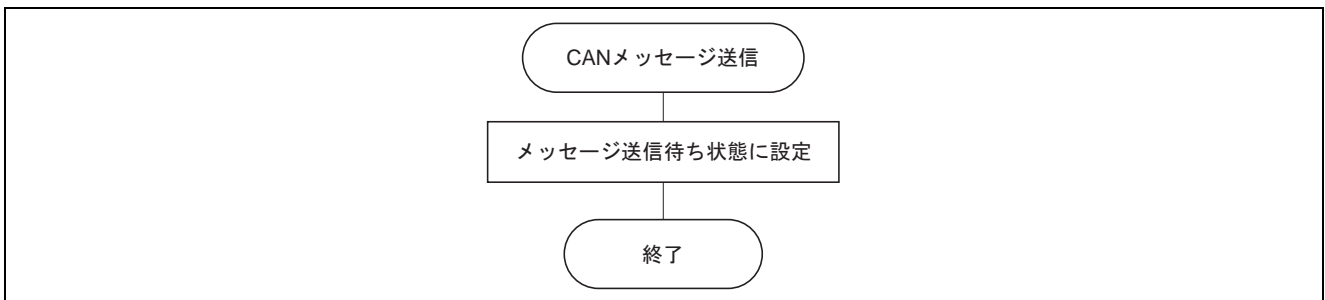
モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[1].MSG_DATA[4]	0x44	送信データを設定。
	RCANET0.MB[1].MSG_DATA[5]	0x55	
	RCANET0.MB[1].MSG_DATA[6]	0x66	
	RCANET0.MB[1].MSG_DATA[7]	0x77	
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に 設定。
メールボックス エンプティ 割り込み	RCANET0.ABACK0.WORD	0x0002	メールボックス 1 の送信アポートフ ラグのクリア。 (クリア条件:1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

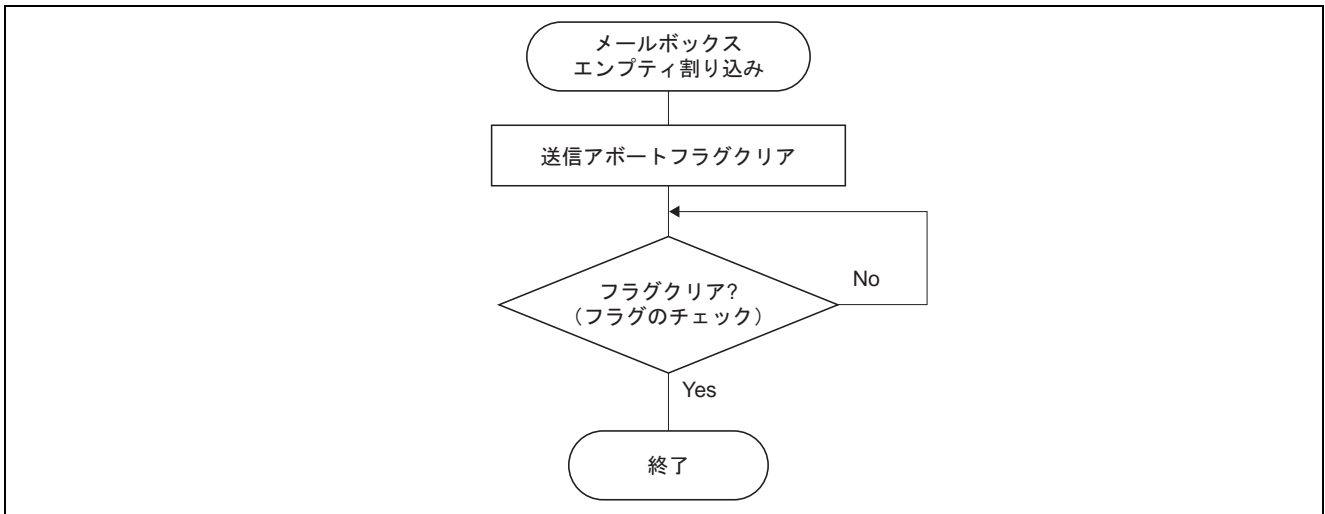
2.5.3 フローチャート  
メインルーチン



(a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

## 2.5.4 プログラムリスト

## (1) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();          /* H8SX/1544 初期化          */
    set_RCAN0_init();        /* RCAN 初期化              */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定  */
    RCAN0_Tx();             /* CANメッセージ送信        */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;      /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;      /* 周辺クロック(5MHz×4=20MHz)          */
    SCKCR.BIT.BCK = 1;      /* 外部バスクロック(5MHz×4=20MHz)      */
    SBYCR.BIT.SSBY = 1;     /* ソフトウェアスタンバイモード        */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN  */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル          */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定        */
}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```



### (2) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose      : for H8SX/1544 RCAN-ET
/*****
#include "iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックスエンティ割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554;/* STDID:H'555,スタンダードフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x08;  /* DART=1(再送信無効),MBC=B'000 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;  /* データ長:8byte */
}

```

```

RCANET0.MB[1].MSG_DATA[0] = 0x00;          /* 送信データ:0x00          */
RCANET0.MB[1].MSG_DATA[1] = 0x11;          /* 送信データ:0x11          */
RCANET0.MB[1].MSG_DATA[2] = 0x22;          /* 送信データ:0x22          */
RCANET0.MB[1].MSG_DATA[3] = 0x33;          /* 送信データ:0x33          */
RCANET0.MB[1].MSG_DATA[4] = 0x44;          /* 送信データ:0x44          */
RCANET0.MB[1].MSG_DATA[5] = 0x55;          /* 送信データ:0x55          */
RCANET0.MB[1].MSG_DATA[6] = 0x66;          /* 送信データ:0x66          */
RCANET0.MB[1].MSG_DATA[7] = 0x77;          /* 送信データ:0x77          */

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1                      */

}

/*****
/*      RCAN start routine          */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0クリア          */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?          */

}

/*****
/*      RCAN send message routine    */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002;     /* MB1を送信待ち状態に設定          */

}

/*****
/*      Mailbox Empty Interrupt routine          */
/*****
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{

    RCANET0.ABACK0.WORD = 0x0002;        /* 送信ボートフラグクリア(クリア条件:1ライト) */
    while(RCANET0.ABACK0.WORD & 0x0002); /* フラグのチェック          */

}
/*****

```

2.6 ニューメッセージコントロール (NMC)

2.6.1 仕様

図 2.6.1 に示すように、ノード A からメッセージ 1,2 を送信し、ノード B で受信します。

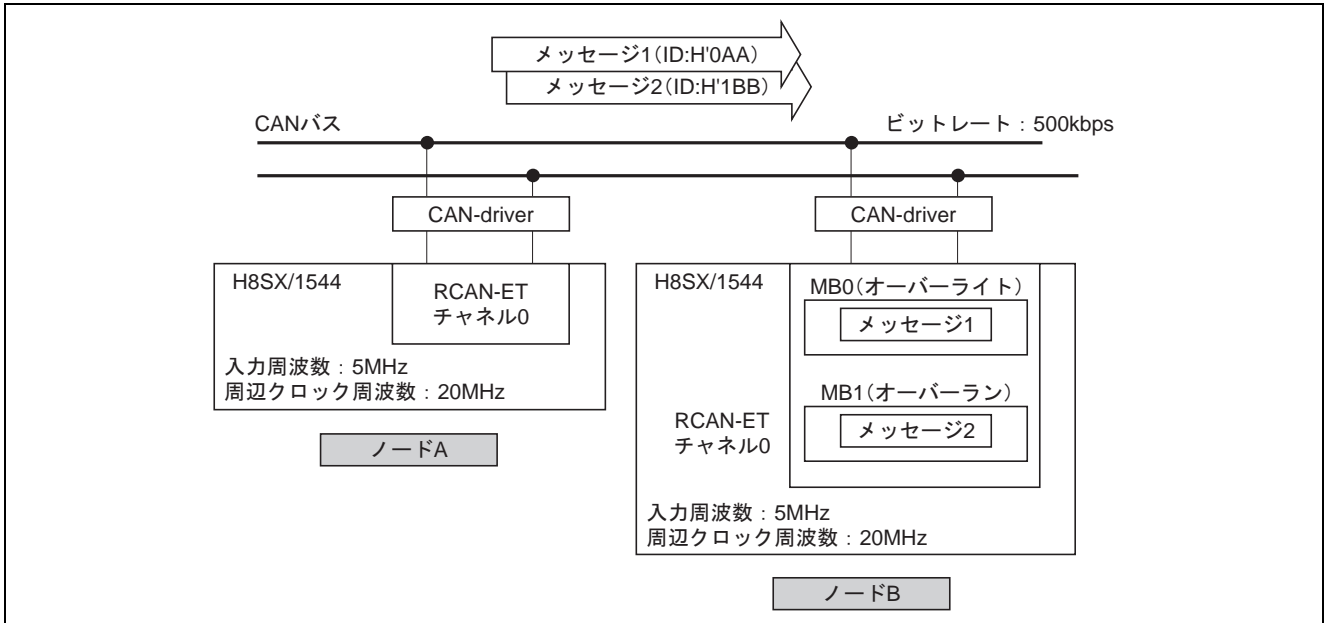


図 2.6.1 通信仕様 (1)

ノード B の受信完了フラグ (RXPR) がクリアされていない状態で、ノード A から再度メッセージ 3,4 を送信すると、ノード B でメールボックスのオーバーライト/オーバーランが発生します。

- メールボックス 0 (MB0) : オーバーライトモード (NMC=1) に設定されているので、メッセージ 3 が上書きされます。
- メールボックス 1 (MB1) : オーバーランモード (NMC=0) に設定されているので、メッセージ 4 は上書きされず、メッセージ 2 がそのまま保持されます。

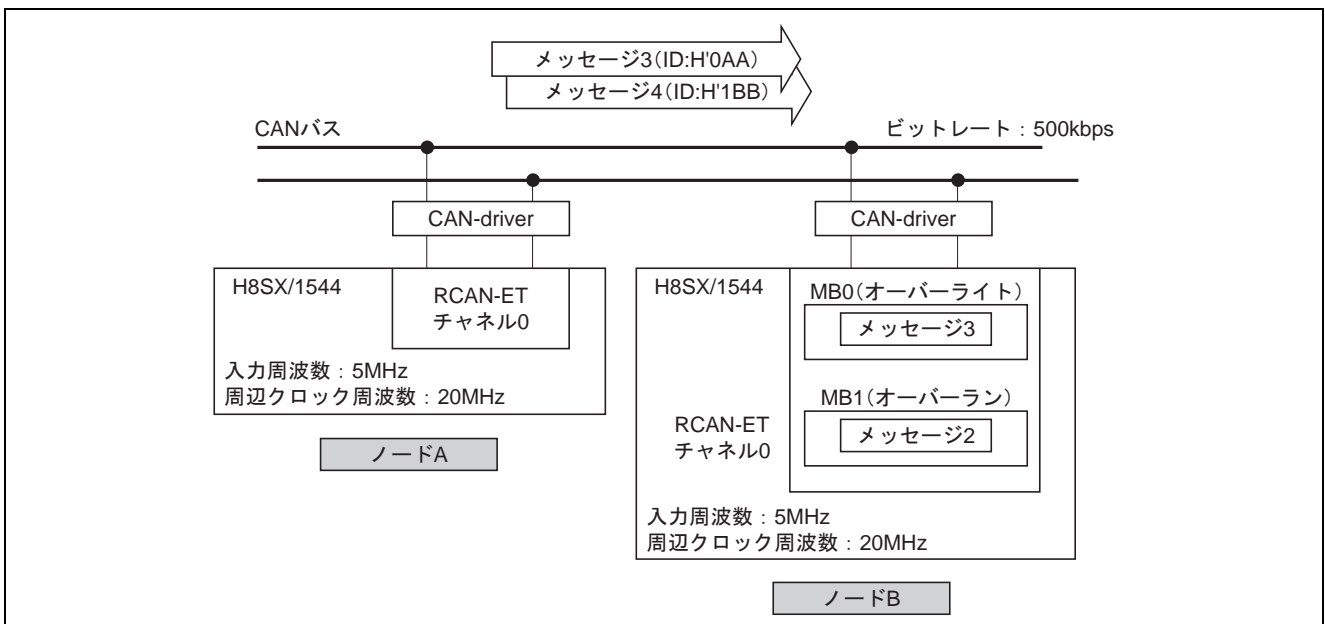


図 2.6.2 通信仕様 (2)

## 2.6.2 ソフトウェア説明

### (1) モジュール説明

表 2.6.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ割り込み	SLE0_0	CAN メッセージ送信完了フラグのクリア。 次メッセージの送信。	
メッセージオーバーライト /オーバーラン割り込み	OVR0_0	未読メッセージステータスフラグのクリア。 未読メッセージを RAM に格納。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリア の初期化。	-

### (2) 使用定数説明

表 2.6.2 使用定数説明

ラベル名	設定値	機能
ID_1	0x0AA	CAN メッセージ ID
ID_2	0x1BB	CAN メッセージ ID

### (3) 使用レジスタ説明

#### (a) 送信側

表 2.6.3 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み (IRR8) を許可に設定。(IMR8=0)
	RCANET0.MBIMR0.WORD	0xFFF9	メールボックス 1,2 を割り込み許可に 設定。
	RCANET0.MB[1].CTRL0.WORD.H	(ID_1<<2)	メールボックス 1 をスタンダード フォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[1].MSG_DATA[0]	0x11	送信データを設定。
	RCANET0.MB[1].MSG_DATA[1]	0x11	
	RCANET0.MB[1].MSG_DATA[2]	0x11	
	RCANET0.MB[1].MSG_DATA[3]	0x11	
	RCANET0.MB[1].MSG_DATA[4]	0x11	
	RCANET0.MB[1].MSG_DATA[5]	0x11	
	RCANET0.MB[1].MSG_DATA[6]	0x11	
	RCANET0.MB[1].MSG_DATA[7]	0x11	
	RCANET0.MB[2].CTRL0.WORD.H	(ID_2<<2)	メールボックス 2 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[2].CTRL1.BYTE.H	0x00	メールボックス 2 を送信用に設定。
	RCANET0.MB[2].CTRL1.BYTE.L	0x08	データ長 ( 8byte ) を設定。
	RCANET0.MB[2].MSG_DATA[0]	0x22	送信データを設定。
	RCANET0.MB[2].MSG_DATA[1]	0x22	
	RCANET0.MB[2].MSG_DATA[2]	0x22	
	RCANET0.MB[2].MSG_DATA[3]	0x22	
	RCANET0.MB[2].MSG_DATA[4]	0x22	
	RCANET0.MB[2].MSG_DATA[5]	0x22	
	RCANET0.MB[2].MSG_DATA[6]	0x22	
	RCANET0.MB[2].MSG_DATA[7]	0x22	
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
RCANET0.BCR0.WORD	0x0001		
RCAN スタート	「 2.1 初期設定 」 参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000006	メールボックス 1,2 を送信待ち状態に設定。
メールボックス エンプティ 割り込み	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)
	RCANET0.TXACK0.WORD	0x0004	メールボックス 2 の送信完了フラグのクリア。(クリア条件: 1 ライト)
	RCANET0.MB[1].MSG_DATA[0]	0x33	送信データを設定。
	RCANET0.MB[1].MSG_DATA[1]	0x33	
	RCANET0.MB[1].MSG_DATA[2]	0x33	
	RCANET0.MB[1].MSG_DATA[3]	0x33	
	RCANET0.MB[1].MSG_DATA[4]	0x33	
	RCANET0.MB[1].MSG_DATA[5]	0x33	
	RCANET0.MB[1].MSG_DATA[6]	0x33	
	RCANET0.MB[1].MSG_DATA[7]	0x33	
	RCANET0.MB[2].MSG_DATA[0]	0x44	
	RCANET0.MB[2].MSG_DATA[1]	0x44	
	RCANET0.MB[2].MSG_DATA[2]	0x44	
	RCANET0.MB[2].MSG_DATA[3]	0x44	
	RCANET0.MB[2].MSG_DATA[4]	0x44	
RCANET0.MB[2].MSG_DATA[5]	0x44		

モジュール名	レジスタ名	設定値	機能
メールボックス エンプティ 割り込み	RCANET0.MB[2].MSG_DATA[6]	0x44	送信データを設定。
	RCANET0.MB[2].MSG_DATA[7]	0x44	
	RCANET0.TXPR0.LONG	0x00000006	メールボックス 1,2 を送信待ち状態に設定。

- 【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。  
 2. ID\_1, ID\_2 は「2.6.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

## (b) 受信側

表 2.6.4 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定	「2.1 初期設定」参照		
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFDFE	メッセージオーバーラン/オーバーライト割り込み (IRR9) を許可に設定。(IMR9=0)
	RCANET0.MBIMR0.WORD	0xFFFC	メールボックス 0,1 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	(ID_1<<2)	メールボックス 0 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x22	メールボックス 0 を受信用に設定。また、オーバーライトモードを設定。(NMC=1, MBC=b'010)
	RCANET0.MB[1].CTRL0.WORD.H	(ID_2<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[1].LAFM.WORD.H	0x0000	メールボックス 1 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x02	メールボックス 1 を受信用に設定。また、オーバーランモードを設定。(NMC=0, MBC=b'010)
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)
RCANET0.BCR0.WORD	0x0001		
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定	「2.1 初期設定」参照		

モジュール名	レジスタ名	設定値	機能
メッセージオーバー ライト/オーバーラン 割り込み	RCANET0.UMSR0.WORD	0x0001	メールボックス 0 の未読メッセ ジステータスフラグのクリア。 (クリア条件: 1 ライト)
	RCANET0.UMSR0.WORD	0x0002	メールボックス 1 の未読メッセ ジステータスフラグのクリア。 (クリア条件: 1 ライト)

- 【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。  
 2. ID\_1, ID\_2 は「2.6.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

#### (4) 使用 RAM 説明

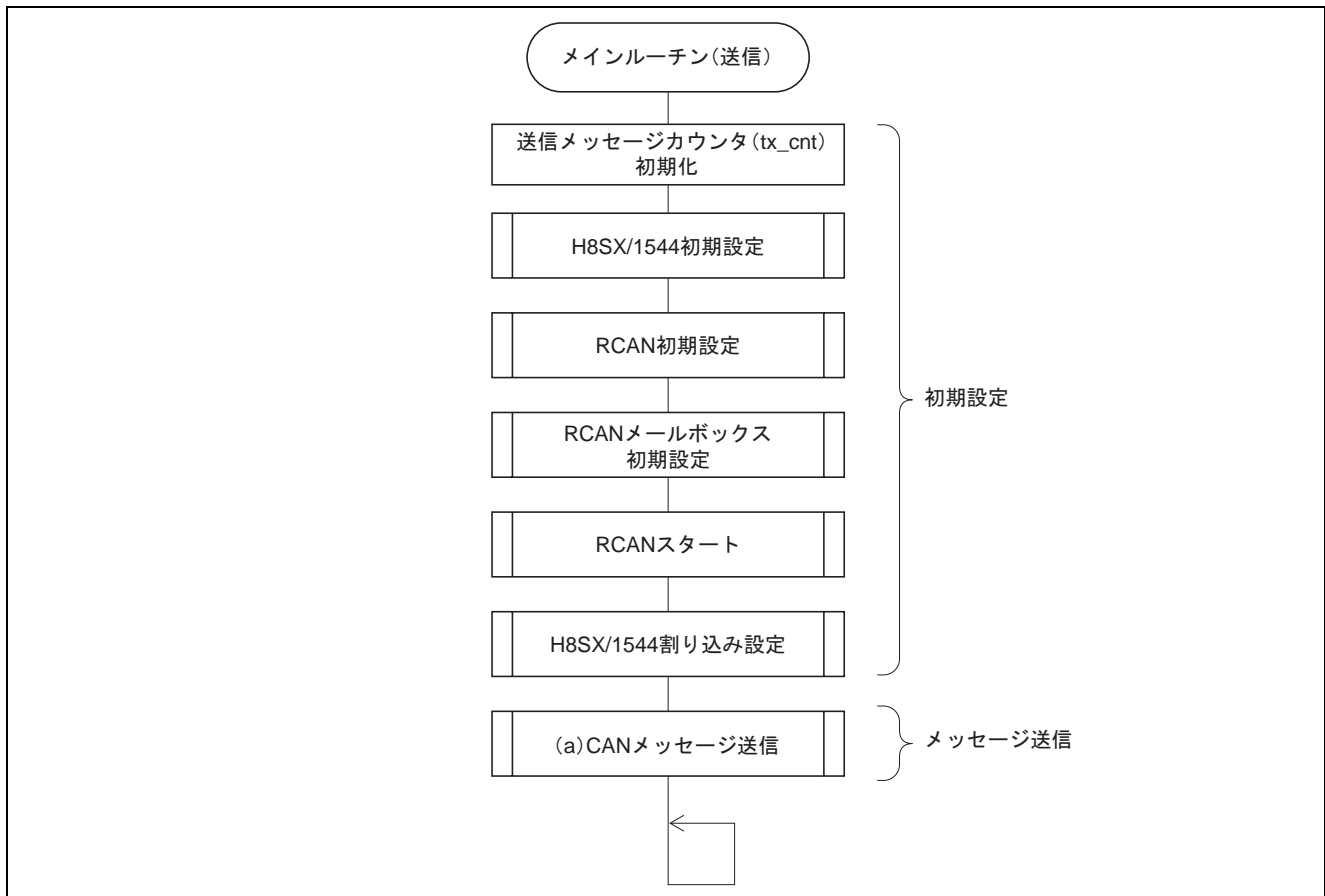
表 2.6.5 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化  メッセージオーバーライト/ オーバーラン割り込み	Mbbuff[2].ID.LONG MBbuff[2].ID.WORD.H MBbuff[2].ID.WORD.L	受信 ID を格納します。
	MBbuff[2].DATA.LONG[0] ~ [1] MBbuff[2].DATA.WORD[0] ~ [3] MBbuff[2].DATA.BYTE[0] ~ [7]	受信データを格納します。
メインルーチン メールボックス エンプティ割り込み	tx_cnt	送信完了メッセージ数を示します。

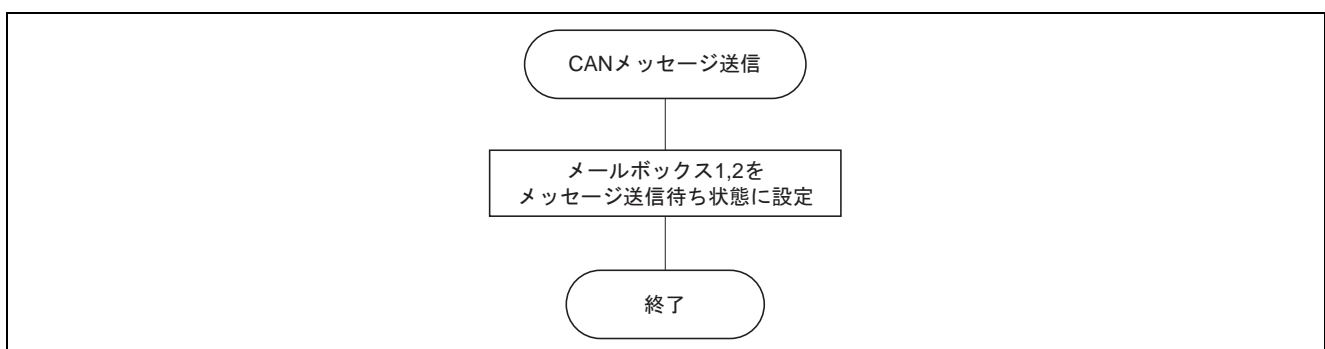
### 2.6.3 フローチャート

#### (1) 送信側フローチャート

##### メインルーチン

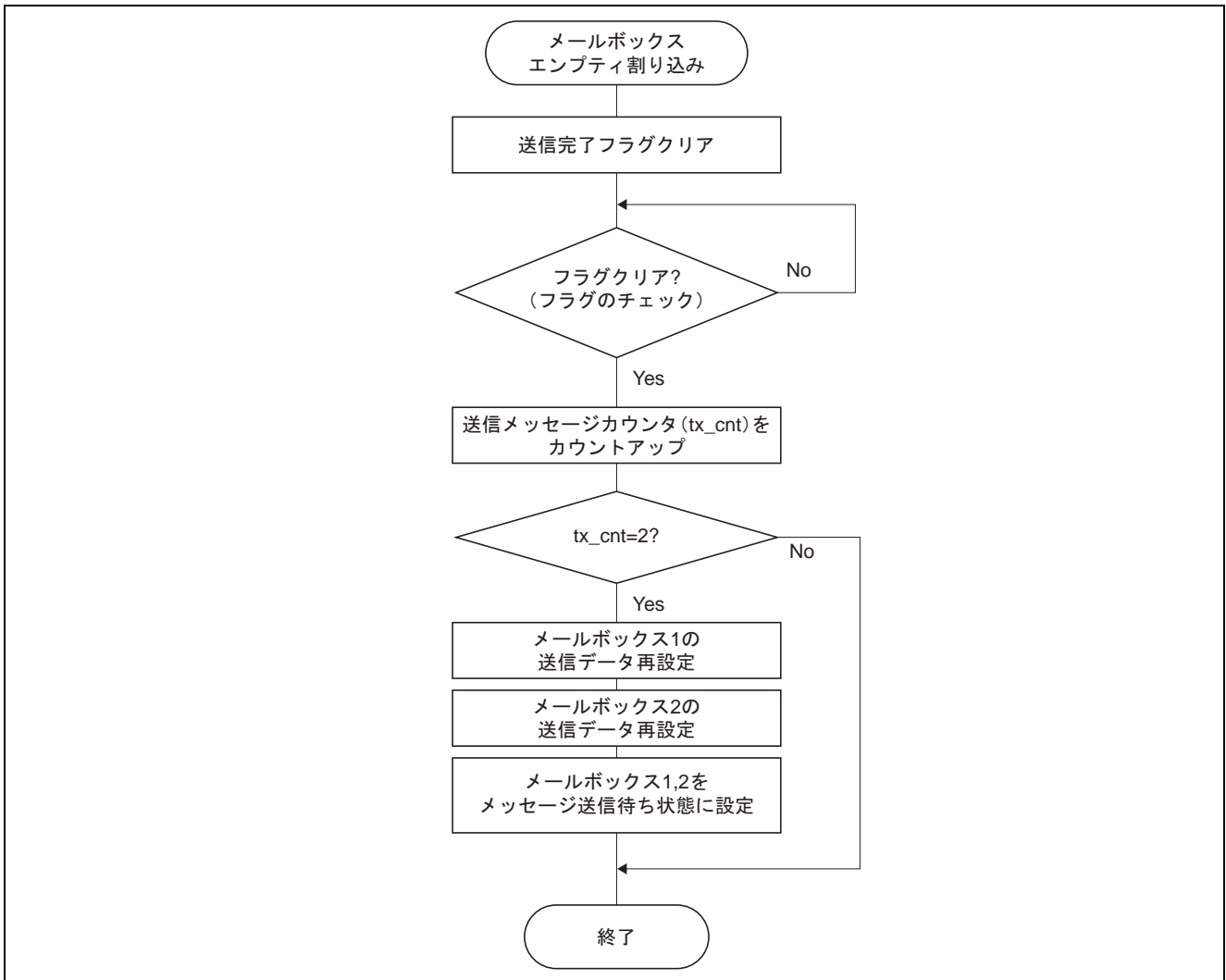


#### (a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)





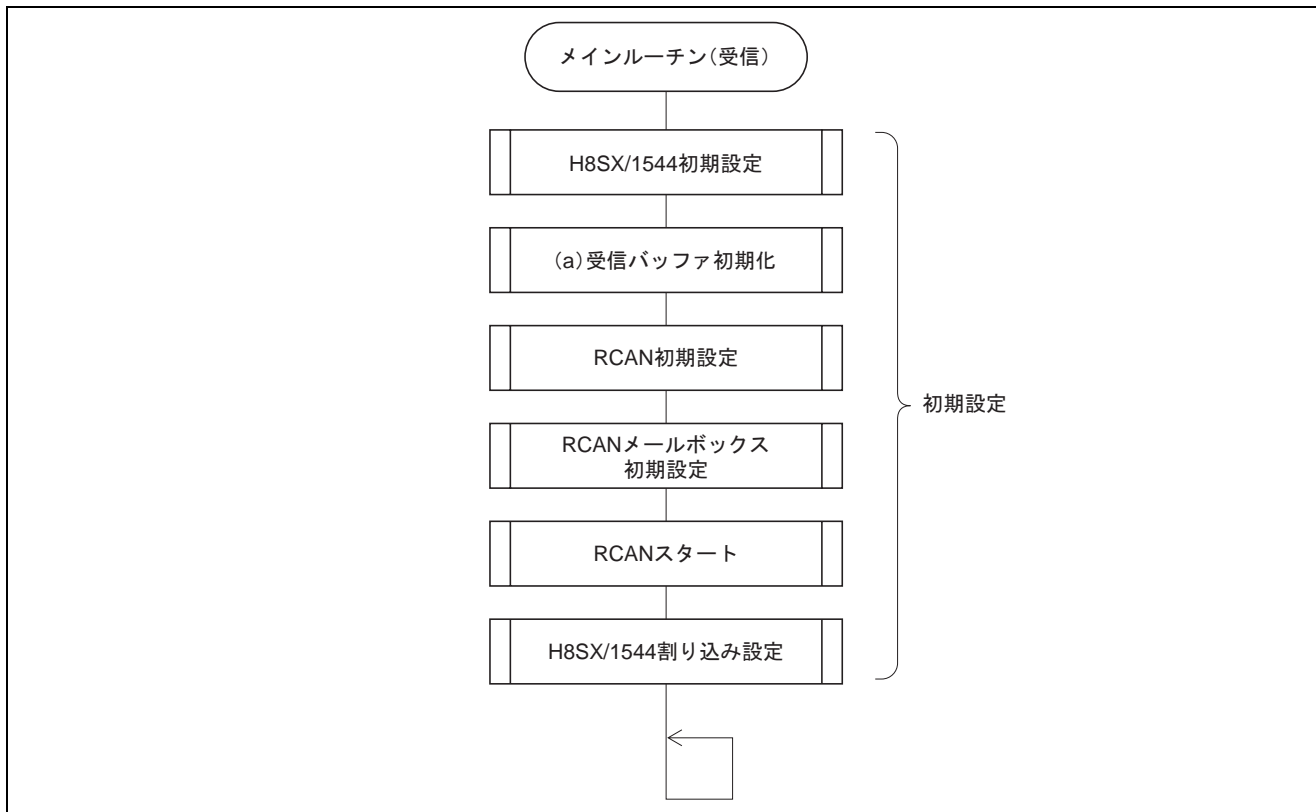
(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



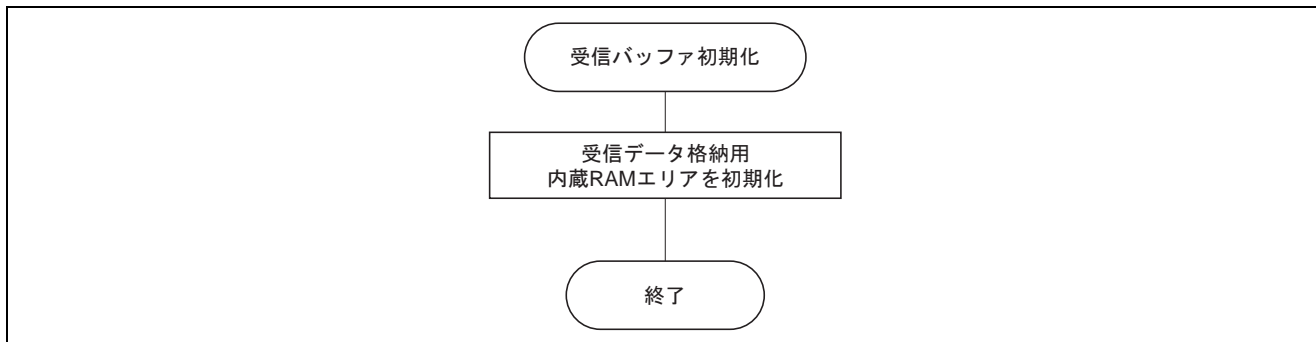
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

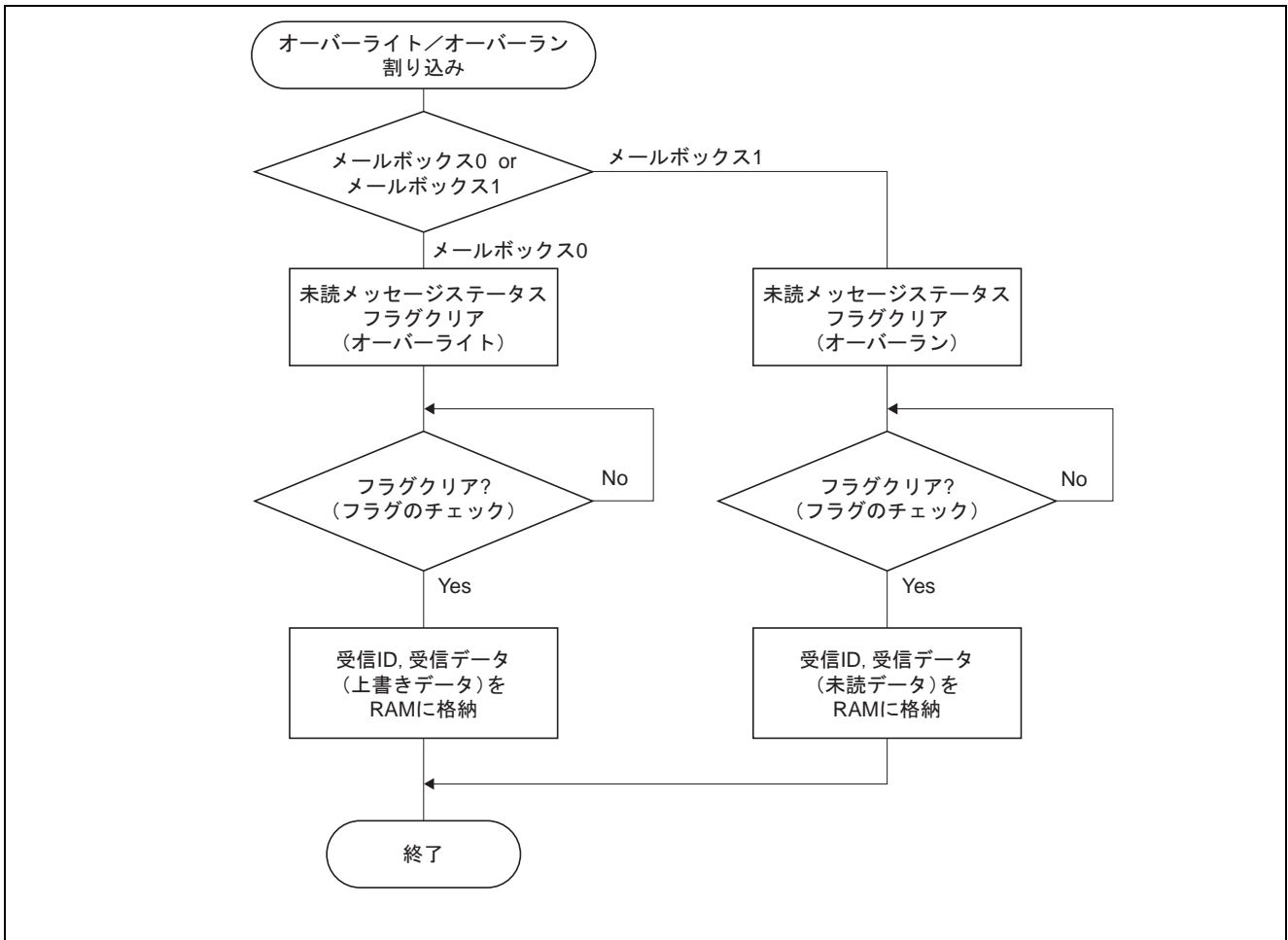
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)



(b) メッセージオーバーラン / オーバーライト割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

## 2.6.4 プログラムリスト

### (1) 送信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*-----*/
unsigned char tx_cnt;
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    tx_cnt = 0;
    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定     */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;        /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;        /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;       /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose      : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*-----*/
extern unsigned char tx_cnt;
/*-----*/
#define ID_1      0x0AA
#define ID_2      0x1BB
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メモリマックス(RAMエリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックス割り込み許可 */
    RCANET0.MBIMR0.WORD = 0xFFF9;       /* MB1-2 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;     /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;     /* データ長:8byte */
    for(i = 0;i < 8;i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x11; /* 送信データ */
    }

    RCANET0.MB[2].CTRL0.WORD.H = (ID_2<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[2].CTRL1.BYTE.H = 0x00;     /* メールボックス2を送信用に設定 */
    RCANET0.MB[2].CTRL1.BYTE.L = 0x08;     /* データ長:8byte */
    for(i = 0;i < 8;i++){
        RCANET0.MB[2].MSG_DATA[i] = 0x22; /* 送信データ */
    }

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/*      RCAN start routine
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RCAN send message routine
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000006;     /* MB1-2 を送信待ち状態に設定 */

}

```

```

/*****
/*      Mailbox Empty Interrupt routine                               */
/*****
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{

    unsigned int i;

    RCANET0.TXACK0.WORD = RCANET0.TXACK0.WORD; /* 送信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.TXACK0.WORD);                /* フラグのチェック */

    tx_cnt++;

    if(tx_cnt == 2){
        for(i = 0;i < 8;i++){
            RCANET0.MB[1].MSG_DATA[i] = 0x33; /* 送信データ */
        }
        for(i = 0;i < 8;i++){
            RCANET0.MB[2].MSG_DATA[i] = 0x44; /* 送信データ */
        }
        RCANET0.TXPR0.LONG = 0x00000006;      /* MB1-2 を送信待ち状態に設定 */
    }

}

/*****

```



## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
*****/
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program                                     */
*****/
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();                /* H8SX/1544 初期化          */
    Clear_MBbuff();                /* データ格納用RAMクリア初期化 */
    set_RCAN0_init();              /* RCAN 初期化              */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();          /* H8SX/1544 割り込み設定    */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
*****/
void set_1544_init(void){
    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;             /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;             /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;             /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;           /* ソフトウェアスタンバイモード      */
    MSTP.CRC.BIT._RCAN01 = 0;     /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20;  /* RCAN-ET 送受信端子レベル          */
    P6.ICR.BIT.B4 = 1;           /* P64 を CRx_0(入力端子)に設定      */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}

/*****/

```

## (b) rcan.c

```

/*****
/* Filename: rcan.c*/
/* Written: '06/06/01 REV.1.00 */
/* Purpose: for H8SX/1544 RCAN-ET */
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff[2];
/*-----*/
#define ID_1      0x0AA
#define ID_2      0x1BB
/*****
/* RCAN Initialize routine */
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine                                     */
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFDFE;          /* メッセージオーバーラン/オーバーライト割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFFC;     /* MB0-1 割り込み許可 */

    RCANET0.MB[0].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダートフォーマット,データレーム */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000;     /* STD_LAFM, IDE_LAFM の設定 */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x22; /* NMC=1(オーバーライトモード),MBC=b'010(受信用) */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_2<<2); /* STDID 設定,スタンダートフォーマット,データレーム */
    RCANET0.MB[1].LAFM.WORD.H = 0x0000;     /* STD_LAFM, IDE_LAFM の設定 */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x02; /* NMC=0(オーバーランモード),MBC=b'010(受信用) */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001;          /* BRP=1 */

}

/*****
/*      RCAN start routine                                               */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE;        /* MCRO クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RAM area Initialize routine                                       */
/*****
void Clear_MBbuff(void){

    unsigned int i;

    for(i = 0;i < 2;i++){                /* データ格納用 RAM 初期化 */
        MBbuff[i].ID.LONG = 0;
        MBbuff[i].DATA.LONG[0] = 0;
        MBbuff[i].DATA.LONG[1] = 0;
    }

}

```

```

/*****
/*      Message Overrun/Overwrite Interrupt routine      */
/*****
#pragma interrupt(OVR0_0)
void OVR0_0(void)
{
    unsigned int i;

    if(RCANET0.UMSR0.WORD&0x0001){          /* IRR9 セット,MB0 オーバ-ライト      */
        RCANET0.UMSR0.WORD = 0x0001;      /* 未読メッセージステータスフラグクリア(クリア条件:1ライト) */
        while(RCANET0.UMSR0.WORD & 0x0001); /* フラグのチェック */
        MBbuff[0].ID.WORD.H = RCANET0.MB[0].CTRL0.WORD.H; /* 受信IDをRAMに格納 */
        MBbuff[0].ID.WORD.L = RCANET0.MB[0].CTRL0.WORD.L; /* 受信IDをRAMに格納 */
        for(i = 0;i < 8;i++){             /* 受信データをRAMに格納 */
            MBbuff[0].DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i];
        }
    }
    else if(RCANET0.UMSR0.WORD&0x0002){    /* IRR9 セット,MB1 オーバ-ライト      */
        RCANET0.UMSR0.WORD = 0x0002;      /* 未読メッセージステータスフラグクリア(クリア条件:1ライト) */
        while(RCANET0.UMSR0.WORD & 0x0002); /* フラグのチェック */
        MBbuff[1].ID.WORD.H = RCANET0.MB[1].CTRL0.WORD.H; /* 受信IDをRAMに格納 */
        MBbuff[1].ID.WORD.L = RCANET0.MB[1].CTRL0.WORD.L; /* 受信IDをRAMに格納 */
        for(i = 0;i < 8;i++){             /* 受信データをRAMに格納 */
            MBbuff[1].DATA.BYTE[i] = RCANET0.MB[1].MSG_DATA[i];
        }
    }
}
/*****

```

2.7 メッセージフィルタリング (LAFM) によるグループ受信

2.7.1 仕様

図 2.7.1 に示すように、ノード A からの 15 種類のメッセージを送信し、ノード B で受信します。ノード B は、受信メッセージのフィルタリング機能 (LAFM) を使用し、メッセージのグループ受信を行います。

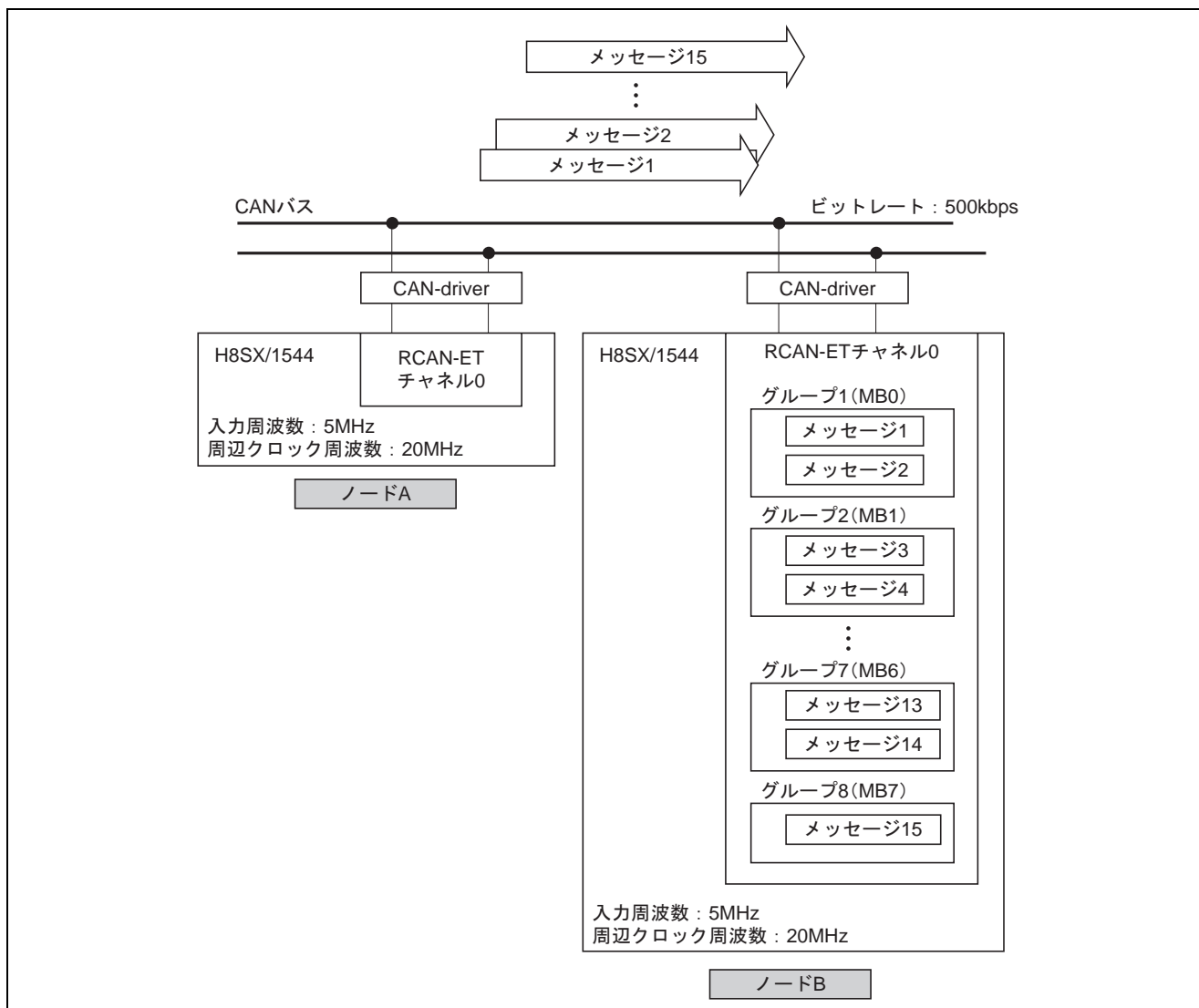


図 2.7.1 通信仕様

表 2.7.1 に、メッセージの仕様を示します。

表 2.7.1 メッセージ仕様

メッセージ	スタンダード ID	データ	受信メールボックス
メッセージ 1	H'011	H'01 01 01 01 01 01 01 01(8byte)	メールボックス 0
メッセージ 2	H'0AA	H'02 02 02 02 02 02 02 02(8byte)	
メッセージ 3	H'122	H'03 03 03 03 03 03 03 03(8byte)	メールボックス 1
メッセージ 4	H'1BB	H'04 04 04 04 04 04 04 04(8byte)	メールボックス 2
メッセージ 5	H'233	H'05 05 05 05 05 05 05 05(8byte)	
メッセージ 6	H'2CC	H'06 06 06 06 06 06 06 06(8byte)	メールボックス 3
メッセージ 7	H'344	H'07 07 07 07 07 07 07 07(8byte)	
メッセージ 8	H'3DD	H'08 08 08 08 08 08 08 08(8byte)	メールボックス 4
メッセージ 9	H'455	H'09 09 09 09 09 09 09 09(8byte)	
メッセージ 10	H'4EE	H'0A 0A 0A 0A 0A 0A 0A 0A(8byte)	メールボックス 5
メッセージ 11	H'566	H'0B 0B 0B 0B 0B 0B 0B 0B(8byte)	
メッセージ 12	H'5FF	H'0C 0C 0C 0C 0C 0C 0C 0C(8byte)	メールボックス 6
メッセージ 13	H'677	H'0D 0D 0D 0D 0D 0D 0D 0D(8byte)	
メッセージ 14	H'6EE	H'0E 0E 0E 0E 0E 0E 0E 0E(8byte)	メールボックス 7
メッセージ 15	H'788	H'0F 0F 0F 0F 0F 0F 0F 0F(8byte)	

【注】 本動作例では、スタンダードフォーマットを使用します。(エクステンデッド ID は設定不要)

## 2.7.2 ソフトウェア説明

## (1) モジュール説明

表 2.7.2 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
データフレーム 受信割り込み (メールボックス 0)	RM0_0	CAN メッセージの受信完了フラグクリア。 メールボックス 0 の受信データを RAM に格納。	
データフレーム 受信割り込み (メールボックス 1~7)	RM1_0	CAN メッセージの受信完了フラグクリア。 メールボックス 1~7 の受信データを RAM に格納。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	-

## (2) 使用定数説明

表 2.7.3 使用定数説明

ラベル名	設定値	機能
ID_1	0x011	CAN メッセージ ID
ID_2	0x0AA	
ID_3	0x122	
ID_4	0x1BB	
ID_5	0x233	
ID_6	0x2CC	
ID_7	0x344	
ID_8	0x3DD	
ID_9	0x455	
ID_10	0x4EE	
ID_11	0x566	
ID_12	0x5FF	
ID_13	0x677	
ID_14	0x6EE	
ID_15	0x788	



ラベル名	設定値	機能
STD_ID_MB0	0x000	受信メールボックスの ID 設定値
STD_ID_MB1	0x100	
STD_ID_MB2	0x200	
STD_ID_MB3	0x300	
STD_ID_MB4	0x400	
STD_ID_MB5	0x500	
STD_ID_MB6	0x600	
STD_ID_MB7	0x700	
STD_LAFM	0x0FF	スタンダード ID のフィルタマスク設定値
IDE_LAFM	0x0000	IDE ビットのフィルタマスク設定値

## (3) 使用レジスタ説明

## (a) 送信側

表 2.7.4 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み( IRR8 )を許可に設定。( IMR8=0 )
	RCANET0.MBIMR0.WORD	0x0001	メールボックス 1~15 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	(ID_1<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0] ~ RCANET0.MB[1].MSG_DATA[7]	0x01 × 8	送信データを設定。
	RCANET0.MB[2].CTRL0.WORD.H	(ID_2<<2)	メールボックス 2 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[2].CTRL1.BYTE.H	0x00	メールボックス 2 を送信用に設定。
	RCANET0.MB[2].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[2].MSG_DATA[0] ~ RCANET0.MB[2].MSG_DATA[7]	0x02 × 8	送信データを設定。
	RCANET0.MB[3].CTRL0.WORD.H	(ID_3<<2)	メールボックス 3 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[3].CTRL1.BYTE.H	0x00	メールボックス 3 を送信用に設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[3].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[3].MSG_DATA[0] ~ RCANET0.MB[3].MSG_DATA[7]	0x03 × 8	送信データを設定。
	RCANET0.MB[4].CTRL0.WORD.H	(ID_4<<2)	メールボックス 4 をスタンダード フォーマット、データフレームに設 定。また、スタンダード ID を設定。
	RCANET0.MB[4].CTRL1.BYTE.H	0x00	メールボックス 4 を送信用に設定。
	RCANET0.MB[4].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[4].MSG_DATA[0] ~ RCANET0.MB[4].MSG_DATA[7]	0x04 × 8	送信データを設定。
	RCANET0.MB[5].CTRL0.WORD.H	(ID_5<<2)	メールボックス 5 をスタンダード フォーマット、データフレームに設 定。また、スタンダード ID を設定。
	RCANET0.MB[5].CTRL1.BYTE.H	0x00	メールボックス 5 を送信用に設定。
	RCANET0.MB[5].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[5].MSG_DATA[0] ~ RCANET0.MB[5].MSG_DATA[7]	0x05 × 8	送信データを設定。
	RCANET0.MB[6].CTRL0.WORD.H	(ID_6<<2)	メールボックス 6 をスタンダード フォーマット、データフレームに設 定。また、スタンダード ID を設定。
	RCANET0.MB[6].CTRL1.BYTE.H	0x00	メールボックス 6 を送信用に設定。
	RCANET0.MB[6].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[6].MSG_DATA[0] ~ RCANET0.MB[6].MSG_DATA[7]	0x06 × 8	送信データを設定。
	RCANET0.MB[7].CTRL0.WORD.H	(ID_7<<2)	メールボックス 7 をスタンダード フォーマット、データフレームに設 定。また、スタンダード ID を設定。
	RCANET0.MB[7].CTRL1.BYTE.H	0x00	メールボックス 7 を送信用に設定。
	RCANET0.MB[7].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[7].MSG_DATA[0] ~ RCANET0.MB[7].MSG_DATA[7]	0x07 × 8	送信データを設定。
	RCANET0.MB[8].CTRL0.WORD.H	(ID_8<<2)	メールボックス 8 をスタンダード フォーマット、データフレームに設 定。また、スタンダード ID を設定。
	RCANET0.MB[8].CTRL1.BYTE.H	0x00	メールボックス 8 を送信用に設定。
RCANET0.MB[8].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。	
RCANET0.MB[8].MSG_DATA[0] ~ RCANET0.MB[8].MSG_DATA[7]	0x08 × 8	送信データを設定。	

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[9].CTRL0.WORD.H	(ID_9<<2)	メールボックス 9 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[9].CTRL1.BYTE.H	0x00	メールボックス 9 を送信用に設定。
	RCANET0.MB[9].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[9].MSG_DATA[0] ~ RCANET0.MB[9].MSG_DATA[7]	0x09 × 8	送信データを設定。
	RCANET0.MB[10].CTRL0.WORD.H	(ID_10<<2)	メールボックス 10 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[10].CTRL1.BYTE.H	0x00	メールボックス 10 を送信用に設定。
	RCANET0.MB[10].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[10].MSG_DATA[0] ~ RCANET0.MB[10].MSG_DATA[7]	0x0A × 8	送信データを設定。
	RCANET0.MB[11].CTRL0.WORD.H	(ID_11<<2)	メールボックス 11 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[11].CTRL1.BYTE.H	0x00	メールボックス 11 を送信用に設定。
	RCANET0.MB[11].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[11].MSG_DATA[0] ~ RCANET0.MB[11].MSG_DATA[7]	0x0B × 8	送信データを設定。
	RCANET0.MB[12].CTRL0.WORD.H	(ID_12<<2)	メールボックス 12 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[12].CTRL1.BYTE.H	0x00	メールボックス 12 を送信用に設定。
	RCANET0.MB[12].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[12].MSG_DATA[0] ~ RCANET0.MB[12].MSG_DATA[7]	0x0C × 8	送信データを設定。
	RCANET0.MB[13].CTRL0.WORD.H	(ID_13<<2)	メールボックス 13 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[13].CTRL1.BYTE.H	0x00	メールボックス 13 を送信用に設定。
	RCANET0.MB[13].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[13].MSG_DATA[0] ~ RCANET0.MB[13].MSG_DATA[7]	0x0D × 8	送信データを設定。
RCANET0.MB[14].CTRL0.WORD.H	(ID_14<<2)	メールボックス 14 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。	
RCANET0.MB[14].CTRL1.BYTE.H	0x00	メールボックス 14 を送信用に設定。	

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[14].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[14].MSG_DATA[0] ~ RCANET0.MB[14].MSG_DATA[7]	0x0E × 8	送信データを設定。
	RCANET0.MB[15].CTRL0.WORD.H	(ID_15<<2)	メールボックス 15 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[15].CTRL1.BYTE.H	0x00	メールボックス 15 を送信用に設定。
	RCANET0.MB[15].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[15].MSG_DATA[0] ~ RCANET0.MB[15].MSG_DATA[7]	0x0F × 8	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み 設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x0000FFFE	メールボックス 1~15 を送信待ち状態に設定。
メールボックス エンプティ 割り込み	RCANET0.TXACK0.WORD	-	メールボックス 1~15 の送信完了フラグのクリア。 (クリア条件: 1 ライト)

- 【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。  
 2. ID\_x(x=15)は「2.7.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

## (b) 受信側

表 2.7.5 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メール ボックス 初期設定	RCANET0.IMR.WORD	0xFFFFD	データフレーム受信割り込み (IRR1) を許可に設定。(IMR1=0)
	RCANET0.MBIMR0.WORD	0xFF00	メールボックス 0~7 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	(STD_ID_MB0<<2)	メールボックス 0 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x03FC	メールボックス 0 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。

モジュール名	レジスタ名	設定値	機能
RCAN メール ボックス 初期設定	RCANET0.MB[1].CTRL0.WORD.H	(STD_ID_MB1<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[1].LAFM.WORD.H	0x03FC	メールボックス 1 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x02	メールボックス 1 を受信用に設定。
	RCANET0.MB[2].CTRL0.WORD.H	(STD_ID_MB2<<2)	メールボックス 2 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[2].LAFM.WORD.H	0x03FC	メールボックス 2 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[2].CTRL1.BYTE.H	0x02	メールボックス 2 を受信用に設定。
	RCANET0.MB[3].CTRL0.WORD.H	(STD_ID_MB3<<2)	メールボックス 3 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[3].LAFM.WORD.H	0x03FC	メールボックス 3 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[3].CTRL1.BYTE.H	0x02	メールボックス 3 を受信用に設定。
	RCANET0.MB[4].CTRL0.WORD.H	(STD_ID_MB4<<2)	メールボックス 4 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[4].LAFM.WORD.H	0x03FC	メールボックス 4 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[4].CTRL1.BYTE.H	0x02	メールボックス 4 を受信用に設定。
	RCANET0.MB[5].CTRL0.WORD.H	(STD_ID_MB5<<2)	メールボックス 5 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[5].LAFM.WORD.H	0x03FC	メールボックス 5 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[5].CTRL1.BYTE.H	0x02	メールボックス 5 を受信用に設定。
	RCANET0.MB[6].CTRL0.WORD.H	(STD_ID_MB6<<2)	メールボックス 6 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
RCANET0.MB[6].LAFM.WORD.H	0x03FC	メールボックス 6 のスタンダード ID、IDE ビットのフィルタマスクを設定。	

モジュール名	レジスタ名	設定値	機能
RCAN メール ボックス 初期設定	RCANET0.MB[6].CTRL1.BYTE.H	0x02	メールボックス 6 を受信用に設定。
	RCANET0.MB[7].CTRL0.WORD.H	(STD_ID_MB7<<2)	メールボックス 7 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[7].LAFM.WORD.H	0x03FC	メールボックス 7 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[7].CTRL1.BYTE.H	0x02	メールボックス 7 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。
	RCANET0.BCR0.WORD	0x0001	( TSEG1=4(5tq),TSEG2=3(4tq), SJV=0,BSP=0,BRP=1 )
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み 設定			
データ フレーム 受信割り込み (メール ボックス 0)	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。 (クリア条件: 1 ライト)
データ フレーム 受信割り込み (メール ボックス 1~7)	RCANET0.RXPR0.WORD	-	メールボックス 1~7 の受信完了フラグのクリア。 (クリア条件: 1 ライト)

【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

2. STD\_ID\_MBx(x=2~7)は「2.7.2 ソフトウェア説明 (2)使用定数説明」のラベル名を示しています。

#### (4) 使用 RAM 説明

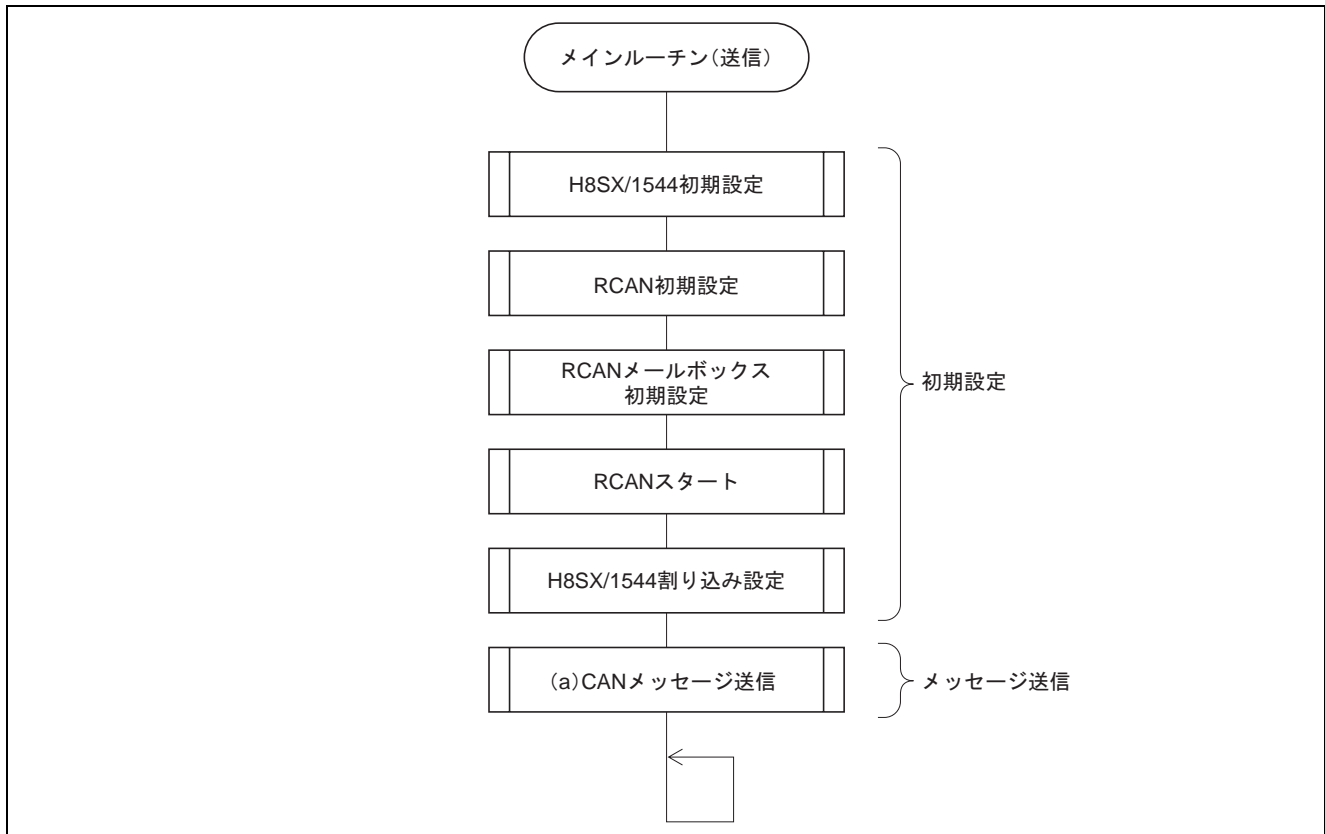
表 2.7.6 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化	Mbbuff[16].ID.LONG MBbuff[16].ID.WORD.H MBbuff[16].ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuff[16].DATA.LONG[0] ~ [1] MBbuff[16].DATA.WORD[0] ~ [3] MBbuff[16].DATA.BYTE[0] ~ [7]	受信データを格納します。

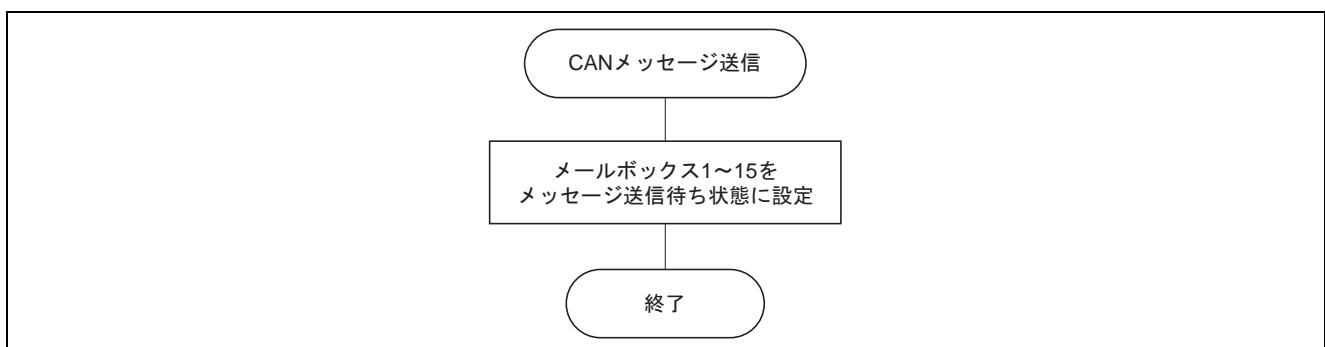
## 2.7.3 フローチャート

### (1) 送信側フローチャート

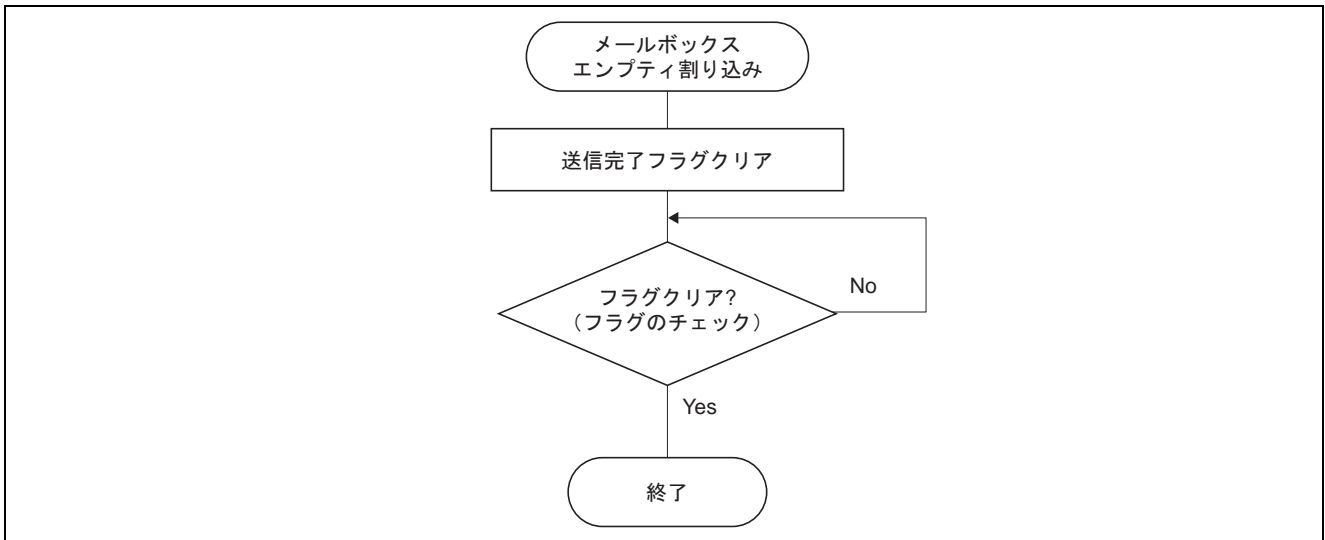
#### メインルーチン



#### (a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)

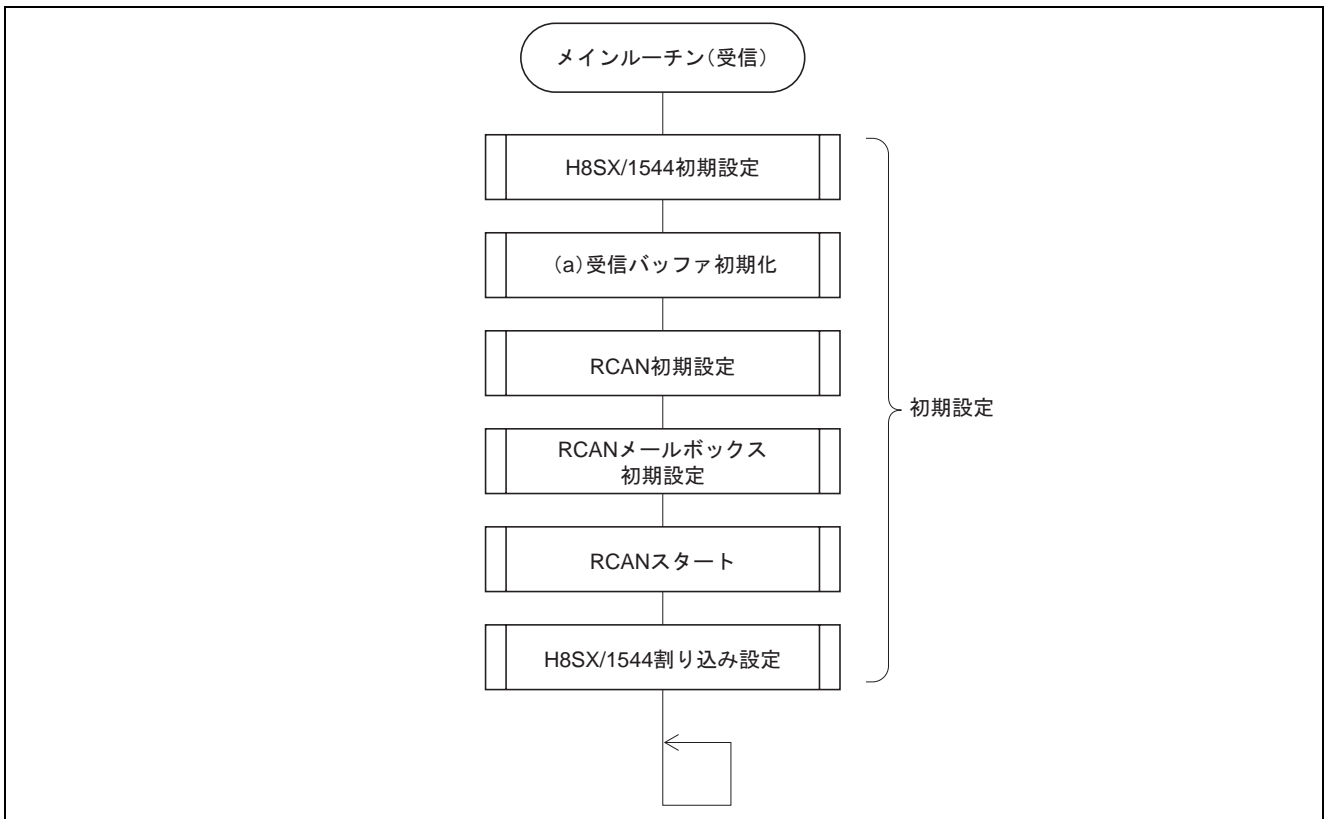


【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

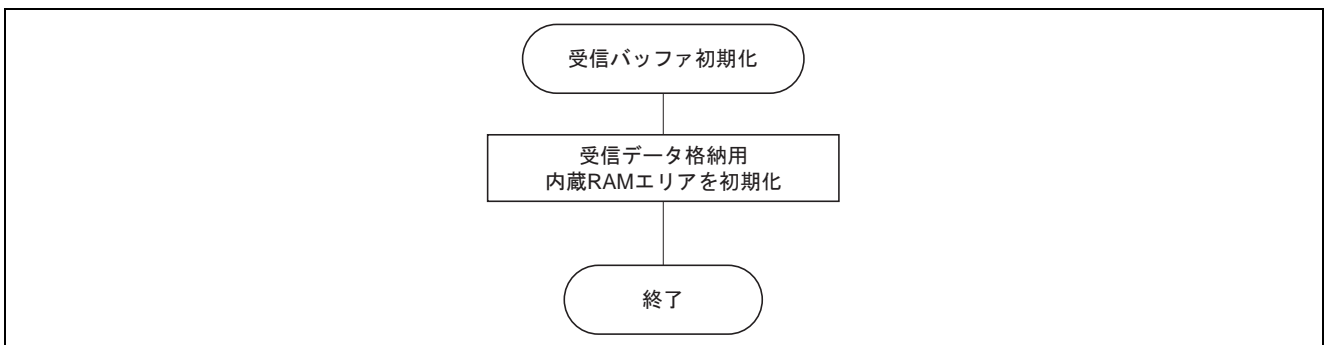


(2) 受信側フローチャート

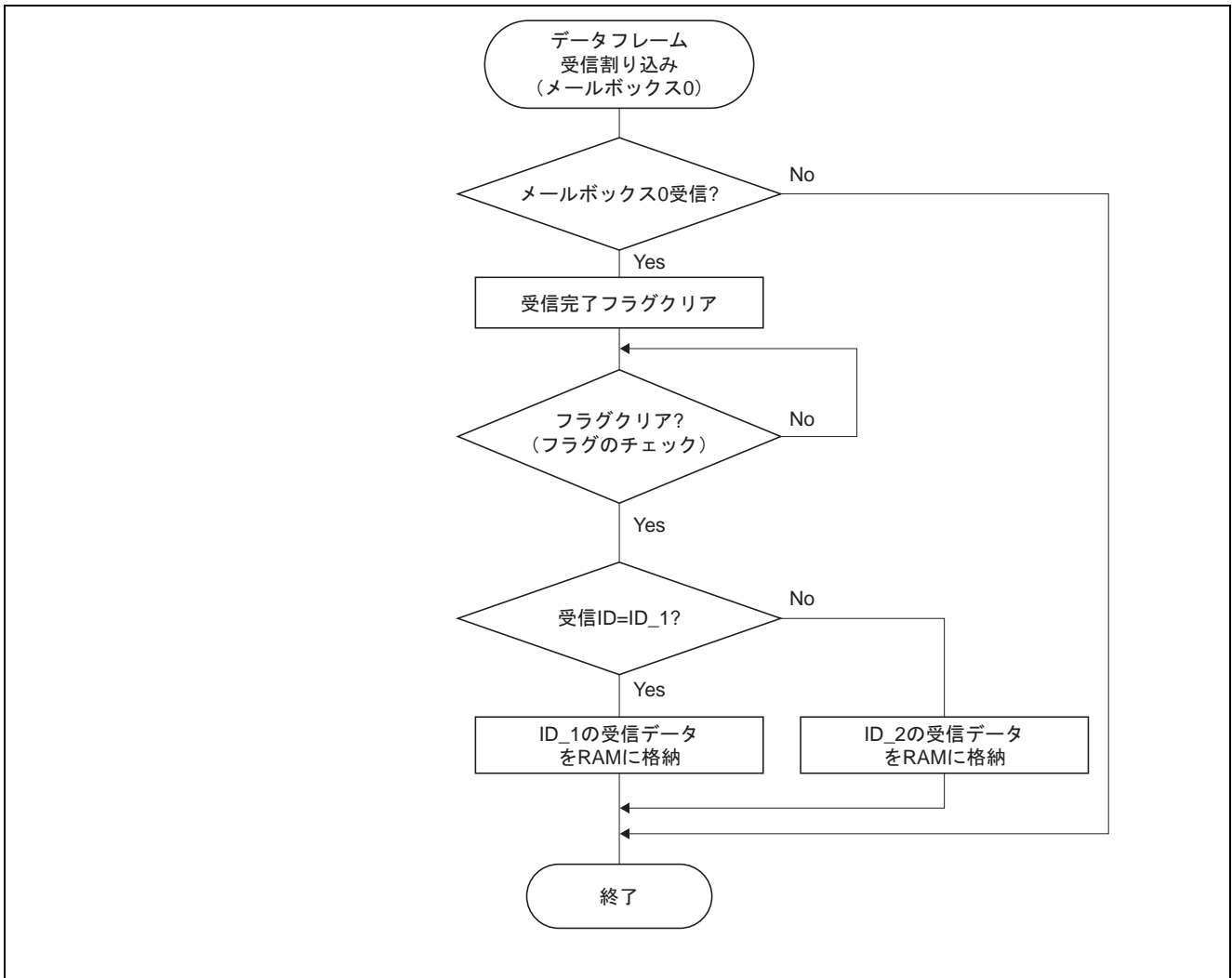
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)

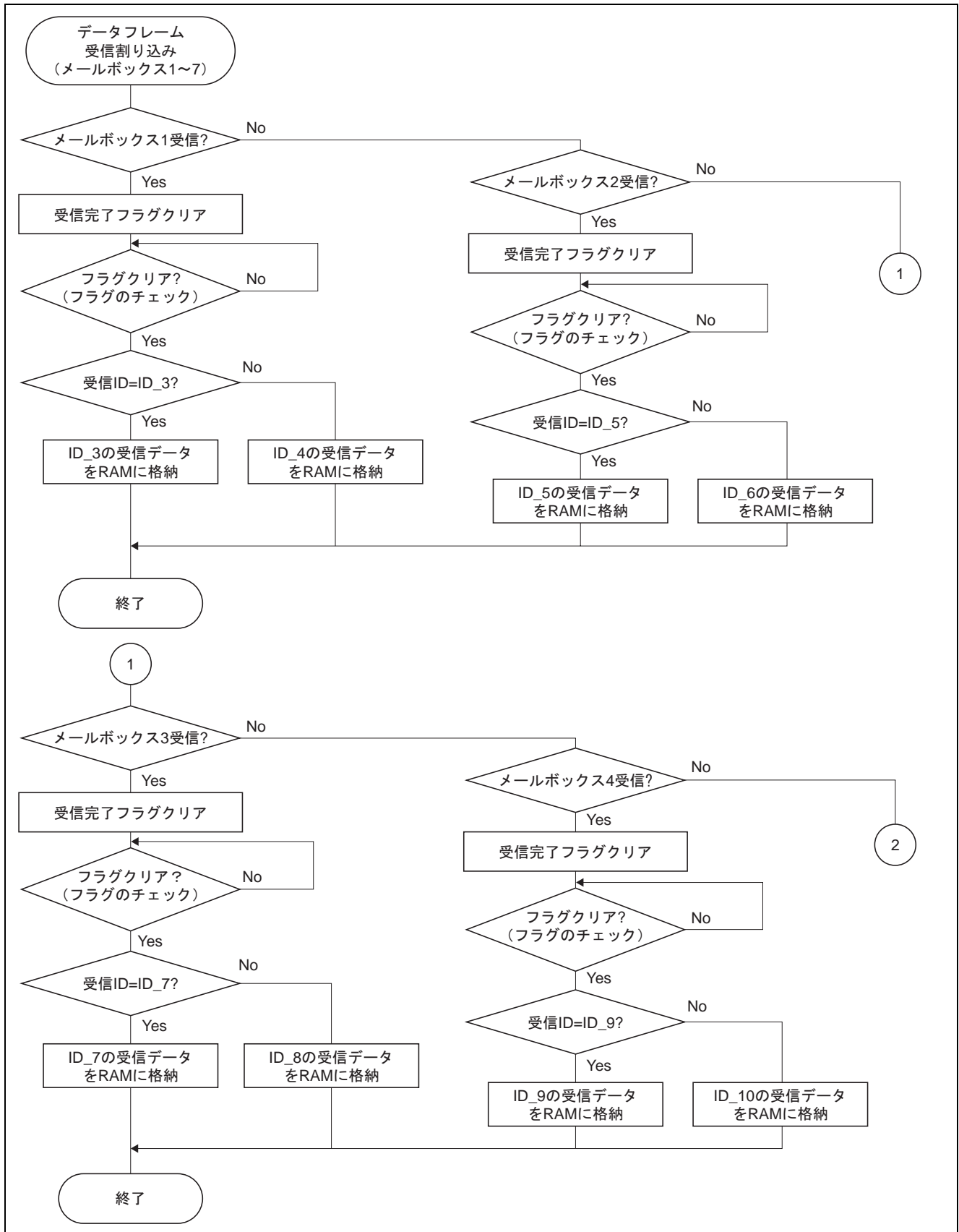


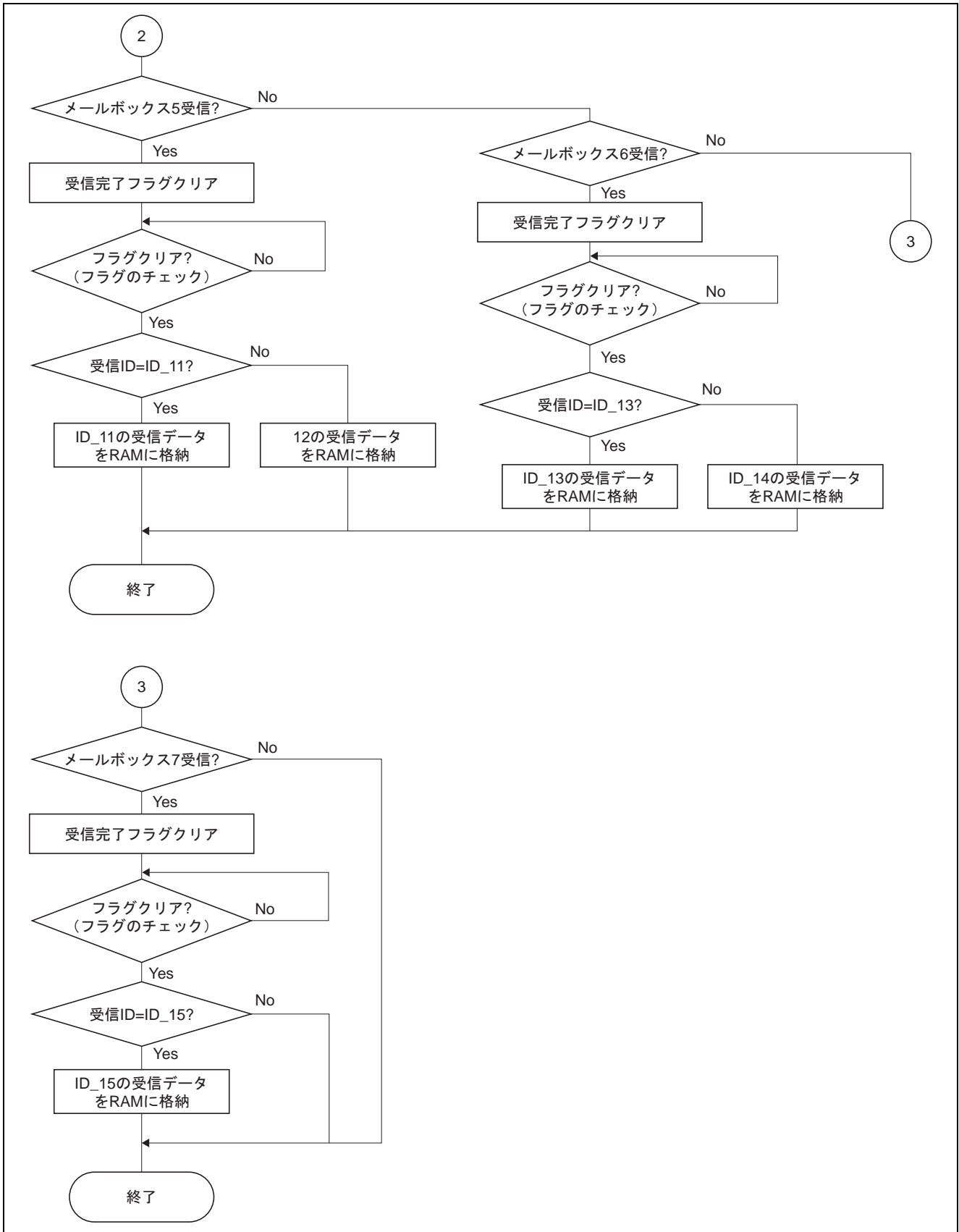
(b) データフレーム受信割り込みルーチン (メールボックス 0, RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(c) データフレーム受信割り込みルーチン (メールボックス1~7,RCAN-ET 共通設定)





【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

### 2.7.4 プログラムリスト

#### (1) 送信側プログラムリスト

##### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定     */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);
}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定     */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;                /* 割り込み制御モード 2      */
    INTC.IPRQ.BIT._RCAN01 = 7;            /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);                     /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
*****/
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*-----*/
#define ID_1      0x011
#define ID_2      0x0AA
#define ID_3      0x122
#define ID_4      0x1BB
#define ID_5      0x233
#define ID_6      0x2CC
#define ID_7      0x344
#define ID_8      0x3DD
#define ID_9      0x455
#define ID_10     0x4EE
#define ID_11     0x566
#define ID_12     0x5FF
#define ID_13     0x677
#define ID_14     0x6EE
#define ID_15     0x788
/*****
/*      RCAN Initialize routine
*****/
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックス割り込み許可          */
    RCANET0.MBIMR0.WORD = 0x0001;       /* MB1-15 割り込み許可                */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;     /* メールボックス 1 を送信用に設定      */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x01;  /* 送信データ                          */
    }

    RCANET0.MB[2].CTRL0.WORD.H = (ID_2<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[2].CTRL1.BYTE.H = 0x00;     /* メールボックス 2 を送信用に設定      */
    RCANET0.MB[2].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[2].MSG_DATA[i] = 0x02;  /* 送信データ                          */
    }

    RCANET0.MB[3].CTRL0.WORD.H = (ID_3<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[3].CTRL1.BYTE.H = 0x00;     /* メールボックス 3 を送信用に設定      */
    RCANET0.MB[3].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[3].MSG_DATA[i] = 0x03;  /* 送信データ                          */
    }

    RCANET0.MB[4].CTRL0.WORD.H = (ID_4<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[4].CTRL1.BYTE.H = 0x00;     /* メールボックス 4 を送信用に設定      */
    RCANET0.MB[4].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[4].MSG_DATA[i] = 0x04;  /* 送信データ                          */
    }

    RCANET0.MB[5].CTRL0.WORD.H = (ID_5<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[5].CTRL1.BYTE.H = 0x00;     /* メールボックス 5 を送信用に設定      */
    RCANET0.MB[5].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[5].MSG_DATA[i] = 0x05;  /* 送信データ                          */
    }

    RCANET0.MB[6].CTRL0.WORD.H = (ID_6<<2); /* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[6].CTRL1.BYTE.H = 0x00;     /* メールボックス 6 を送信用に設定      */
    RCANET0.MB[6].CTRL1.BYTE.L = 0x08;     /* データ長:8byte                      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[6].MSG_DATA[i] = 0x06;  /* 送信データ                          */
    }
}

```



```

RCANET0.MB[7].CTRL0.WORD.H = (ID_7<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[7].CTRL1.BYTE.H = 0x00; /* メルッス 7 を送信用に設定 */
RCANET0.MB[7].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[7].MSG_DATA[i] = 0x07; /* 送信データ */
}

RCANET0.MB[8].CTRL0.WORD.H = (ID_8<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[8].CTRL1.BYTE.H = 0x00; /* メルッス 8 を送信用に設定 */
RCANET0.MB[8].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[8].MSG_DATA[i] = 0x08; /* 送信データ */
}

RCANET0.MB[9].CTRL0.WORD.H = (ID_9<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[9].CTRL1.BYTE.H = 0x00; /* メルッス 9 を送信用に設定 */
RCANET0.MB[9].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[9].MSG_DATA[i] = 0x09; /* 送信データ */
}

RCANET0.MB[10].CTRL0.WORD.H = (ID_10<<2); /*STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[10].CTRL1.BYTE.H = 0x00; /* メルッス 10 を送信用に設定 */
RCANET0.MB[10].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[10].MSG_DATA[i] = 0x0A; /* 送信データ */
}

RCANET0.MB[11].CTRL0.WORD.H = (ID_11<<2); /*STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[11].CTRL1.BYTE.H = 0x00; /* メルッス 11 を送信用に設定 */
RCANET0.MB[11].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[11].MSG_DATA[i] = 0x0B; /* 送信データ */
}

RCANET0.MB[12].CTRL0.WORD.H = (ID_12<<2); /*STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[12].CTRL1.BYTE.H = 0x00; /* メルッス 12 を送信用に設定 */
RCANET0.MB[12].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[12].MSG_DATA[i] = 0x0C; /* 送信データ */
}

RCANET0.MB[13].CTRL0.WORD.H = (ID_13<<2); /*STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[13].CTRL1.BYTE.H = 0x00; /* メルッス 13 を送信用に設定 */
RCANET0.MB[13].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[13].MSG_DATA[i] = 0x0D; /* 送信データ */
}
    
```

```

RCANET0.MB[14].CTRL0.WORD.H = (ID_14<<2); /*STDID 設定,スタンダードフォーマット,データレーム*/
RCANET0.MB[14].CTRL1.BYTE.H = 0x00;          /* メッセージ 14 を送信用に設定 */
RCANET0.MB[14].CTRL1.BYTE.L = 0x08;          /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[14].MSG_DATA[i] = 0x0E;        /* 送信データ */
}

RCANET0.MB[15].CTRL0.WORD.H = (ID_15<<2); /*STDID 設定,スタンダードフォーマット,データレーム*/
RCANET0.MB[15].CTRL1.BYTE.H = 0x00;          /* メッセージ 15 を送信用に設定 */
RCANET0.MB[15].CTRL1.BYTE.L = 0x08;          /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[15].MSG_DATA[i] = 0x0F;        /* 送信データ */
}

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/*      RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE; /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RCAN send message routine */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x0000FFFE; /* MB1-15 を送信待ち状態に設定 */

}

/*****
/*      Mailbox Empty Interrupt routine */
/*****
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{
    RCANET0.TXACK0.WORD = RCANET0.TXACK0.WORD; /* 送信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.TXACK0.WORD); /* フラグのチェック */

}
/*****

```

## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c
/* Written       :   '06/06/01  REV.1.00
/* Purpose       :   for H8SX/1544  RCAN-ET
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMクリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */
}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
*****/
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff[16];
/*-----*/
#define ID_1      0x011
#define ID_2      0x0AA
#define ID_3      0x122
#define ID_4      0x1BB
#define ID_5      0x233
#define ID_6      0x2CC
#define ID_7      0x344
#define ID_8      0x3DD
#define ID_9      0x455
#define ID_10     0x4EE
#define ID_11     0x566
#define ID_12     0x5FF
#define ID_13     0x677
#define ID_14     0x6EE
#define ID_15     0x788

#define STD_ID_MB0    0x000
#define STD_ID_MB1    0x100
#define STD_ID_MB2    0x200
#define STD_ID_MB3    0x300
#define STD_ID_MB4    0x400
#define STD_ID_MB5    0x500
#define STD_ID_MB6    0x600
#define STD_ID_MB7    0x700
    
```

```

#define STD_LAFM          0x0FF
#define IDE_LAFM         0x0000

/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAMエリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFFFFD;        /* データフレーム受信割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFF00;      /* MB0-7 割り込み許可 */

    RCANET0.MB[0].CTRL0.WORD.H = (STD_ID_MB0<<2);/* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[0].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2));/*STD_LAFM,IDE_LAFM の設定*/
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;        /* メールボックス0を受信用に設定 */

    RCANET0.MB[1].CTRL0.WORD.H = (STD_ID_MB1<<2);/* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[1].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2));/*STD_LAFM,IDE_LAFM の設定*/
    RCANET0.MB[1].CTRL1.BYTE.H = 0x02;        /* メールボックス1を受信用に設定 */

    RCANET0.MB[2].CTRL0.WORD.H = (STD_ID_MB2<<2);/* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[2].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2));/*STD_LAFM,IDE_LAFM の設定*/
    RCANET0.MB[2].CTRL1.BYTE.H = 0x02;        /* メールボックス2を受信用に設定 */

    RCANET0.MB[3].CTRL0.WORD.H = (STD_ID_MB3<<2);/* STDID 設定,スタンダードフォーマット,データフレーム*/
    RCANET0.MB[3].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2));/*STD_LAFM,IDE_LAFM の設定*/
    RCANET0.MB[3].CTRL1.BYTE.H = 0x02;        /* メールボックス3を受信用に設定 */

```

```

RCANET0.MB[4].CTRL0.WORD.H = (STD_ID_MB4<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[4].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2)); /*STD_LAFM, IDE_LAFM の設定*/
RCANET0.MB[4].CTRL1.BYTE.H = 0x02; /* メールボックス 4 を受信用に設定 */

RCANET0.MB[5].CTRL0.WORD.H = (STD_ID_MB5<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[5].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2)); /*STD_LAFM, IDE_LAFM の設定*/
RCANET0.MB[5].CTRL1.BYTE.H = 0x02; /* メールボックス 5 を受信用に設定 */

RCANET0.MB[6].CTRL0.WORD.H = (STD_ID_MB6<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[6].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2)); /*STD_LAFM, IDE_LAFM の設定*/
RCANET0.MB[6].CTRL1.BYTE.H = 0x02; /* メールボックス 6 を受信用に設定 */

RCANET0.MB[7].CTRL0.WORD.H = (STD_ID_MB7<<2); /* STDID 設定,スタンダードフォーマット,データレム*/
RCANET0.MB[7].LAFM.WORD.H = ((IDE_LAFM)|(STD_LAFM<<2)); /*STD_LAFM, IDE_LAFM の設定*/
RCANET0.MB[7].CTRL1.BYTE.H = 0x02; /* メールボックス 6 を受信用に設定 */

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/* RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/* RAM area Initialize routine */
/*****
void Clear_MBbuff(void){

    unsigned int i;

    for(i = 0;i < 16;i++){ /* データ格納用 RAM I/A初期化 */
        MBbuff[i].ID.LONG = 0;
        MBbuff[i].DATA.LONG[0] = 0;
        MBbuff[i].DATA.LONG[1] = 0;
    }

}

```

```

/*****
/*      Data Frame Received Interrupt routine(MailBox0)          */
/*****
#pragma interrupt(RM0_0)
void RM0_0(void)
{
    unsigned int i;
    unsigned short rcv_id;

    /* MB0      */
    if(RCANET0.RXPR0.WORD&0x0001){          /* IRR1 ㊦,MB0 受信          */
        RCANET0.RXPR0.WORD = 0x0001;      /* 受信完了㊦㊦㊦(㊦㊦条件:1㊦㊦) */
        while(RCANET0.RXPR0.WORD & 0x0001); /* ㊦㊦の㊦㊦          */

        rcv_id = RCANET0.MB[0].CTRL0.WORD.H;
        rcv_id = ((rcv_id>>2)&0x07FF);
        switch(rcv_id){
            case ID_1:                    /* 受信 ID=ID_1(0x011)      */
                for(i = 0;i < 8;i++){    /* 受信㊦㊦㊦を RAM に格納    */
                    MBbuff[1].DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i];
                }
                break;
            case ID_2:                    /* 受信 ID=ID_2(0x0AA)      */
                for(i = 0;i < 8;i++){    /* 受信㊦㊦㊦を RAM に格納    */
                    MBbuff[2].DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i];
                }
                break;
            default:
                break;
        }
    }
}

/*****
/*      Data Frame Received Interrupt routine(MailBox1~7)        */
/*****
#pragma interrupt(RM1_0)
void RM1_0(void)
{
    unsigned int i;
    unsigned short rcv_id;

    /* MB1      */
    if(RCANET0.RXPR0.WORD&0x0002){          /* IRR1 ㊦,MB1 受信          */
        RCANET0.RXPR0.WORD = 0x0002;      /* 受信完了㊦㊦㊦(㊦㊦条件:1㊦㊦) */
        while(RCANET0.RXPR0.WORD & 0x0002); /* ㊦㊦の㊦㊦          */

        rcv_id = RCANET0.MB[1].CTRL0.WORD.H;
        rcv_id = ((rcv_id>>2)&0x07FF);
    }
}

```



```

switch(rcv_id){
    case ID_3:                /* 受信 ID=ID_3(0x122)          */
        for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
            MBbuff[3].DATA.BYTE[i] = RCANET0.MB[1].MSG_DATA[i];
        }
        break;
    case ID_4:                /* 受信 ID=ID_4(0x1BB)          */
        for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
            MBbuff[4].DATA.BYTE[i] = RCANET0.MB[1].MSG_DATA[i];
        }
        break;
    default:
        break;
}
}
/* MB2 */
else if(RCANET0.RXPR0.WORD&0x0004){ /* IRR1 設定,MB2 受信          */
    RCANET0.RXPR0.WORD = 0x0004; /* 受信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0004); /* フラグのチェック          */

    rcv_id = RCANET0.MB[2].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);
    switch(rcv_id){
        case ID_5:            /* 受信 ID=ID_5(0x233)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[5].DATA.BYTE[i] = RCANET0.MB[2].MSG_DATA[i];
            }
            break;
        case ID_6:            /* 受信 ID=ID_6(0x2CC)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[6].DATA.BYTE[i] = RCANET0.MB[2].MSG_DATA[i];
            }
            break;
        default:
            break;
    }
}
}
/* MB3 */
else if(RCANET0.RXPR0.WORD&0x0008){ /* IRR1 設定,MB3 受信          */
    RCANET0.RXPR0.WORD = 0x0008; /* 受信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0008); /* フラグのチェック          */

    rcv_id = RCANET0.MB[3].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);

```

```

switch(rcv_id){
    case ID_7:                /* 受信 ID=ID_7(0x344)          */
        for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
            MBbuff[7].DATA.BYTE[i] = RCANET0.MB[3].MSG_DATA[i];
        }
        break;
    case ID_8:                /* 受信 ID=ID_8(0x3DD)          */
        for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
            MBbuff[8].DATA.BYTE[i] = RCANET0.MB[3].MSG_DATA[i];
        }
        break;
    default:
        break;
}
}
/* MB4 */
else if(RCANET0.RXPR0.WORD&0x0010){ /* IRR1 設定, MB4 受信          */
    RCANET0.RXPR0.WORD = 0x0010; /* 受信完了フラグクリア(クリア条件: 1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0010); /* フラグのチェック          */

    rcv_id = RCANET0.MB[4].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);
    switch(rcv_id){
        case ID_9:            /* 受信 ID=ID_9(0x455)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[9].DATA.BYTE[i] = RCANET0.MB[4].MSG_DATA[i];
            }
            break;
        case ID_10:           /* 受信 ID=ID_10(0x4EE)         */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[10].DATA.BYTE[i] = RCANET0.MB[4].MSG_DATA[i];
            }
            break;
        default:
            break;
    }
}
/* MB5 */
else if(RCANET0.RXPR0.WORD&0x0020){ /* IRR1 設定, MB5 受信          */
    RCANET0.RXPR0.WORD = 0x0020; /* 受信完了フラグクリア(クリア条件: 1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0020); /* フラグのチェック          */

    rcv_id = RCANET0.MB[5].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);
    switch(rcv_id){
        case ID_11:           /* 受信 ID=ID_11(0x566)         */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[11].DATA.BYTE[i] = RCANET0.MB[5].MSG_DATA[i];
            }
            break;
    }
}

```

```

        case ID_12:                /* 受信 ID=ID_12(0x5FF)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[12].DATA.BYTE[i] = RCANET0.MB[5].MSG_DATA[i];
            }
            break;
        default:
            break;
    }
}
/* MB6 */
else if(RCANET0.RXPR0.WORD&0x0040){ /* IRR1 設定,MB6 受信          */
    RCANET0.RXPR0.WORD = 0x0040; /* 受信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.RXPR0.WORD & 0x0040); /* フラグのチェック          */

    rcv_id = RCANET0.MB[6].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);
    switch(rcv_id){
        case ID_13:                /* 受信 ID=ID_13(0x677)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[13].DATA.BYTE[i] = RCANET0.MB[6].MSG_DATA[i];
            }
            break;
        case ID_14:                /* 受信 ID=ID_14(0x6EE)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[14].DATA.BYTE[i] = RCANET0.MB[6].MSG_DATA[i];
            }
            break;
        default:
            break;
    }
}
/* MB7 */
else if(RCANET0.RXPR0.WORD&0x0080){ /* IRR1 設定,MB7 受信          */
    RCANET0.RXPR0.WORD = 0x0080; /* 受信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.RXPR0.WORD & 0x0080); /* フラグのチェック          */

    rcv_id = RCANET0.MB[7].CTRL0.WORD.H;
    rcv_id = ((rcv_id>>2)&0x07FF);
    switch(rcv_id){
        case ID_15:                /* 受信 ID=ID_15(0x788)          */
            for(i = 0;i < 8;i++){ /* 受信データを RAM に格納      */
                MBbuff[15].DATA.BYTE[i] = RCANET0.MB[7].MSG_DATA[i];
            }
            break;
        default:
            break;
    }
}
}
}
/*****/

```

2.8 メッセージフィルタリング (LAFM) の IDE ビットマスク

2.8.1 仕様

図 2.8.1 に示すように、ノード A から、2 種類のスタンダードフォーマット (STD\_format) のメッセージと 2 種類のエクステンデッドフォーマット (EXT\_format) のメッセージを送信します。ノード B は、受信メッセージのフィルタリング機能 (LAFM) を使用して IDE ビットマスクを設定し、1 種類のメールアドレスで全メッセージを受信します。

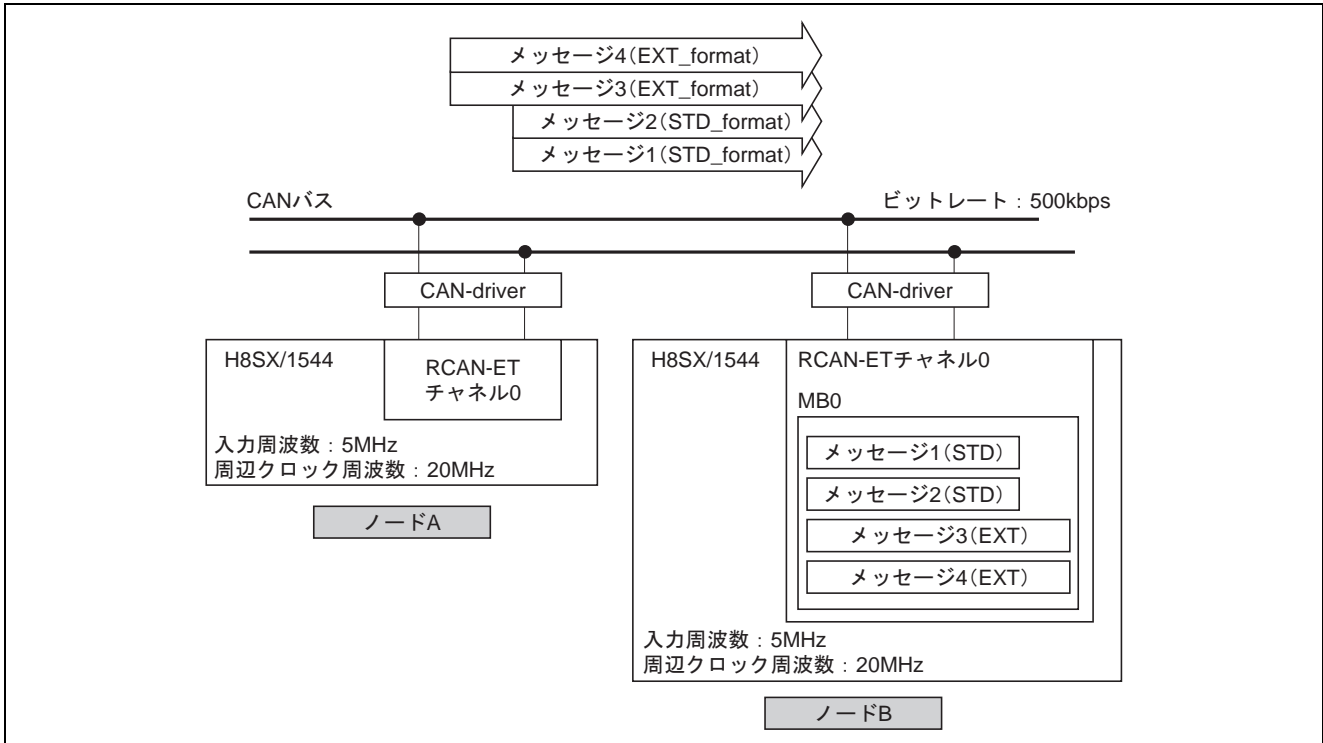


図 2.8.1 通信仕様

表 2.8.1 に、メッセージの仕様を示します。

表 2.8.1 メッセージ仕様

メッセージ	フォーマット	スタンダード ID	エクステンデッド ID	データ
メッセージ 1	スタンダード	H'0AA	-	H'01 01 01 01 01 01 01 01(8byte)
メッセージ 2	スタンダード	H'1BB	-	H'02 02 02 02 02 02 02 02(8byte)
メッセージ 3	エクステンデッド	H'2CC	H'2AAAA	H'03 03 03 03 03 03 03 03(8byte)
メッセージ 4	エクステンデッド	H'3DD	H'25555	H'04 04 04 04 04 04 04 04(8byte)

### 2.8.2 ソフトウェア説明

#### (1) モジュール説明

表 2.8.2 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
RCAN-ET 共通			
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
データフレーム 受信割り込み	RM0_0	CAN メッセージの受信完了フラグクリア。 受信データを RAM に格納。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの 初期化。	-

#### (2) 使用レジスタ説明

##### (a) 送信側

表 2.8.3 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み (IRR8) を許可に設定。(IMR8=0)
	RCANET0.MBIMR0.WORD	0xFFE1	メールボックス 1~4 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x02A8	メールボックス 1 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID (H'0AA) を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0] ~ RCANET0.MB[1].MSG_DATA[7]	0x01 × 8	送信データを設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[2].CTRL0.WORD.H	0x06EC	メールボックス 2 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID (H'1BB) を設定。
	RCANET0.MB[2].CTRL1.BYTE.H	0x00	メールボックス 2 を送信用に設定。
	RCANET0.MB[2].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[2].MSG_DATA[0] ~ RCANET0.MB[2].MSG_DATA[7]	0x02 × 8	送信データを設定。
	RCANET0.MB[3].CTRL0.WORD.H	0x8B32	メールボックス 3 をエクステンデッドフォーマット、データフレームに設定。 また、スタンダード ID (H'2CC) とエクステンデッド ID (H'2AAAA) を設定。
	RCANET0.MB[3].CTRL0.WORD.L	0xAAAA	
	RCANET0.MB[3].CTRL1.BYTE.H	0x00	メールボックス 3 を送信用に設定。
	RCANET0.MB[3].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[3].MSG_DATA[0] ~ RCANET0.MB[3].MSG_DATA[7]	0x03 × 8	送信データを設定。
	RCANET0.MB[4].CTRL0.WORD.H	0x8F76	メールボックス 4 をエクステンデッドフォーマット、データフレームに設定。 また、スタンダード ID (H'3DD) とエクステンデッド ID (H'25555) を設定。
	RCANET0.MB[4].CTRL0.WORD.L	0x5555	
	RCANET0.MB[4].CTRL1.BYTE.H	0x00	メールボックス 4 を送信用に設定。
	RCANET0.MB[4].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[4].MSG_DATA[0] ~ RCANET0.MB[4].MSG_DATA[7]	0x04 × 8	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x0000001E	メールボックス 1~4 を送信待ち状態に設定。
メールボックス エンプティ 割り込み	RCANET0.TXACK0.WORD	-	メールボックス 1~4 の送信完了フラグのクリア。 (クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

## (b) 受信側

表 2.8.4 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFFFD	データフレーム受信割り込み (IRR1) を許可に設定。(IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFE	メールボックス 0 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	0x0000	メールボックス 0 をスタンダードフォーマット、データフレームに設定。
	RCANET0.MB[0].CTRL0.WORD.L	0x0000	
	RCANET0.MB[0].LAFM.WORD.H	0x9FFF	メールボックス 0 の全 ID ビット(スタンダード ID、エクステンデッド ID、IDE ビット)を無効に設定。 任意のフォーマット/ID のメッセージをメールボックス 0 で受信可。
	RCANET0.MB[0].LAFM.WORD.L	0xFFFF	
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	P <sub>φ</sub> =20MHz 時 500kbps に設定。 (TSEG1=4(5tq),TSEG2=3(4tq),SJW=0, BSP=0,BRP=1)
RCANET0.BCR0.WORD	0x0001		
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
データフレーム 受信割り込み	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

## (3) 使用 RAM 説明

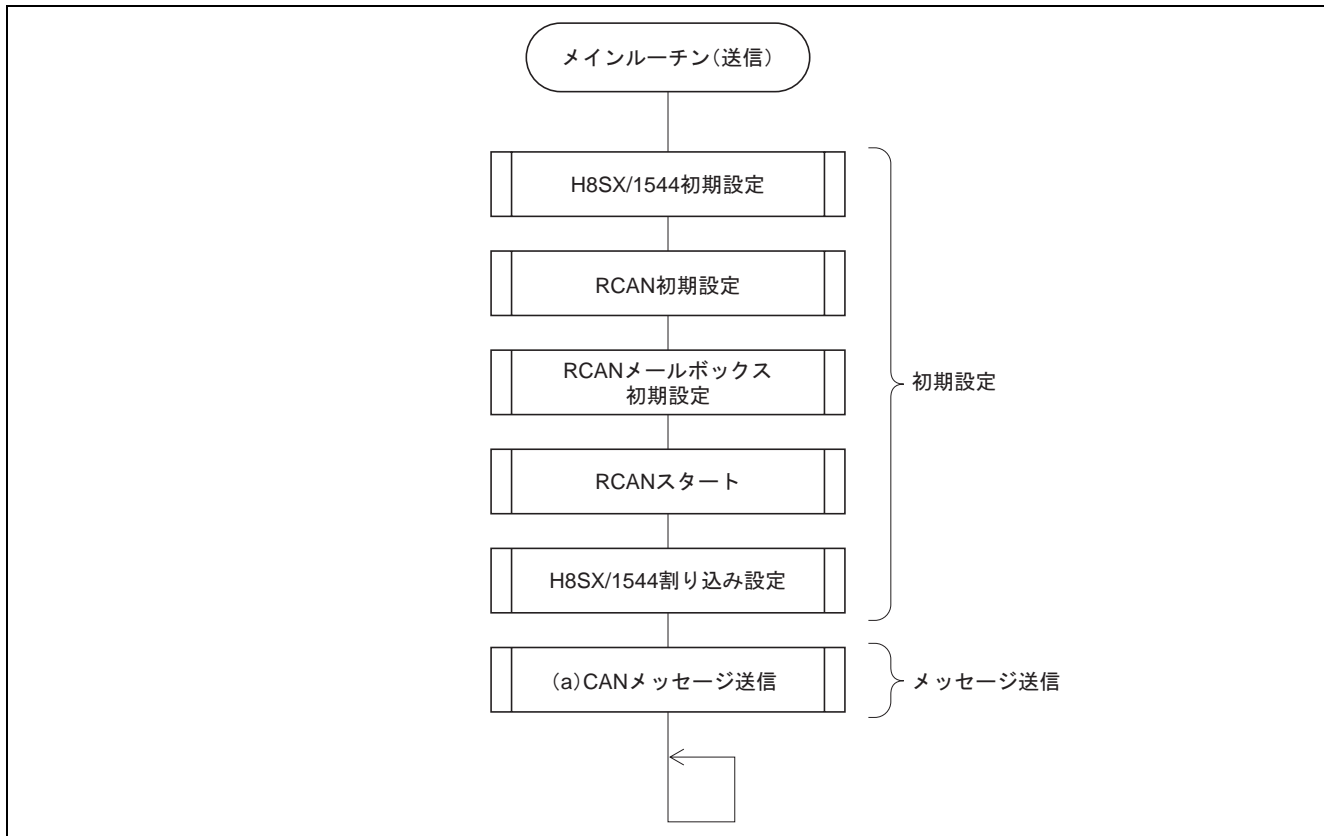
表 2.8.5 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ 初期化	Mbbuff[4].ID.LONG MBbuff[4].ID.WORD.H MBbuff[4].ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuff[4].DATA.LONG[0] ~ [1] MBbuff[4].DATA.WORD[0] ~ [3] MBbuff[4].DATA.BYTE[0] ~ [7]	受信データを格納します。
メインルーチン データフレーム 受信割り込み	Rcv_cnt	受信メッセージ数を示します。

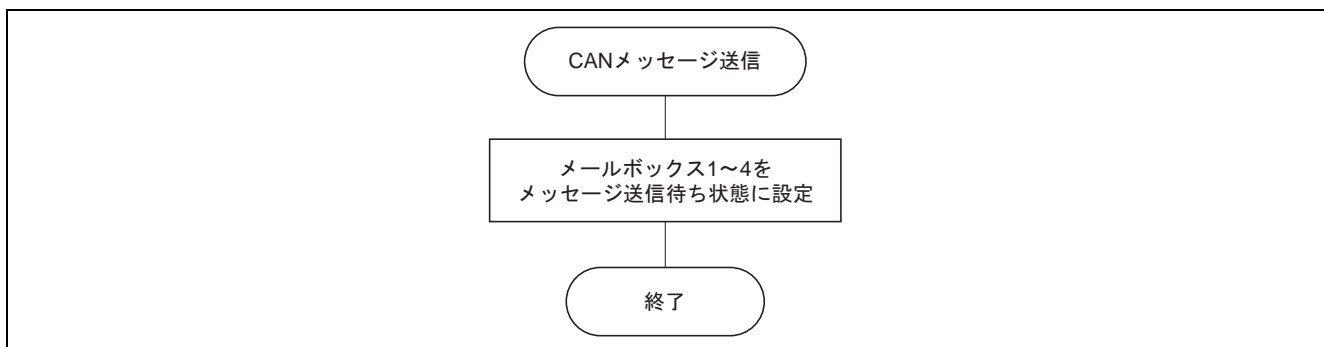
2.8.3 フローチャート

(1) 送信側フローチャート

メインルーチン

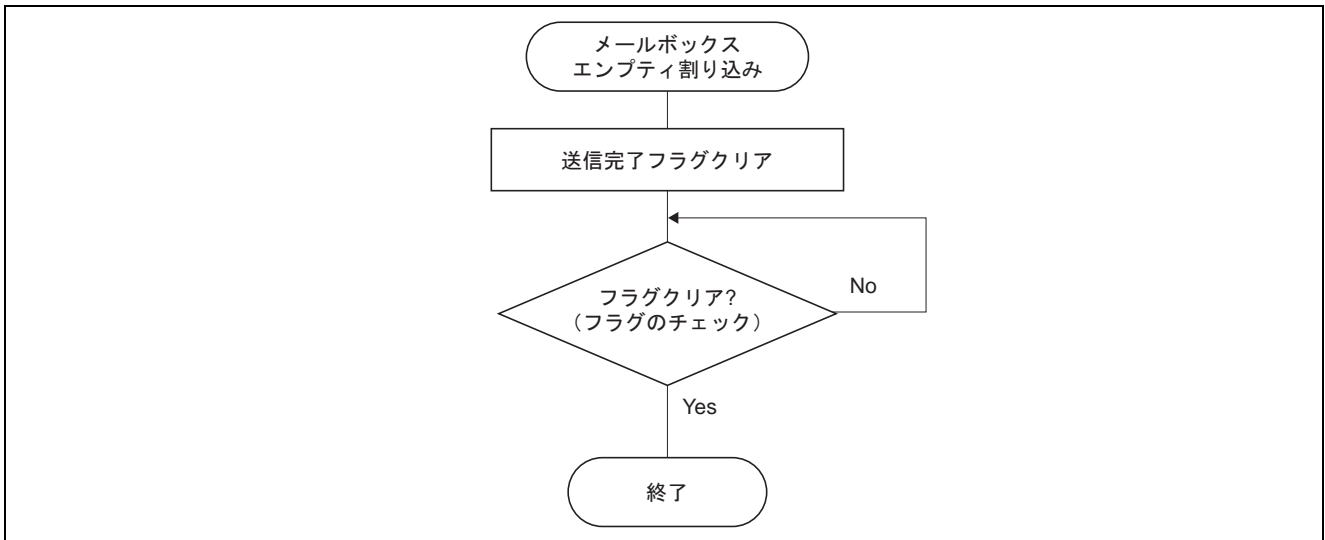


(a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)





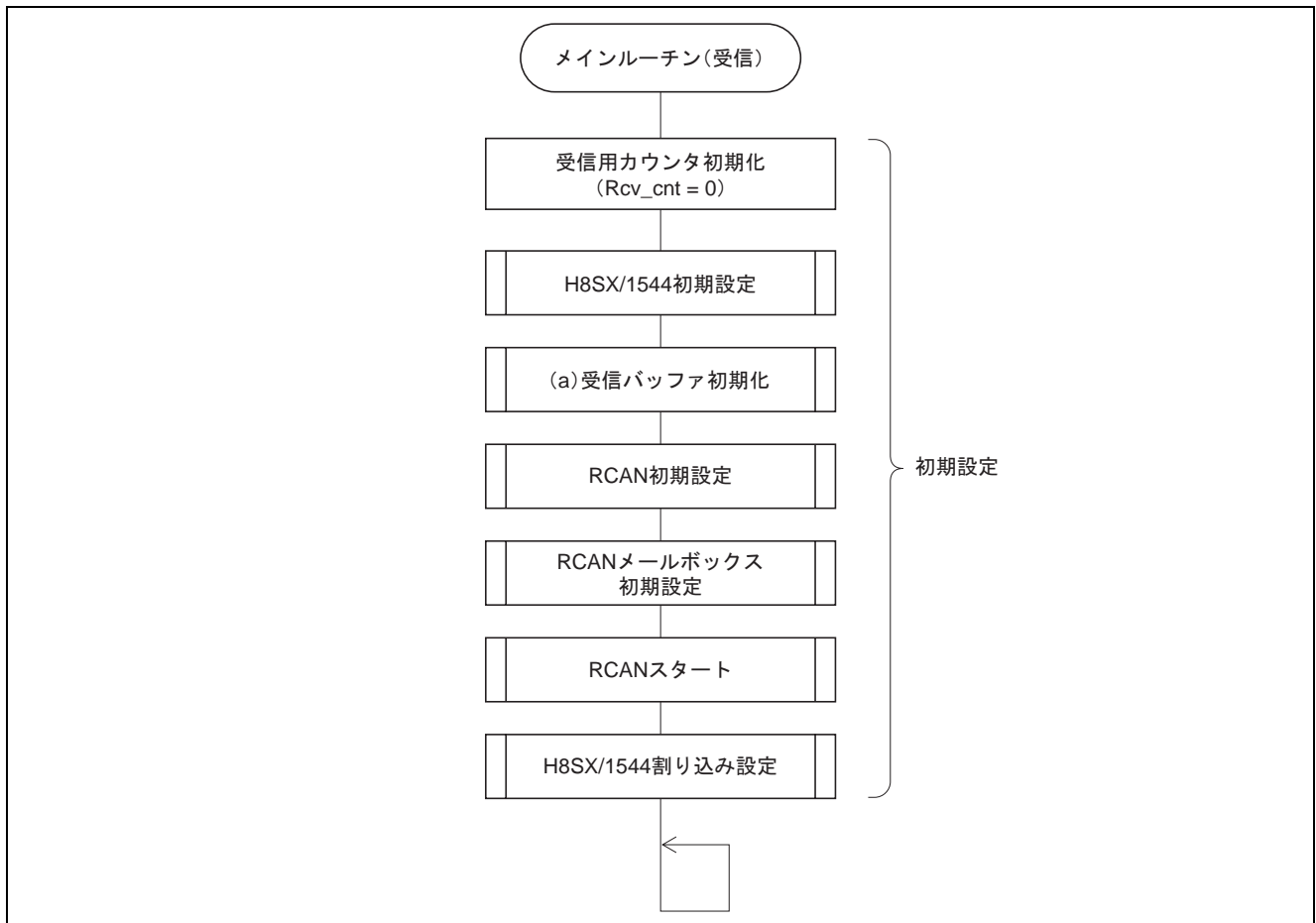
(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



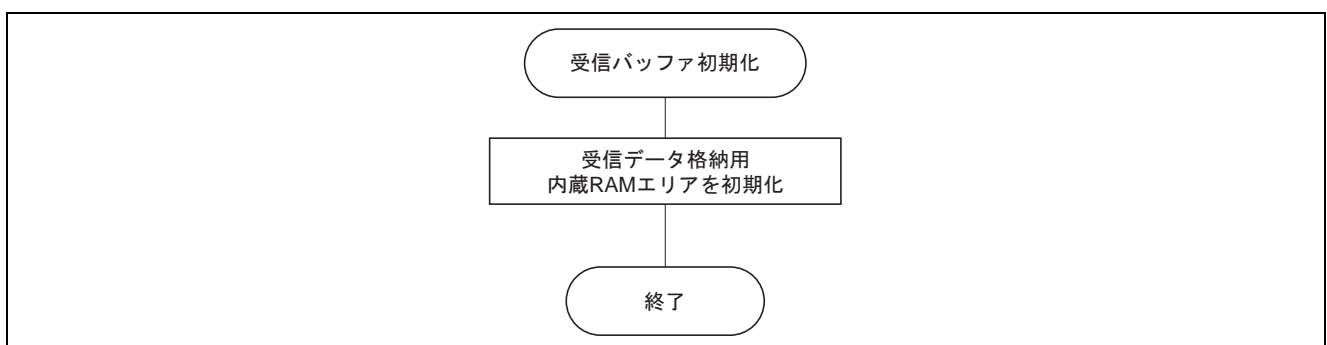
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

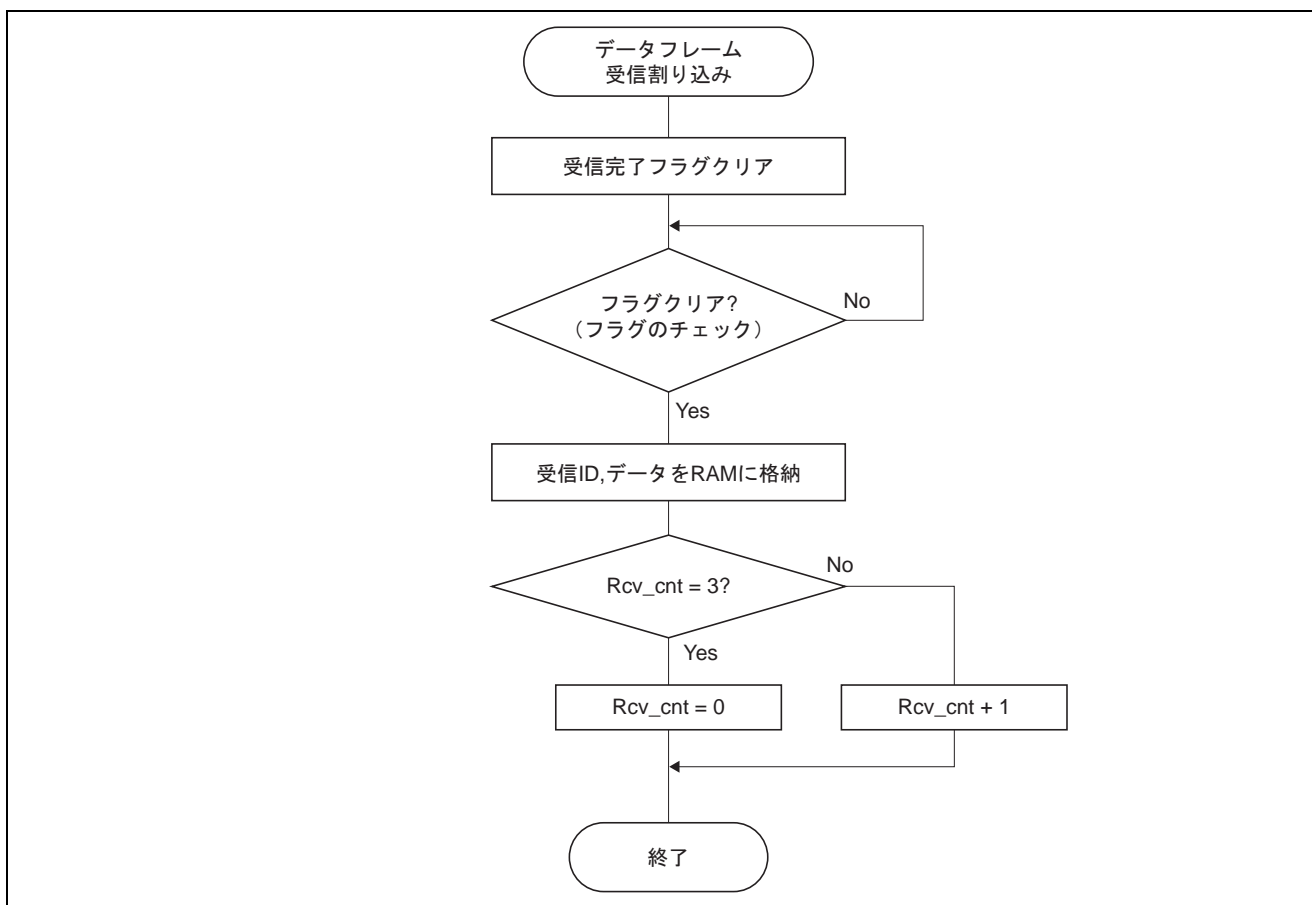
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)



(b) データフレーム受信割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

## 2.8.4 プログラムリスト

### (1) 送信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{

    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定   */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);

}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;        /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;        /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;       /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定       */

}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose      : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックスインプティ割り込み許可 */
    RCANET0.MBIMR0.WORD = 0xFFE1;       /* MB1-4 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x02A8; /* STDID:H'0AA,スタンダードフォーマット,データレム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;   /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;   /* データ長:8byte */
    for(i = 0;i < 8;i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x01; /* 送信データ */
    }
}

```

```

RCANET0.MB[2].CTRL0.WORD.H = 0x06EC; /* STDID:H'1BB,スタンダードフォーマット,データレム */
RCANET0.MB[2].CTRL1.BYTE.H = 0x00; /* メルックス2を送信用に設定 */
RCANET0.MB[2].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[2].MSG_DATA[i] = 0x02; /* 送信データ */
}

RCANET0.MB[3].CTRL0.WORD.H = 0x8B32; /* STDID:H'2CC,エクステンデッドフォーマット,データレム*/
RCANET0.MB[3].CTRL0.WORD.L = 0xAAAA; /* EXTID:H'2AAAA */
RCANET0.MB[3].CTRL1.BYTE.H = 0x00; /* メルックス3を送信用に設定 */
RCANET0.MB[3].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[3].MSG_DATA[i] = 0x03; /* 送信データ */
}

RCANET0.MB[4].CTRL0.WORD.H = 0x8F76; /* STDID:H'3DD,エクステンデッドフォーマット,データレム*/
RCANET0.MB[4].CTRL0.WORD.L = 0x5555; /* EXTID:H'25555 */
RCANET0.MB[4].CTRL1.BYTE.H = 0x00; /* メルックス4を送信用に設定 */
RCANET0.MB[4].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[4].MSG_DATA[i] = 0x04; /* 送信データ */
}

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/* RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/* RCAN send message routine */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x0000001E; /* MB1-4を送信待ち状態に設定 */

}
    
```

```

/*****/
/*      Mailbox Empty Interrupt routine      */
/*****/
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{
    RCANETO.TXACK0.WORD = RCANETO.TXACK0.WORD; /* 送信完了フラグクリア(クリア条件:1ライト) */
    while(RCANETO.TXACK0.WORD);                /* フラグのチェック */
}
/*****/

```



## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c
/* Written       :   '06/06/01  REV.1.00
/* Purpose       :   for H8SX/1544  RCAN-ET
*****/
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*-----*/
extern unsigned char Rcv_cnt;
/*****
/*      Main program
*****/
#pragma entry main(sp=0xFFC000)
void main(void)
{
    Rcv_cnt = 0;                /*  受信カウンタ初期化          */

    set_1544_init();           /*  H8SX/1544  初期化          */
    Clear_MBbuff();           /*  データ格納用RAMエリア初期化  */
    set_RCAN0_init();         /*  RCAN  初期化              */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /*  H8SX/1544  割り込み設定    */
    while(1);

}
/*****
/*      H8SX/1544 Initialize routine
*****/
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /*  システムクロック設定(5MHz×8=40MHz)  */
    SCKCR.BIT.PCK = 1;        /*  周辺クロック(5MHz×4=20MHz)          */
    SCKCR.BIT.BCK = 1;        /*  外部バスクロック(5MHz×4=20MHz)      */
    SBYCR.BIT.SSBY = 1;       /*  ソフトウェアスタンバイモード        */
    MSTP.CRC.BIT._RCAN01 = 0; /*  モジュールストップモード解除:RCAN  */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /*  RCAN-ET 送受信端子レベル          */
    P6.ICR.BIT.B4 = 1;         /*  P64 を CRx_0(入力端子)に設定      */
}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff[4];
/*-----*/
unsigned char Rcv_cnt;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFFFD;           /* データレム受信割り込み許可      */
    RCANET0.MBIMR0.WORD &= 0xFFFE;       /* MB0 割り込み許可                */

    RCANET0.MB[0].CTRL0.WORD.H = 0x0000; /* STD_ID, データレム              */
    RCANET0.MB[0].CTRL0.WORD.L = 0x0000; /* EXT_ID                          */
    RCANET0.MB[0].LAFM.WORD.H = 0x9FFF;  /* STD_LAFM, IDE_LAFM の設定      */
    RCANET0.MB[0].LAFM.WORD.L = 0xFFFF;  /* EXT_LAFM の設定                */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;    /* メルボックス0を受信用に設定    */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz)*/
    RCANET0.BCR0.WORD = 0x0001;      /* BRP=1                            */

}

/*****
/*      RCAN start routine      */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;         /* MCR0 クリア                      */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?                          */

}

/*****
/*      RAM area Initialize routine      */
/*****
void Clear_MBbuff(void){

    unsigned int i;

    for(i = 0; i < 4; i++){             /* データ格納用 RAM クリア初期化    */
        MBbuff[i].ID.LONG = 0;
        MBbuff[i].DATA.LONG[0] = 0;
        MBbuff[i].DATA.LONG[1] = 0;
    }

}

```

```

/*****
/*      Data Frame Received Interrupt routine(MailBox0)      */
/*****
#pragma interrupt(RM0_0)
void RM0_0(void)
{
    unsigned int i;

    RCANET0.RXPR0.WORD = 0x0001;          /* 受信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0001);  /* フラグのチェック */

    MBbuff[Rcv_cnt].ID.WORD.H = RCANET0.MB[0].CTRL0.WORD.H; /* 受信 ID を RAM に格納 */
    MBbuff[Rcv_cnt].ID.WORD.L = RCANET0.MB[0].CTRL0.WORD.L; /* 受信 ID を RAM に格納 */
    for(i = 0; i < 8; i++){              /* 受信データを RAM に格納 */
        MBbuff[Rcv_cnt].DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i];
    }

    if(Rcv_cnt == 3){
        Rcv_cnt = 0;
    }
    else{
        Rcv_cnt++;
    }
}
/*****

```

## 2.9 CAN wake-up 機能

### 2.9.1 仕様

図 2.9.1 に示すように、ノード A とノード B を初期設定後、ノード B のみソフトウェアスタンバイモードに移します。

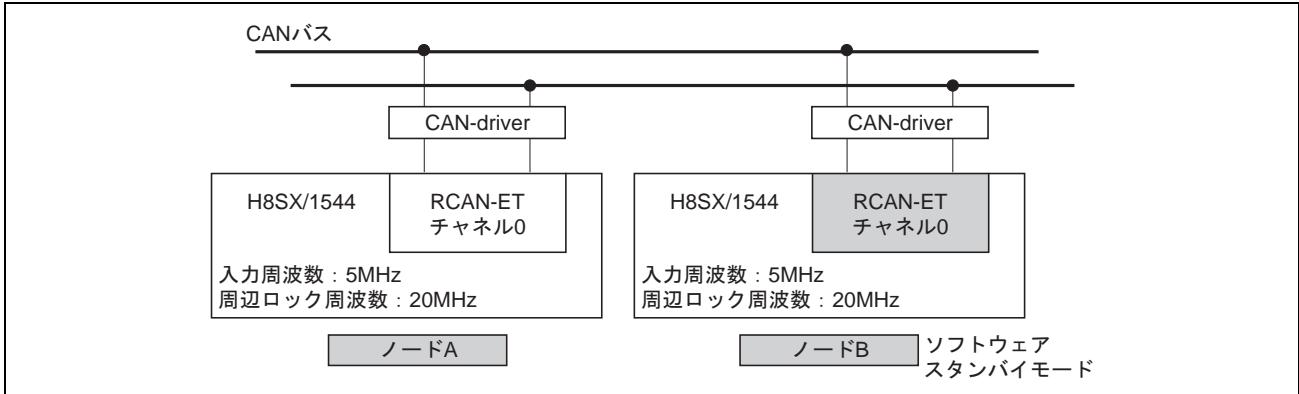


図 2.9.1 通信仕様 (1)

次に、図 2.9.2 に示すように、ノード A からスタンダード ID : H'555 のデータフレームを送信します。ノード B は、ポート 64 (CRx\_0/IRQ12 兼用) の Low レベル検出で IRQ 割り込みが発生し、ソフトウェアスタンバイモードから復帰します。復帰後、RCAN-ET の初期設定を行い、データフレームを受信します。

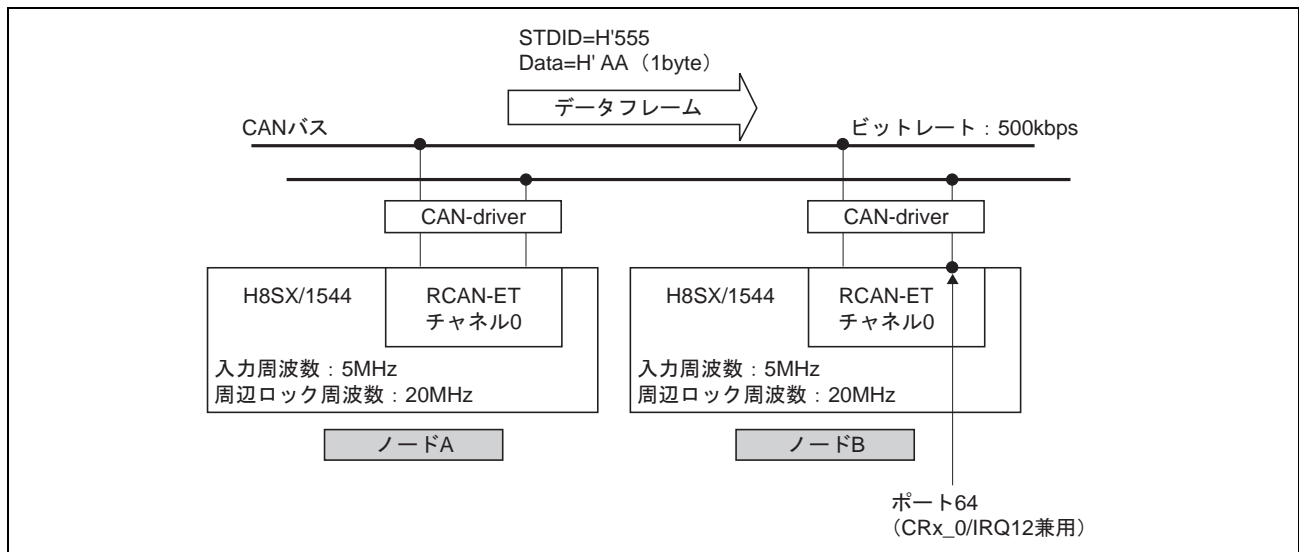


図 2.9.2 通信仕様 (2)

**【注】** ソフトウェアスタンバイモードでは RCAN-ET はリセット状態になりますので、ソフトウェアスタンバイモードから復帰後、RCAN-ET の初期設定が必要となります。ソフトウェアスタンバイモード時の CPU や周辺モジュールの動作状態については、「H8SX/1544 ハードウェアマニュアル 第 24 章 低消費電力」を参照してください。

## 2.9.2 ソフトウェア説明

## (1) モジュール説明

表 2.9.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init	H8SX/1544 の RCAN-ET、IRQ 割り込み優先レベル設定。ソフトウェアスタンバイ解除 IRQ 設定。	
H8SX/1544 入出力端子設定	set_1544_IO_init	H8SX/1544 の RCAN-ET 送受信端子の設定。	
IRQ 割り込み	IRQ12	H8SX/1544 の I/O および RCAN-ET の初期設定。	
RCAN 初期設定	set_RCAN0_init	「2.1 初期設定」参照	RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
データフレーム 受信割り込み	RM0_0	CAN メッセージの受信完了フラグクリア。 受信データを RAM に格納。	
受信バッファ初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	

## (2) 使用レジスタ説明

## (a) 送信側

表 2.9.2 使用レジスタ説明（送信側）

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み（IRR8）を許可に設定。（IMR8=0）
	RCANET0.MBIMR0.WORD	0xFFFFD	メールボックス 1 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス 1 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID（H'555）を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x01	データ長（1byte）を設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[1].MSG_DATA[0]	0xAA	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。
メールボックス エンpty 割り込み	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

(b) 受信側

表 2.9.3 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メール ボックス 初期設定	RCANET0.IMR.WORD	0xFFFFD	データフレーム受信割り込み (IRR1) を許可に設定。(IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFFE	メールボックス 0 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	0x1554	メールボックス 0 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID (H'555) を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み 設定	INTC.INTCR.BIT.INTM	2	割り込みコントローラの割り込み制御モードを割り込み制御モード 2 に設定。
	INTC.IPRD.BIT._IRQ12	7	IRQ12 の割り込み優先レベルを設定。
	INTC.IPRQ.BIT._RCAN01	6	RCAN-ET の割り込み優先レベルを設定。
	INTC.SSIER.BIT.SS12	1	ソフトウェアスタンバイ解除 IRQ を設定。 ソフトウェアスタンバイ状態で IRQ12 割り込みが発生すると、発振安定時間経過後、ソフトウェアスタンバイ状態から復帰する。
	INTC.IER.BIT.IRQ12E	1	IRQ12 割り込み要求を許可。



モジュール名	レジスタ名	設定値	機能
IRQ 割り込み	INTC.IER.BIT.IRQ12E	0	IRQ12 割り込み要求を禁止。
H8SX/1544 入出力端子 設定	RCANET0.RCANMON.BYTE	0x20	RCAN-ET の送受信端子をイネーブルに設定。
	P6.ICR.BIT.B4	1	P64 (67Pin) を RCAN-ET チャンネル 0 の受信端子 (CRx_0) に設定。
データ フレーム 受信 割り込み	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

### (3) 使用 RAM 説明

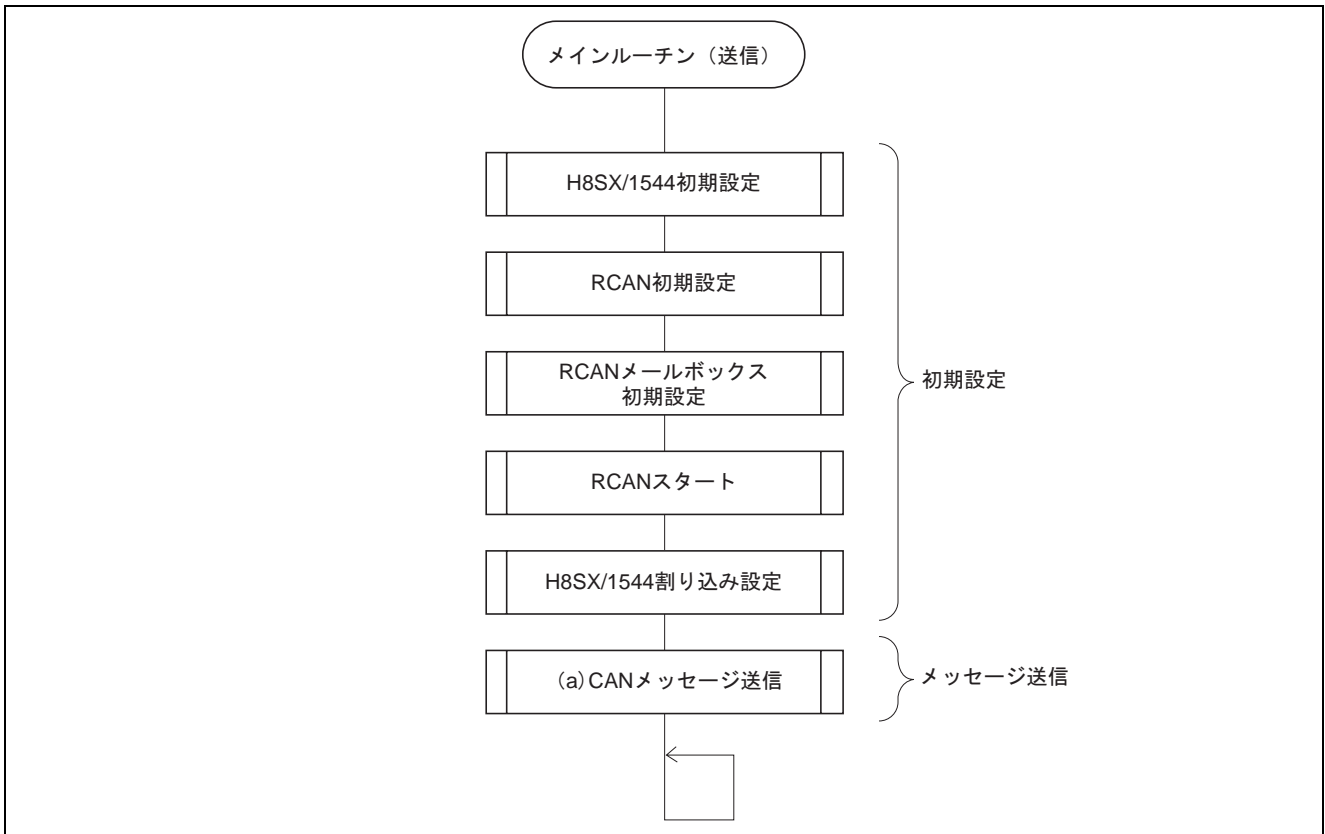
表 2.9.4 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ 初期化	Mbbuf.ID.LONG MBbuf.ID.WORD.H MBbuf.ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuf.DATA.LONG[0] ~ [1] MBbuf.DATA.WORD[0] ~ [3] MBbuf.DATA.BYTE[0] ~ [7]	受信データを格納します。

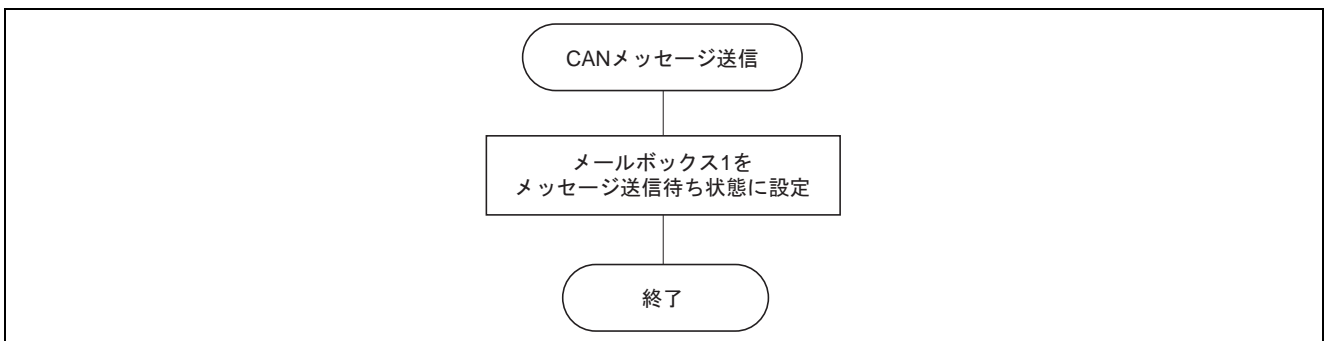
2.9.3 フローチャート

(1) 送信側フローチャート

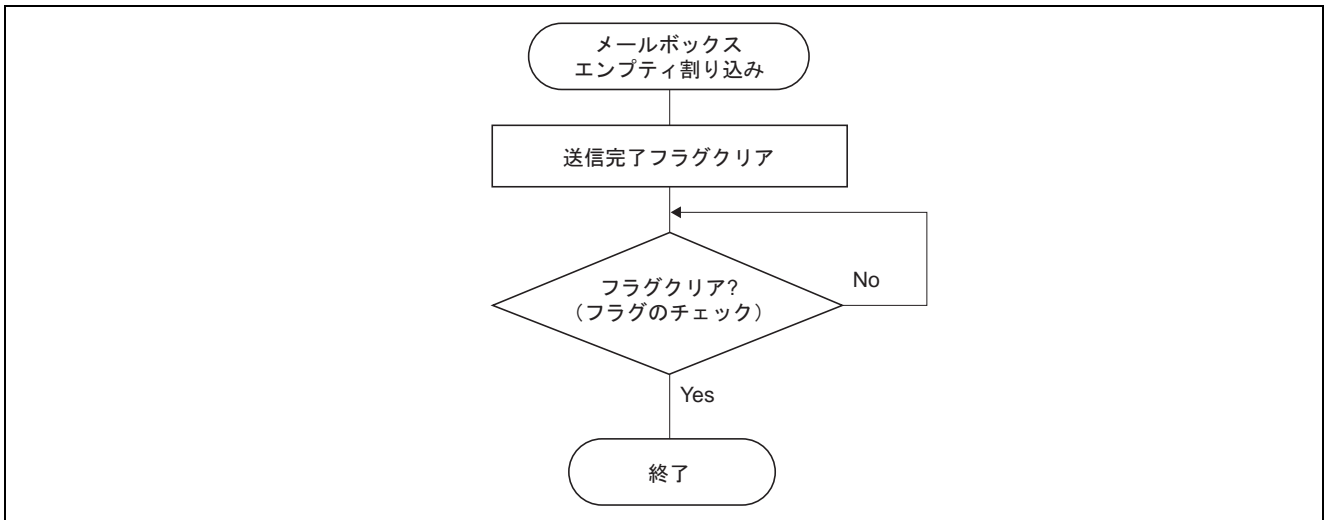
メインルーチン



(a) CANメッセージ送信ルーチン (RCAN-ET 共通設定)



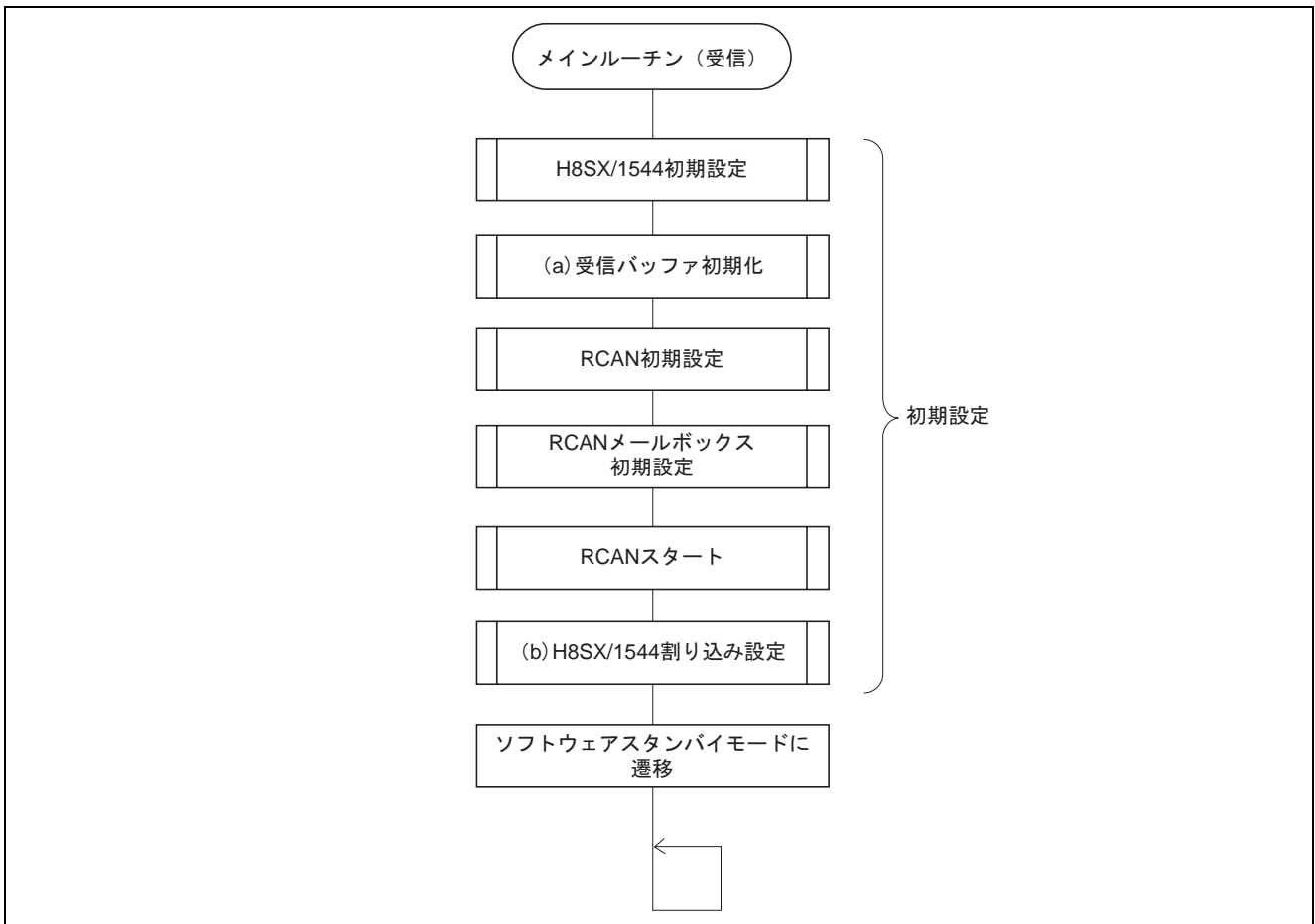
(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



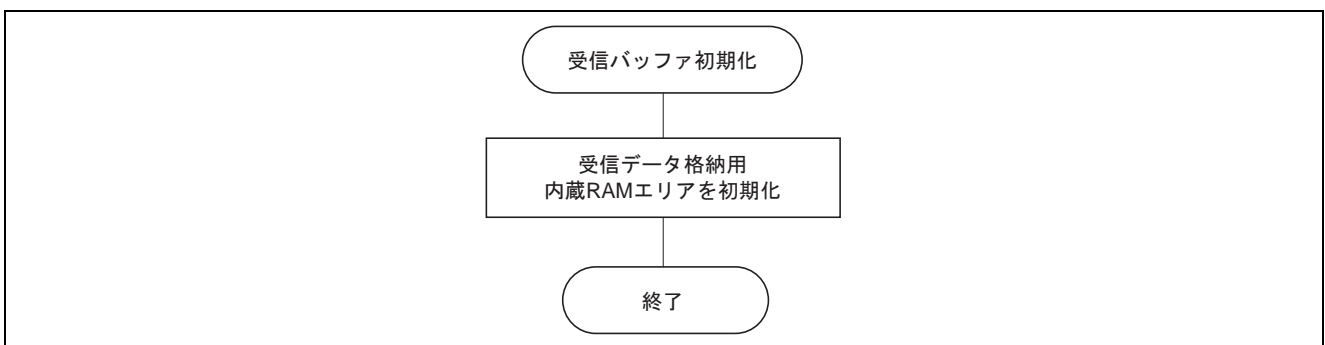
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

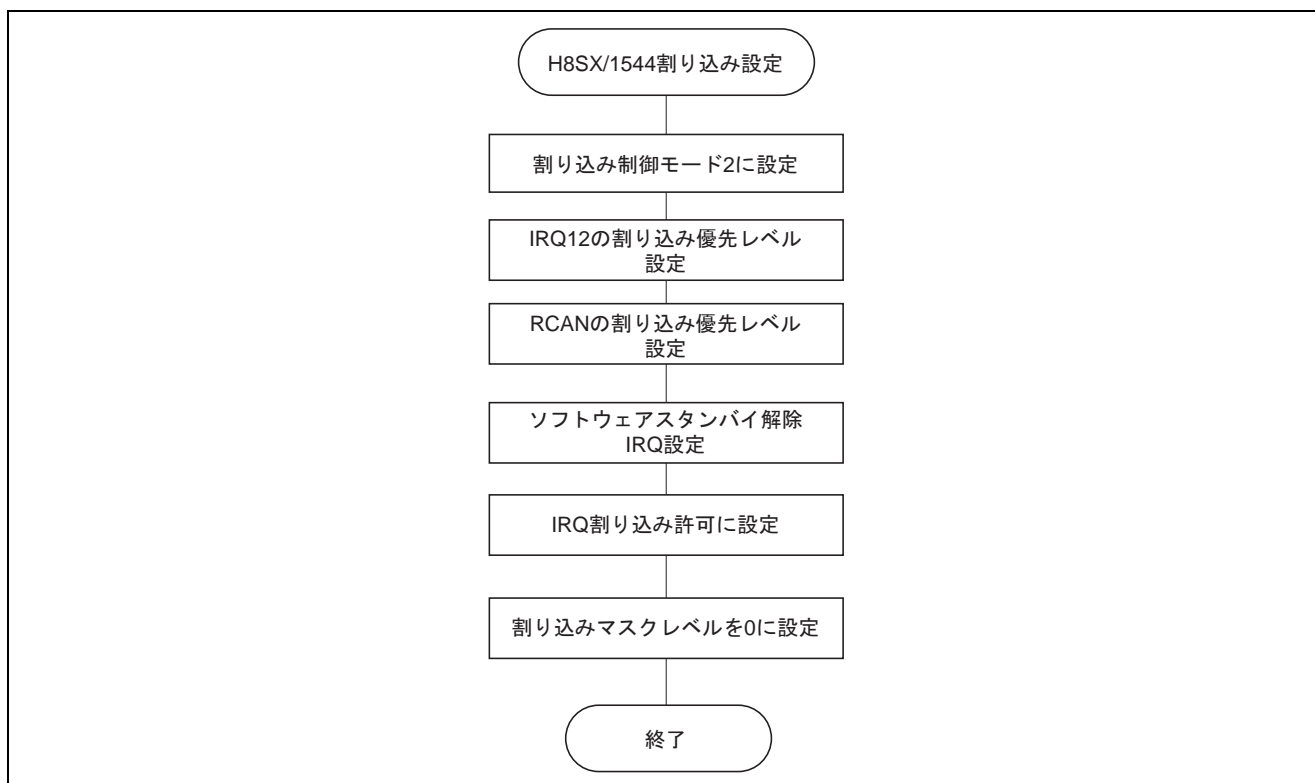
メインルーチン



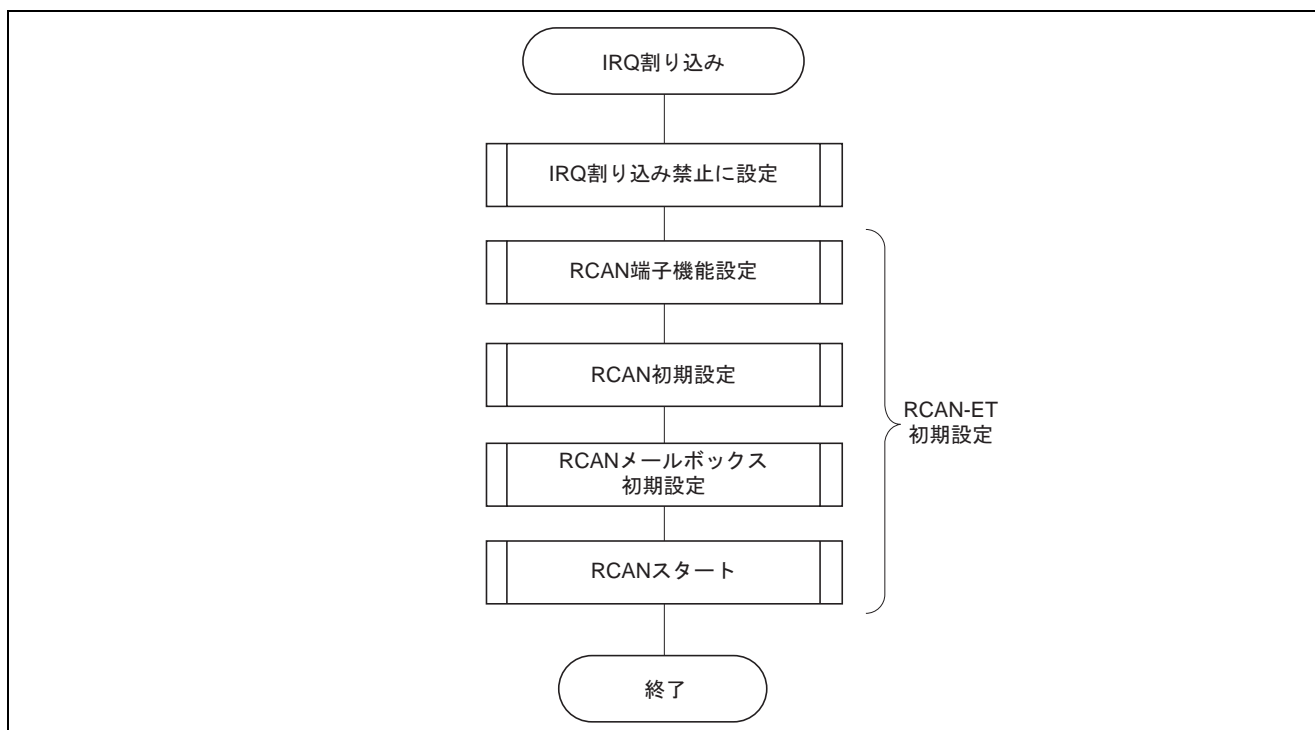
(a) 受信バッファ初期化ルーチン (デバイス個別設定)



### (b) H8SX/1544 割り込み設定ルーチン (デバイス個別設定)

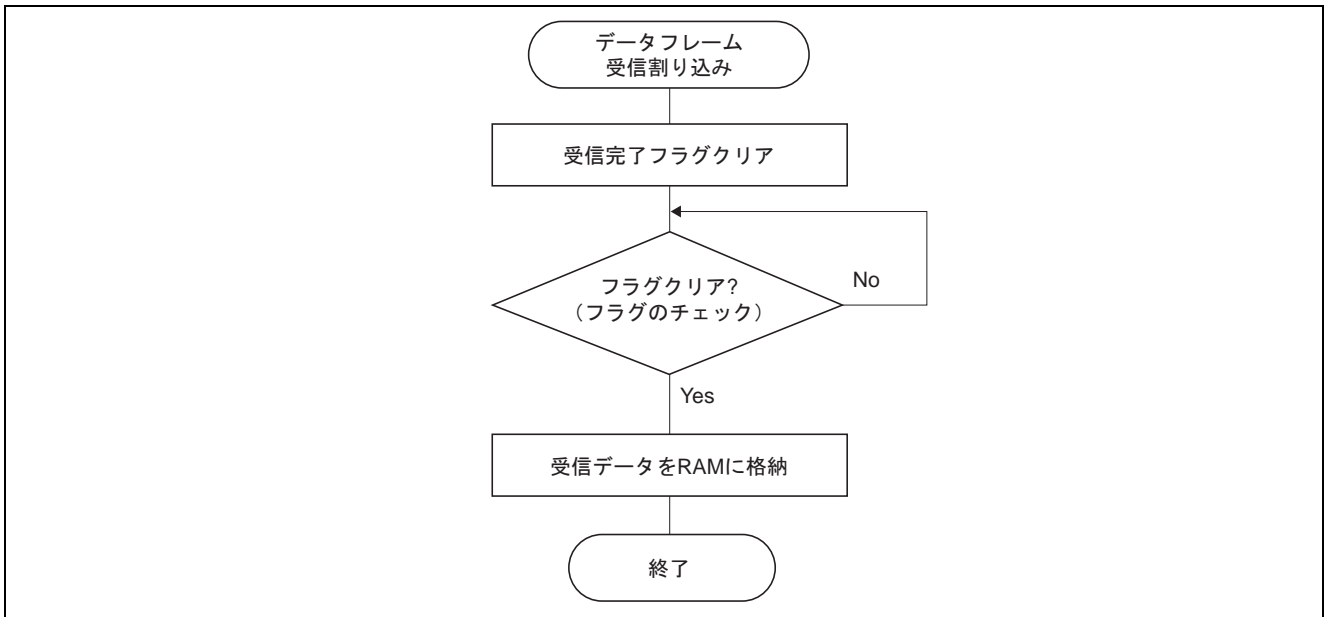


### (c) IRQ 割り込みルーチン (デバイス個別設定)



【注】 IRQ 割り込み発生後、RCAN 初期設定終了までは CAN メッセージの送受信はできません。

(d) データフレーム受信割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

### 2.9.4 プログラムリスト

#### (1) 送信側プログラムリスト

##### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定   */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);
}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0(入力端子)に設定     */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```



## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose      : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAM I/O)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFF;          /* メールボックスインポート割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554; /* ID:H'555,スタンダードフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;   /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x01;   /* データ長:1byte */
    RCANET0.MB[1].MSG_DATA[0] = 0xAA;    /* 送信データ:0xAA */

```

```

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****/
/* RCAN start routine */
/*****/
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****/
/* RCAN send message routine */
/*****/
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002; /* MB1 を送信待ち状態に設定 */

}

/*****/
/* Mailbox Empty Interrupt routine */
/*****/
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{
    RCANET0.TXACK0.WORD = 0x0002; /* 送信完了フラグクリア(クリア条件:1ライト) */
    while(RCANET0.TXACK0.WORD & 0x0002); /* フラグのチェック */
}
/*****/

```

### (2) 受信側プログラムリスト

#### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_IO_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMエリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    sleep();                 /* ソフトウェアスタンバイモードに遷移 */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */
}

```

```

/*****
/*      H8SX/1544 I/O Initialize routine      */
/*****
void set_1544_IO_init(void){

    RCANET0.RCANMON.BYTE = 0x20;          /* RCAN-ET 送受信端子レベル      */
    P6.ICR.BIT.B4 = 1;                  /* P64 を CRx_0(入力端子)に設定  */

}

/*****
/*      H8SX/1544 INTC Initialize routine      */
/*****
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;             /* 割り込み制御モード 2          */
    INTC.IPRD.BIT._IRQ12 = 7;           /* 割り込み優先レベル設定：IRQ12 */
    INTC.IPRQ.BIT._RCAN01 = 6;          /* 割り込み優先レベル設定：RCAN  */
    INTC.SSIER.BIT.SSI12 = 1;           /* ソフトウェアスタック解除 IRQ 設定：IRQ12 */
    INTC.IER.BIT.IRQ12E = 1;            /* IRQ 割り込み許可：IRQ12       */
    set_imask_exr(0);                   /* 割り込みマスクレベル 0        */

}
/*****

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
extern void set_1544_IO_init(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/リト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え: MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFFFFD;          /* データレム受信割り込み許可      */
    RCANET0.MBIMR0.WORD &= 0xFFFFE;      /* MB0 割り込み許可                */

    RCANET0.MB[0].CTRL0.WORD.H = 0x1554; /* ID:H'555,スタンダートフォーマット,データレム */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000; /* STD_LAFM, IDE_LAFM の設定        */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;  /* メールボックス0を受信用に設定    */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1                            */

}

/*****
/*      RCAN start routine                  */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE;          /* MCR0 クリア                        */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?                            */

}

/*****
/*      RAM area Initialize routine         */
/*****
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;                    /* データ格納用RAMエリア初期化      */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;

}
    
```

```

/*****
/*      IRQ Interrupt routine                                     */
/*****
#pragma interrupt(IRQ12)
void IRQ12(void){

    INTC.IER.BIT.IRQ12E = 0;          /*  IRQ 割り込み禁止：IRQ12      */

    set_1544_IO_init();              /*  H8SX/1544 I/O 初期化        */

    set_RCAN0_init();               /*  RCAN 初期化                  */
    set_RCANOMB_init();
    set_RCAN0_start();

}

/*****
/*      Data Frame Received Interrupt routine(MailBox0)         */
/*****
#pragma interrupt(RM0_0)
void RM0_0(void)
{

    RCANET0.RXPR0.WORD = 0x0001;     /*  受信完了フラグクリア(クリア条件：1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0001); /*  フラグのチェック              */

    MBbuff.DATA.BYTE[0] = RCANET0.MB[0].MSG_DATA[0]; /*  受信データをRAMに格納          */

}
/*****

```

2.10 DMAC を使用した受信メッセージの転送

2.10.1 仕様

図 2.10.1 に示すように、ノード A からの 15 種類のメッセージを送信し、ノード B で受信します。ノード B は、メールボックス 0 で全メッセージを受信し、DMAC を使用してメールボックス 0 から内蔵 RAM に受信データを転送します。

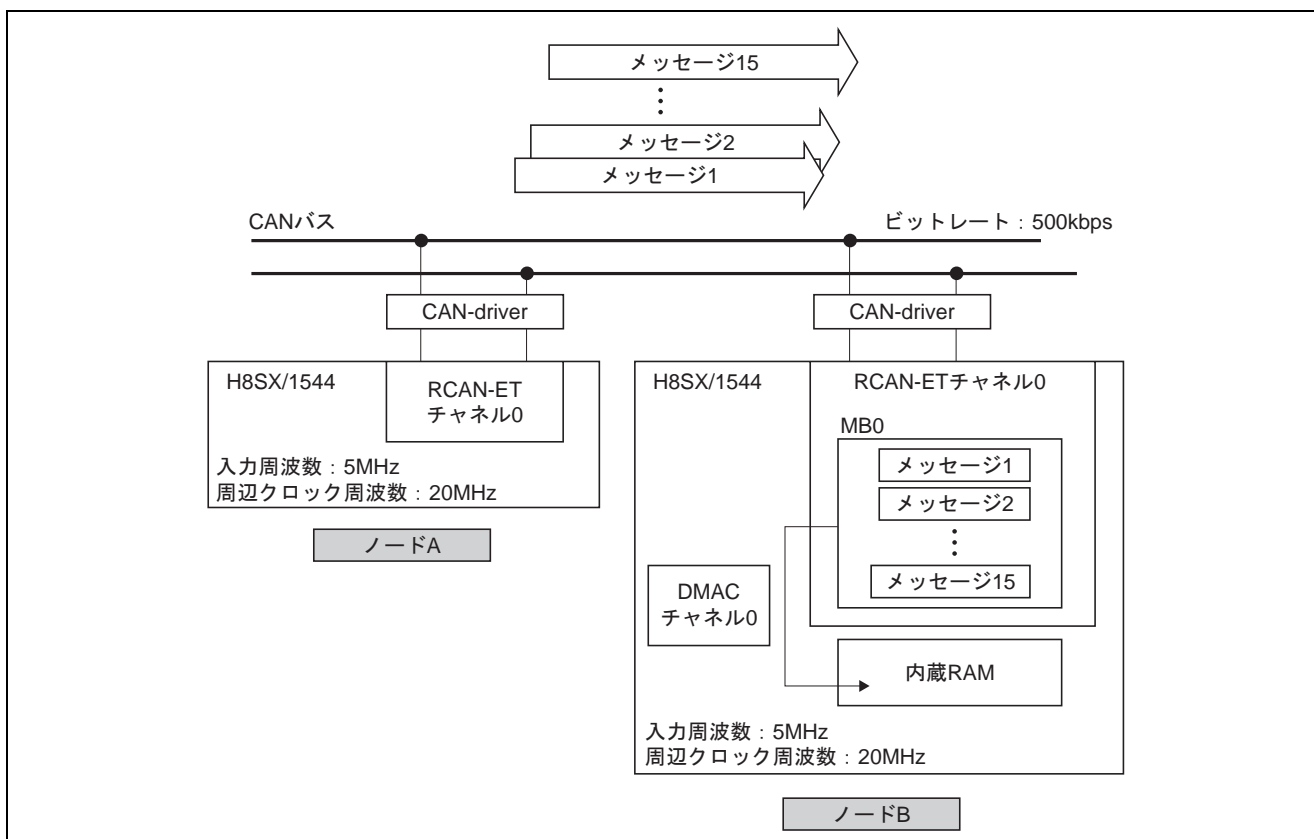


図 2.10.1 通信仕様



表 2.10.1 に、メッセージの仕様を示します。

表 2.10.1 メッセージ仕様

メッセージ	スタンダード ID	データ
メッセージ 1	H'011	H'01 01 01 01 01 01 01 01 (8byte)
メッセージ 2	H'0AA	H'02 02 02 02 02 02 02 02 (8byte)
メッセージ 3	H'122	H'03 03 03 03 03 03 03 03 (8byte)
メッセージ 4	H'1BB	H'04 04 04 04 04 04 04 04 (8byte)
メッセージ 5	H'233	H'05 05 05 05 05 05 05 05 (8byte)
メッセージ 6	H'2CC	H'06 06 06 06 06 06 06 06 (8byte)
メッセージ 7	H'344	H'07 07 07 07 07 07 07 07 (8byte)
メッセージ 8	H'3DD	H'08 08 08 08 08 08 08 08 (8byte)
メッセージ 9	H'455	H'09 09 09 09 09 09 09 09 (8byte)
メッセージ 10	H'4EE	H'0A 0A 0A 0A 0A 0A 0A 0A (8byte)
メッセージ 11	H'566	H'0B 0B 0B 0B 0B 0B 0B 0B (8byte)
メッセージ 12	H'5FF	H'0C 0C 0C 0C 0C 0C 0C 0C (8byte)
メッセージ 13	H'677	H'0D 0D 0D 0D 0D 0D 0D 0D (8byte)
メッセージ 14	H'6EE	H'0E 0E 0E 0E 0E 0E 0E 0E (8byte)
メッセージ 15	H'788	H'0F 0F 0F 0F 0F 0F 0F 0F (8byte)

【注】 本動作例では、スタンダードフォーマットを使用します。(エクステンデッド ID は設定不要)

## 2.10.2 ソフトウェア説明

## (1) モジュール説明

表 2.10.2 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	H8SX/1544 のクロック設定、送受信端子の設定。 RCAN-ET、DMAC のモジュールストップモード解除。	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init	H8SX/1544 の RCAN-ET、DMAC 割り込み優先レベル 設定。	
DMAC 初期設定	set_DMACH0_init	H8SX/1544 の DMAC チャンネル 0 の初期設定。	
DMAC 転送終了 割り込み	DMTEND0	DMA 転送終了割り込みフラグのクリア。データフ レーム受信完了フラグのクリア。次メッセージの転送 先アドレス、転送サイズの設定。	
RCAN 初期設定	set_RCAN0_init	「2.1 初期設定」参照	RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CANメッセージ 送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス エンプティ 割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
受信バッファ 初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	

## (2) 使用定数説明

表 2.10.3 使用定数説明

ラベル名	設定値	機能
ID_1	0x011	CAN メッセージ ID
ID_2	0x0AA	
ID_3	0x122	
ID_4	0x1BB	
ID_5	0x233	
ID_6	0x2CC	
ID_7	0x344	
ID_8	0x3DD	
ID_9	0x455	
ID_10	0x4EE	
ID_11	0x566	
ID_12	0x5FF	
ID_13	0x677	
ID_14	0x6EE	
ID_15	0x788	

## (3) 使用レジスタ説明

## (a) 送信側

表 2.10.4 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFF	メールボックスエンプティ割り込み (IRR8) を許可に設定。 (IMR8=0)
	RCANET0.MBIMR0.WORD	0x0001	メールボックス 1~15 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	(ID_1<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0] ~ RCANET0.MB[1].MSG_DATA[7]	0x01 x 8	送信データを設定。
	RCANET0.MB[2].CTRL0.WORD.H	(ID_2<<2)	メールボックス 2 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[2].CTRL1.BYTE.H	0x00	メールボックス 2 を送信用に設定。
	RCANET0.MB[2].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[2].MSG_DATA[0] ~ RCANET0.MB[2].MSG_DATA[7]	0x02 x 8	送信データを設定。
	RCANET0.MB[3].CTRL0.WORD.H	(ID_3<<2)	メールボックス 3 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[3].CTRL1.BYTE.H	0x00	メールボックス 3 を送信用に設定。
	RCANET0.MB[3].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[3].MSG_DATA[0] ~ RCANET0.MB[3].MSG_DATA[7]	0x03 x 8	送信データを設定。
	RCANET0.MB[4].CTRL0.WORD.H	(ID_4<<2)	メールボックス 4 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[4].CTRL1.BYTE.H	0x00	メールボックス 4 を送信用に設定。
	RCANET0.MB[4].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[4].MSG_DATA[0] ~ RCANET0.MB[4].MSG_DATA[7]	0x04 x 8	送信データを設定。

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[5].CTRL0.WORD.H	(ID_5<<2)	メールボックス 5 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[5].CTRL1.BYTE.H	0x00	メールボックス 5 を送信用に設定。
	RCANET0.MB[5].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[5].MSG_DATA[0] ~ RCANET0.MB[5].MSG_DATA[7]	0x05 x 8	送信データを設定。
	RCANET0.MB[6].CTRL0.WORD.H	(ID_6<<2)	メールボックス 6 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[6].CTRL1.BYTE.H	0x00	メールボックス 6 を送信用に設定。
	RCANET0.MB[6].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[6].MSG_DATA[0] ~ RCANET0.MB[6].MSG_DATA[7]	0x06 x 8	送信データを設定。
	RCANET0.MB[7].CTRL0.WORD.H	(ID_7<<2)	メールボックス 7 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[7].CTRL1.BYTE.H	0x00	メールボックス 7 を送信用に設定。
	RCANET0.MB[7].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[7].MSG_DATA[0] ~ RCANET0.MB[7].MSG_DATA[7]	0x07 x 8	送信データを設定。
	RCANET0.MB[8].CTRL0.WORD.H	(ID_8<<2)	メールボックス 8 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[8].CTRL1.BYTE.H	0x00	メールボックス 8 を送信用に設定。
	RCANET0.MB[8].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[8].MSG_DATA[0] ~ RCANET0.MB[8].MSG_DATA[7]	0x08 x 8	送信データを設定。
	RCANET0.MB[9].CTRL0.WORD.H	(ID_9<<2)	メールボックス 9 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[9].CTRL1.BYTE.H	0x00	メールボックス 9 を送信用に設定。
	RCANET0.MB[9].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[9].MSG_DATA[0] ~ RCANET0.MB[9].MSG_DATA[7]	0x09 x 8	送信データを設定。
RCANET0.MB[10].CTRL0.WORD.H	(ID_10<<2)	メールボックス 10 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。	
RCANET0.MB[10].CTRL1.BYTE.H	0x00	メールボックス 10 を送信用に設定。	

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[10].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。
	RCANET0.MB[10].MSG_DATA[0] ~ RCANET0.MB[10].MSG_DATA[7]	0x0A x 8	送信データを設定。
	RCANET0.MB[11].CTRL0.WORD.H	(ID_11<<2)	メールボックス 11 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[11].CTRL1.BYTE.H	0x00	メールボックス 11 を送信用に設定。
	RCANET0.MB[11].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。
	RCANET0.MB[11].MSG_DATA[0] ~ RCANET0.MB[11].MSG_DATA[7]	0x0B x 8	送信データを設定。
	RCANET0.MB[12].CTRL0.WORD.H	(ID_12<<2)	メールボックス 12 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[12].CTRL1.BYTE.H	0x00	メールボックス 12 を送信用に設定。
	RCANET0.MB[12].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。
	RCANET0.MB[12].MSG_DATA[0] ~ RCANET0.MB[12].MSG_DATA[7]	0x0C x 8	送信データを設定。
	RCANET0.MB[13].CTRL0.WORD.H	(ID_13<<2)	メールボックス 13 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[13].CTRL1.BYTE.H	0x00	メールボックス 13 を送信用に設定。
	RCANET0.MB[13].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。
	RCANET0.MB[13].MSG_DATA[0] ~ RCANET0.MB[13].MSG_DATA[7]	0x0D x 8	送信データを設定。
	RCANET0.MB[14].CTRL0.WORD.H	(ID_14<<2)	メールボックス 14 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[14].CTRL1.BYTE.H	0x00	メールボックス 14 を送信用に設定。
	RCANET0.MB[14].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。
	RCANET0.MB[14].MSG_DATA[0] ~ RCANET0.MB[14].MSG_DATA[7]	0x0E x 8	送信データを設定。
	RCANET0.MB[15].CTRL0.WORD.H	(ID_15<<2)	メールボックス 15 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID を設定。
	RCANET0.MB[15].CTRL1.BYTE.H	0x00	メールボックス 15 を送信用に設定。
RCANET0.MB[15].CTRL1.BYTE.L	0x08	データ長（8byte）を設定。	

モジュール名	レジスタ名	設定値	機能
RCAN メールボックス 初期設定	RCANET0.MB[15].MSG_DATA[0] ~ RCANET0.MB[15].MSG_DATA[7]	0x0F x 8	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 ( TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1 )
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み 設定			
CANメッセージ 送信	RCANET0.TXPR0.LONG	0x0000FFFE	メールボックス 1~15 を送信待ち 状態に設定。
メールボックス エンプティ 割り込み	RCANET0.TXACK0.WORD	-	メールボックス 1~15 の送信完了 フラグのクリア。 (クリア条件:1 ライト)

【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

2. ID\_x (x=11~15)は「2.10.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

(b) 受信側

表 2.10.5 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	SCKCR.BIT.ICK	0	システムクロックの周波数を 40MHz (入力クロック:5MHz x 8) に設定。
	SCKCR.BIT.BCK	1	周辺モジュールクロックの周波数 を 20MHz (入力クロック:5MHz x 4) に設定。
	SCKCR.BIT.PCK	1	外部バスクロックの周波数を 20MHz (入力クロック:5MHz x 4) に設定。
	SBYCR.BIT.SSBY	1	Sleep 命令実行後、ソフトウェアス タンバイモードに遷移するよう に設定。
	MSTP.CRA.BIT._DMAC	0	DMAC のモジュールストップモ ードを解除。
	MSTP.CRC.BIT._RCAN01	0	RCAN-ET のモジュールストップ モードを解除。
	RCANET0.RCANMON.BYTE	0x20	RCAN-ET の送受信端子をイネー ブルに設定。
	P6.ICR.BIT.B4	1	P64 (67Pin) を RCAN-ET チャン ネル 0 の受信端子( CRx_0 )に設定。
DMAC 初期設定	DMAC0.DMDR.LONG	0x000090A0	データサイズをロングワード、転送 モードをブロック転送モードに設 定。また、DMAC の起動要因を、内 蔵モジュール割り込みに設定。

モジュール名	レジスタ名	設定値	機能
DMAC 初期設定	DMAC0.DACR.LONG	0x00220000	アドレスモードをデュアルアドレスモード、ブロックエリアを転送元アドレスに設定。また、アドレス更新モードを、転送元/転送先共にアドレス増加に設定。
	DMAC0.DSAR	注 2	転送元のアドレスを指定。
	DMAC0.DDAR	注 2	転送先のアドレスを指定。
	DMAC0.DOFR	0x00000000	転送元アドレス、転送先アドレスのアドレス更新のオフセット値を設定。(オフセットなしに設定)
	DMAC0.DTCR	0x00000008	転送するデータの総転送サイズ(8byte)を設定。
	DMAC0.DBSR	0x00020000	ブロックサイズ(2ロングワード)を設定。
	DMAC0.DMRSR	0xDC	DMAC 転送の起動要因に使用する内蔵モジュール割り込みのベクタ番号を指定。 <ul style="list-style-type: none"> <li>割り込み要因: RCAN-ET チャンネル 0 のメールボックス 0 のメッセージ受信</li> <li>ベクタ番号: 220(=h'DC)</li> </ul>
	DMAC0.DMDR.BIT.DTIE	1	データ転送終了割り込みを許可に設定。
	DMAC0.DMDR.BIT.DTE	1	データ転送を許可に設定。
RCAN 初期設定	「2.1 初期設定」参照		
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFFFFD	データフレーム受信割り込み(IRR1)を許可に設定。(IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFFE	メールボックス 0 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	0x0000	メールボックス 0 をスタンダードフォーマット、データフレームに設定。
	RCANET0.MB[0].LAFM.WORD.H	0x1FFC	メールボックス 0 のスタンダード ID、IDE ビットのフィルタマスクを設定。(スタンダード ID のみ任意の ID のメッセージを受信可能)
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み 設定	INTC.INTCR.BIT.INTM	2	割り込みコントローラの割り込み制御モードを、割り込み制御モード 2 に設定。

モジュール名	レジスタ名	設定値	機能
H8SX/1544 割り込み 設定	INTC.IPRI.BIT._DMAC0	7	DMAC の割り込み優先レベルを設定。
	INTC.IPRQ.BIT._RCAN01	0	RCAN-ET の割り込み優先レベルを設定。
DMAC 転送終了 割り込み	DMAC0.DMDR.LONG	0xFFFEFFFF	データ転送終了割り込みフラグ (DTIF) クリア。
	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。 (クリア条件: 1 ライト)
	DMAC0.DDAR	注 2	転送先のアドレスを指定。
	DMAC0.DTCR	0x00000008	転送するデータの総転送サイズ (8byte) を設定。
	DMAC0.DBSR	0x00020000	ブロックサイズ (2 ロングワード) を設定。
	DMAC0.DMDR.BIT.DTE	1	データ転送を許可に設定。
	DMAC0.DMDR.BIT.DTIE	0	データ転送終了割り込みを禁止に設定。

- 【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。  
 2. 設定値は、「2.10.4 プログラムリスト (2) 受信側プログラムリスト」を参照してください。

(4) 使用 RAM 説明

表 2.10.6 使用 RAM 説明

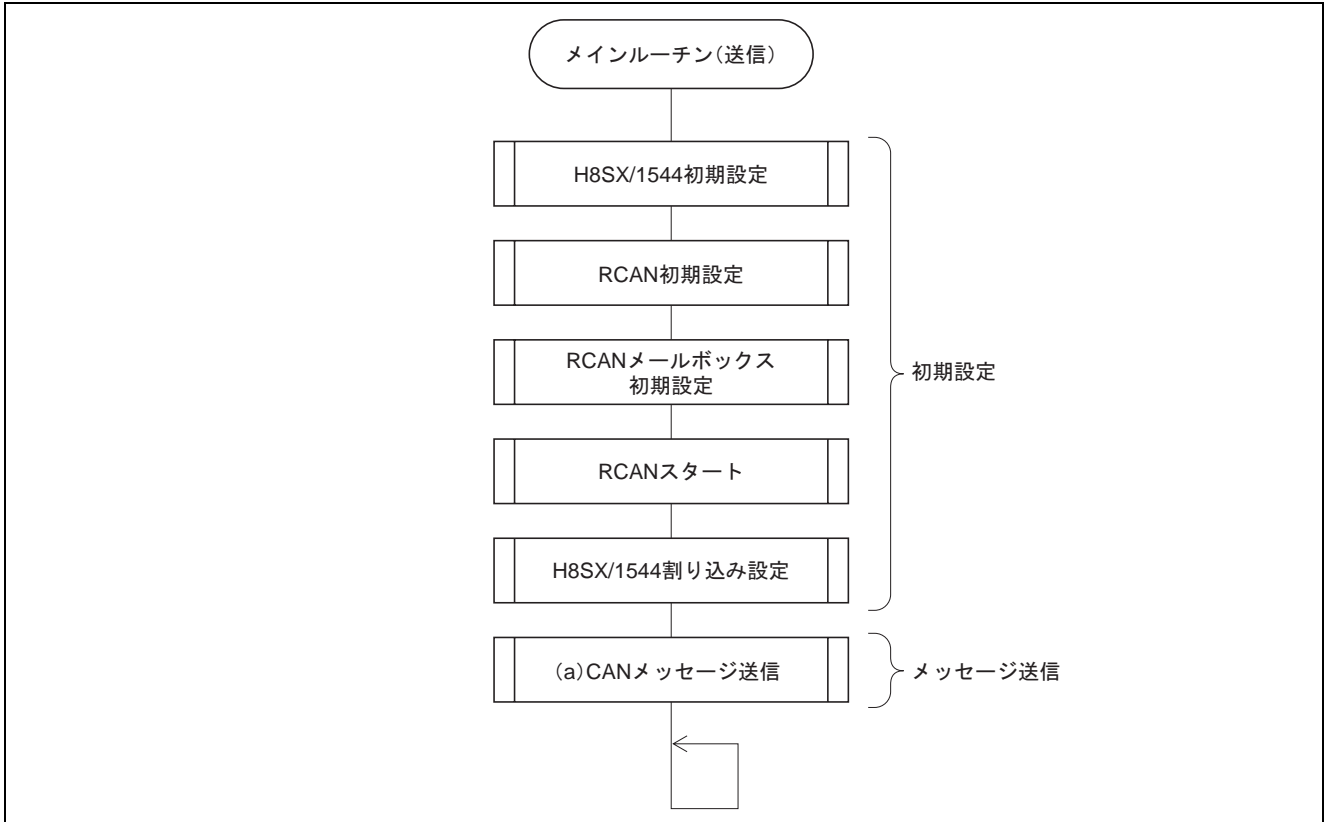
モジュール名	ラベル名	機能
受信バッファ 初期化	Mbbuff[15].ID.LONG MBbuff[15].ID.WORD.H MBbuff[15].ID.WORD.L	受信 ID を格納します。
データフレーム 受信割り込み	MBbuff[15].DATA.LONG[0] ~ [1] MBbuff[15].DATA.WORD[0] ~ [3] MBbuff[15].DATA.BYTE[0] ~ [7]	受信データを格納します。
メインルーチン DMAC 転送終了 割り込み	Rcv_cnt	DMA 転送が終了した受信メッセージ数を示します。



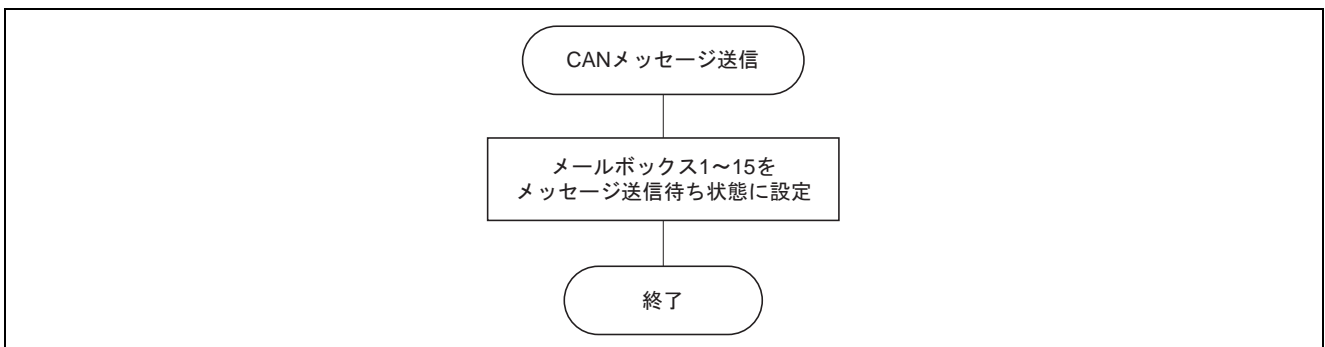
2.10.3 フローチャート

(1) 送信側フローチャート

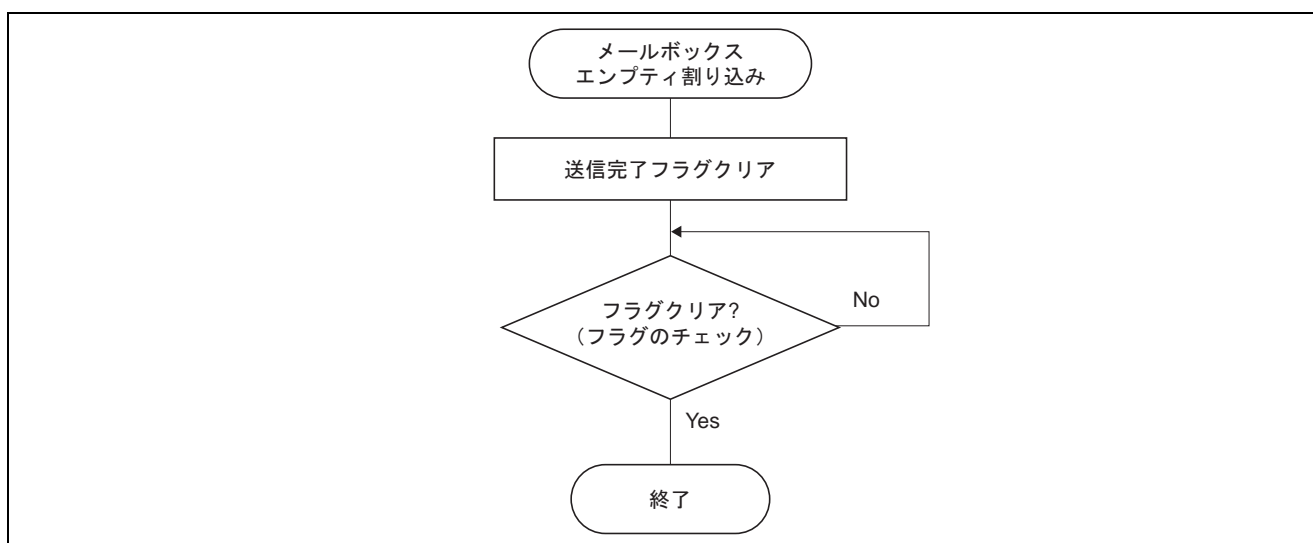
メインルーチン



(a) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



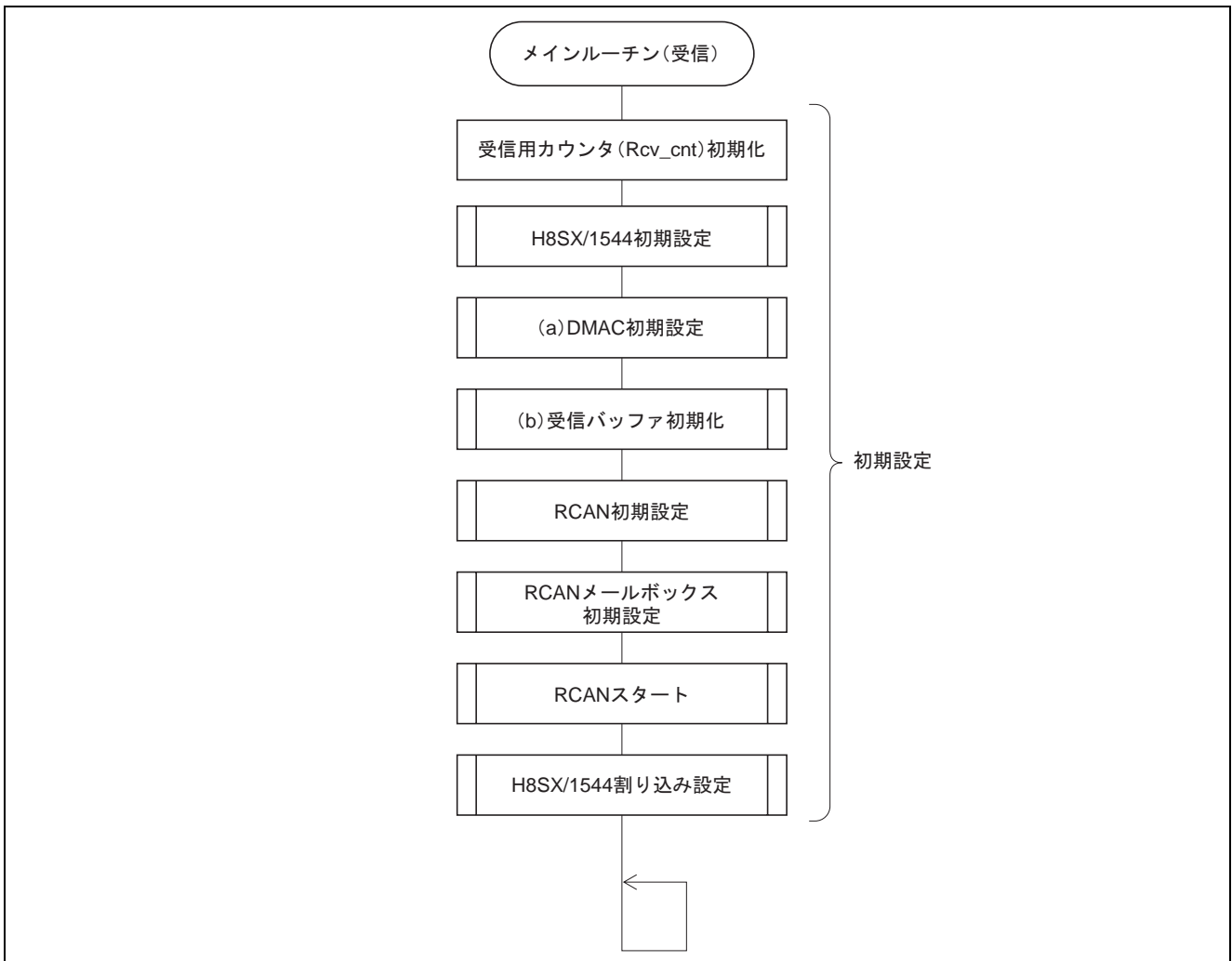
(b) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



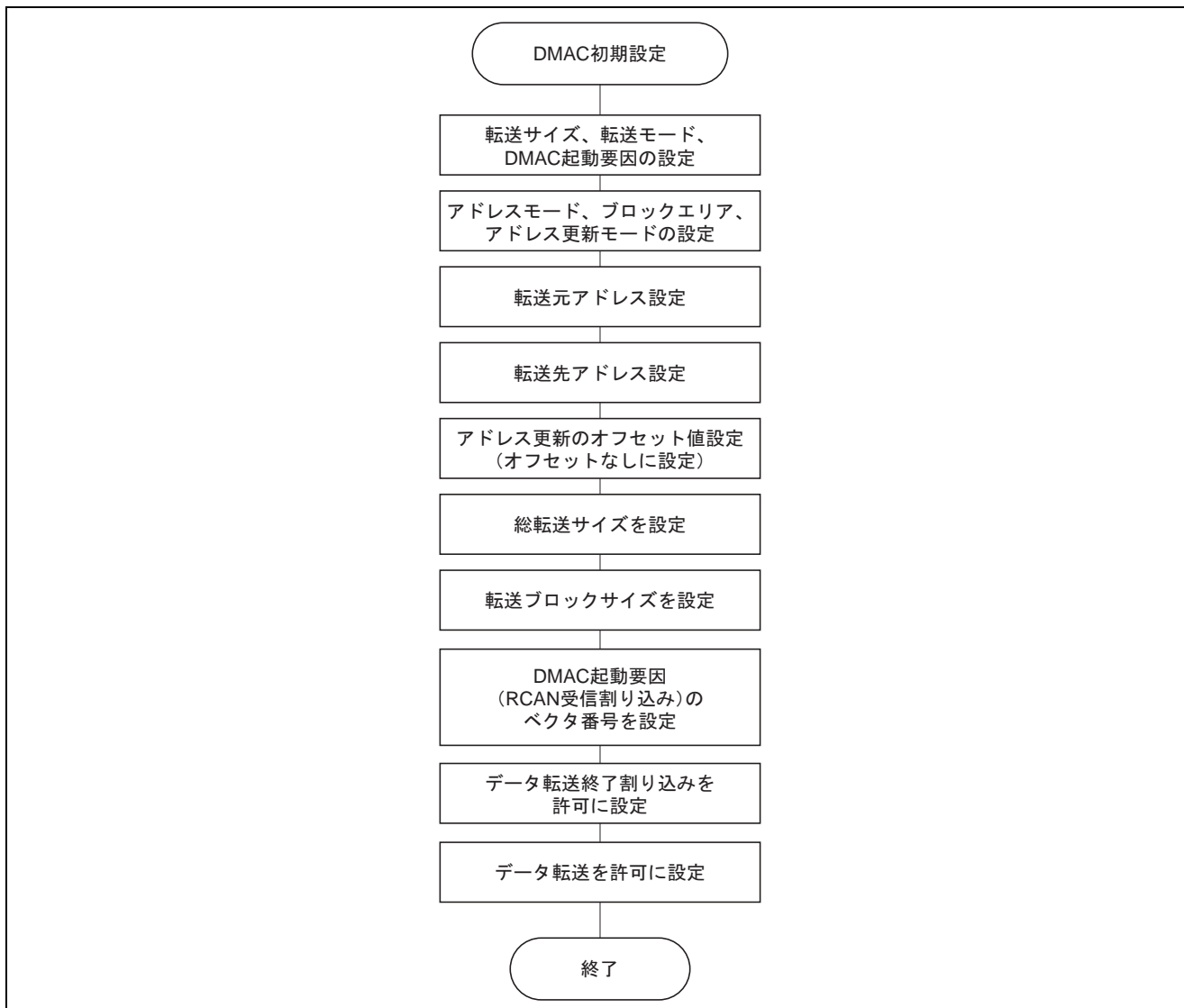
【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(2) 受信側フローチャート

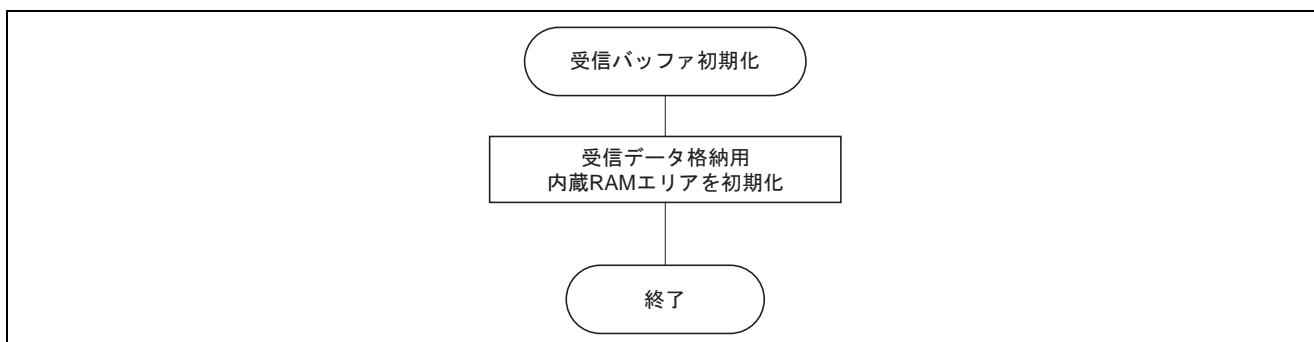
メインルーチン



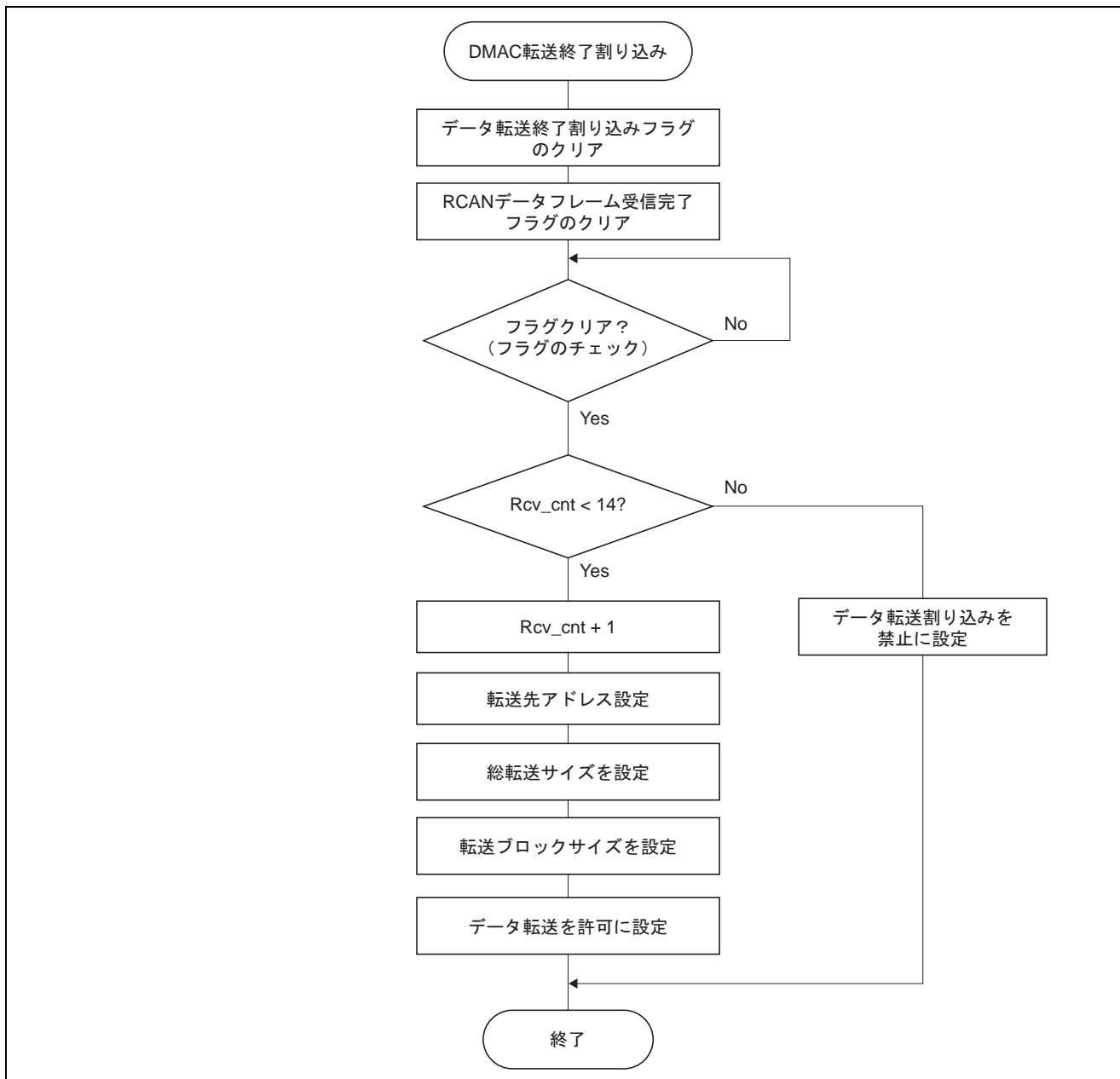
(a) DMAC 初期設定ルーチン (デバイス個別設定)



(b) 受信バッファ初期化ルーチン (デバイス個別設定)



(c) DMAC 転送終了割り込みルーチン (デバイス個別設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

## 2.10.4 プログラムリスト

## (1) 送信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include"iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    set_RCAN0_init();         /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();     /* H8SX/1544 割り込み設定   */
    RCAN0_Tx();              /* CANメッセージ送信         */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;        /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;        /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;        /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;       /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */
}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
*****/
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
/*-----*/
#define ID_1      0x011
#define ID_2      0x0AA
#define ID_3      0x122
#define ID_4      0x1BB
#define ID_5      0x233
#define ID_6      0x2CC
#define ID_7      0x344
#define ID_8      0x3DD
#define ID_9      0x455
#define ID_10     0x4EE
#define ID_11     0x566
#define ID_12     0x5FF
#define ID_13     0x677
#define ID_14     0x6EE
#define ID_15     0x788
/*****
/*      RCAN Initialize routine
*****/
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008); /* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホルト/スリープ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え : MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* マルチアクセス(RAM I/O)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```



```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFF;                /* メールボックス割り込み許可      */
    RCANET0.MBIMR0.WORD = 0x0001;            /* MB1-15 割り込み許可            */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;      /* メールボックス 1 を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x01;   /* 送信データ                        */
    }

    RCANET0.MB[2].CTRL0.WORD.H = (ID_2<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[2].CTRL1.BYTE.H = 0x00;      /* メールボックス 2 を送信用に設定 */
    RCANET0.MB[2].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[2].MSG_DATA[i] = 0x02;   /* 送信データ                        */
    }

    RCANET0.MB[3].CTRL0.WORD.H = (ID_3<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[3].CTRL1.BYTE.H = 0x00;      /* メールボックス 3 を送信用に設定 */
    RCANET0.MB[3].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[3].MSG_DATA[i] = 0x03;   /* 送信データ                        */
    }

    RCANET0.MB[4].CTRL0.WORD.H = (ID_4<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[4].CTRL1.BYTE.H = 0x00;      /* メールボックス 4 を送信用に設定 */
    RCANET0.MB[4].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[4].MSG_DATA[i] = 0x04;   /* 送信データ                        */
    }

    RCANET0.MB[5].CTRL0.WORD.H = (ID_5<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[5].CTRL1.BYTE.H = 0x00;      /* メールボックス 5 を送信用に設定 */
    RCANET0.MB[5].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[5].MSG_DATA[i] = 0x05;   /* 送信データ                        */
    }

    RCANET0.MB[6].CTRL0.WORD.H = (ID_6<<2); /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[6].CTRL1.BYTE.H = 0x00;      /* メールボックス 6 を送信用に設定 */
    RCANET0.MB[6].CTRL1.BYTE.L = 0x08;      /* データ長:8byte                  */
    for(i = 0;i < 8;i++){
        RCANET0.MB[6].MSG_DATA[i] = 0x06;   /* 送信データ                        */
    }
}

```

```

RCANET0.MB[7].CTRL0.WORD.H = (ID_7<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[7].CTRL1.BYTE.H = 0x00; /* メルッス 7 を送信用に設定 */
RCANET0.MB[7].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[7].MSG_DATA[i] = 0x07; /* 送信データ */
}

RCANET0.MB[8].CTRL0.WORD.H = (ID_8<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[8].CTRL1.BYTE.H = 0x00; /* メルッス 8 を送信用に設定 */
RCANET0.MB[8].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[8].MSG_DATA[i] = 0x08; /* 送信データ */
}

RCANET0.MB[9].CTRL0.WORD.H = (ID_9<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[9].CTRL1.BYTE.H = 0x00; /* メルッス 9 を送信用に設定 */
RCANET0.MB[9].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[9].MSG_DATA[i] = 0x09; /* 送信データ */
}

RCANET0.MB[10].CTRL0.WORD.H = (ID_10<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[10].CTRL1.BYTE.H = 0x00; /* メルッス 10 を送信用に設定 */
RCANET0.MB[10].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[10].MSG_DATA[i] = 0x0A; /* 送信データ */
}

RCANET0.MB[11].CTRL0.WORD.H = (ID_11<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[11].CTRL1.BYTE.H = 0x00; /* メルッス 11 を送信用に設定 */
RCANET0.MB[11].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[11].MSG_DATA[i] = 0x0B; /* 送信データ */
}

RCANET0.MB[12].CTRL0.WORD.H = (ID_12<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[12].CTRL1.BYTE.H = 0x00; /* メルッス 12 を送信用に設定 */
RCANET0.MB[12].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[12].MSG_DATA[i] = 0x0C; /* 送信データ */
}

RCANET0.MB[13].CTRL0.WORD.H = (ID_13<<2); /* STDID 設定,スタンダードフォーマット,データレム */
RCANET0.MB[13].CTRL1.BYTE.H = 0x00; /* メルッス 13 を送信用に設定 */
RCANET0.MB[13].CTRL1.BYTE.L = 0x08; /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[13].MSG_DATA[i] = 0x0D; /* 送信データ */
}
    
```

```

RCANET0.MB[14].CTRL0.WORD.H = (ID_14<<2); /* STDID 設定,スタンダードフォーマット,データレーム*/
RCANET0.MB[14].CTRL1.BYTE.H = 0x00;          /* メッセージ 14 を送信用に設定 */
RCANET0.MB[14].CTRL1.BYTE.L = 0x08;          /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[14].MSG_DATA[i] = 0x0E;        /* 送信データ */
}

RCANET0.MB[15].CTRL0.WORD.H = (ID_15<<2); /* STDID 設定,スタンダードフォーマット,データレーム*/
RCANET0.MB[15].CTRL1.BYTE.H = 0x00;          /* メッセージ 15 を送信用に設定 */
RCANET0.MB[15].CTRL1.BYTE.L = 0x08;          /* データ長:8byte */
for(i = 0;i < 8;i++){
    RCANET0.MB[15].MSG_DATA[i] = 0x0F;        /* 送信データ */
}

/*ビットレート=500kbps*/
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
RCANET0.BCR0.WORD = 0x0001;          /* BRP=1 */

}

/*****
/*      RCAN start routine */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFFE;        /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RCAN send message routine */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x0000FFFE;        /* MB1-15 を送信待ち状態に設定 */

}

/*****
/*      Mailbox Empty Interrupt routine */
/*****
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{
    RCANET0.TXACK0.WORD = RCANET0.TXACK0.WORD; /* 送信完了フラグ クリア(クリア条件:1 ライト) */
    while(RCANET0.TXACK0.WORD); /* フラグのチェック */

}
/*****

```

## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET              */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
extern void set_DMACH0_init(void);
/*-----*/
unsigned char Rcv_cnt;
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    Rcv_cnt = 0;                                     /* 受信カウンタ初期化          */
    set_1544_init();                                 /* H8SX/1544 初期化            */
    set_DMACH0_init();                               /* DMAC 初期化                 */
    Clear_MBbuff();                                 /* データ格納用 RAM クリア初期化 */
    set_RCAN0_init();                               /* RCAN 初期化                 */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();                           /* H8SX/1544 割り込み設定      */
    while(1);
}
/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;                               /* システムクロック設定 (5MHz x 8=40MHz) */
    SCKCR.BIT.PCK = 1;                               /* 周辺クロック (5MHz x 4=20MHz)         */
    SCKCR.BIT.BCK = 1;                               /* 外部バスクロック (5MHz x 4=20MHz)     */
    SBYCR.BIT.SSBY = 1;                              /* ソフトウェアスタンバイモード         */
    MSTP.CRA.BIT._DMAC = 0;                          /* モジュールストップモード解除: DMAC   */
    MSTP.CRC.BIT._RCAN01 = 0;                        /* モジュールストップモード解除: RCAN   */
    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20;                    /* RCAN-ET 送受信端子レベル           */
    P6.ICR.BIT.B4 = 1;                              /* P64 を CRx_0 (入力端子) に設定       */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;                /* 割り込み制御モード 2                */
    INTC.IPRI.BIT._DMAC0 = 7;              /* 割り込み優先レベル: DMAC            */
    INTC.IPRQ.BIT._RCAN01 = 0;            /* 割り込み優先レベル: RCAN(割り込み禁止) */
    set_imask_exr(0);                     /* 割り込みマスクレベル 0              */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void Clear_MBbuff(void);
void set_DMACH0_init(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff[15];
/*-----*/
extern unsigned char Rcv_cnt;
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルベックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine      */
/*****
void set_RCAN0MB_init(void){

    RCANET0.IMR.WORD &= 0xFFFD;          /* データレム受信割り込み許可      */
    RCANET0.MBIMR0.WORD &= 0xFFFE;      /* MB0 割り込み許可                */

    RCANET0.MB[0].CTRL0.WORD.H = 0x0000; /* STDID 設定,スタンダードフォーマット,データレム */
    RCANET0.MB[0].LAFM.WORD.H = 0x1FFC; /* STD_LAFM の設定                  */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;  /* メールボックス 0 を受信用に設定 */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1                            */

}

/*****
/*      RCAN start routine      */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0 クリア                      */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0?                          */

}

/*****
/*      RAM area Initialize routine      */
/*****
void Clear_MBbuff(void){

    unsigned int i;

    for(i = 0;i < 15;i++){              /* データ格納用 RAM 初期化          */
        MBbuff[i].ID.LONG = 0;
        MBbuff[i].DATA.LONG[0] = 0;
        MBbuff[i].DATA.LONG[1] = 0;
    }

}

```

```

/*****
/*      DMAC Initialize routine
/*****
void set_DMACH0_init(void){

    DMACH0.DMDR.LONG = 0x000090A0; /* 転送サイズ:ロングワード,ブロック転送,起動要因:内蔵レジスタ */
    DMACH0.DACR.LONG = 0x00220000; /* データアドレス,ブロックアドレス:転送元,転送元/転送先アドレス増加 */
    DMACH0.DSAR = (void*)&RCANET0.MB[0].MSG_DATA[0]; /* 転送元アドレス */
    DMACH0.DDAR = (void*)&MBbuff[Rcv_cnt].DATA.BYTE[0]; /* 転送先アドレス */
    DMACH0.DOFR = 0x00000000; /* アドレスオフセットなし */
    DMACH0.DTCR = 0x00000008; /* 総転送サイズ:8byte */
    DMACH0.DBSR = 0x00020000; /* ブロックサイズ(2ロングワード) */
    DMACH0.DMRSR = 0xDC; /* 割り込み要因:RM0_0(RCANチャネル0のMB0受信割り込み) */
    DMACH0.DMDR.BIT.DTIE = 1; /* データ転送終了割り込み許可 */
    DMACH0.DMDR.BIT.DTE = 1; /* データ転送許可 */

}

/*****
/*      DMAC Transmit End Interrupt routine
/*****
#pragma interrupt(DMTEND0)
void DMTEND0(void)
{

    DMACH0.DMDR.LONG &= 0xFFFFFFF; /* データ転送終了割り込みフラグ(DTIF)クリア */
    RCANET0.RXPR0.WORD = 0x0001; /* RCANメッセージ受信完了フラグクリア */
    while(RCANET0.RXPR0.WORD & 0x0001); /* フラグのチェック */

    if(Rcv_cnt < 14){ /* 全メッセージ(15メッセージ)転送未完了 */
        Rcv_cnt++;
        DMACH0.DDAR = (void*)&MBbuff[Rcv_cnt].DATA.BYTE[0]; /* 転送先アドレス */
        DMACH0.DTCR = 0x00000008; /* 総転送サイズ:8byte */
        DMACH0.DBSR = 0x00020000; /* ブロックサイズ */
        DMACH0.DMDR.BIT.DTE = 1; /* データ転送許可 */
    }
    else{ /* 全メッセージ(15メッセージ)転送完了 */
        DMACH0.DMDR.BIT.DTIE = 0; /* 転送終了割り込み禁止 */
    }

}
/*****

```



## 2.11 メールボックスの再設定

### 2.11.1 仕様

図 2.11.1 に示すように、ノード A のメールボックス 1 からスタンダード ID:H'1BB のメッセージを送信し、ノード B のメールボックス 1 で受信します。

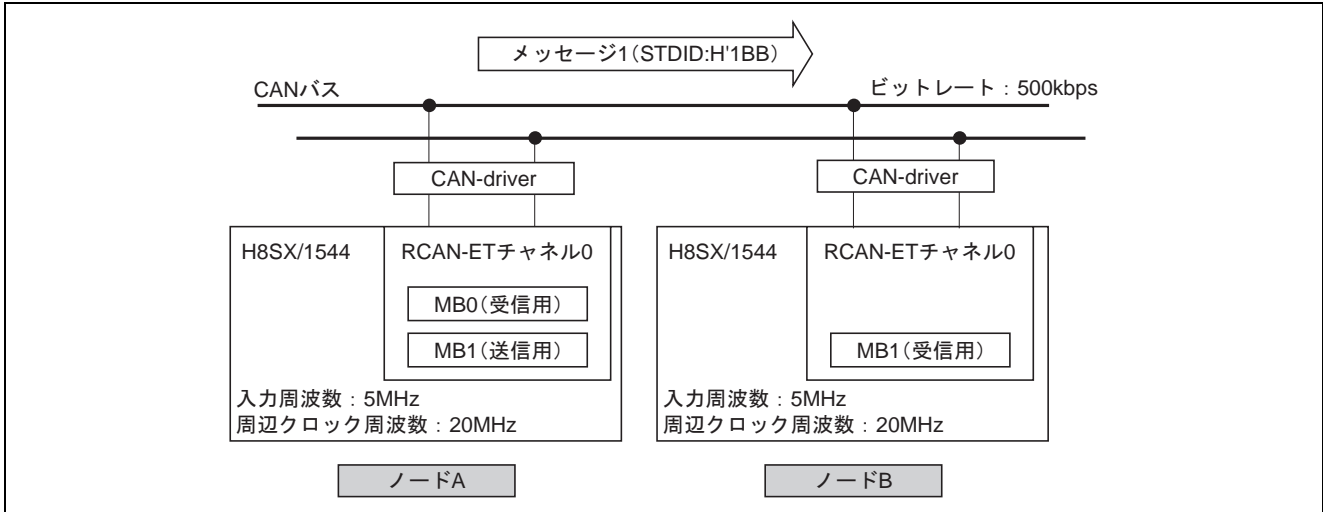


図 2.11.1 通信仕様 (1)

次に、図 2.11.2 に示すように、ノード B はメールボックス 1 を受信用から送信用に変更し、スタンダード ID:H'0AA を再設定してメッセージを送信します。メッセージはノード A のメールボックス 0 で受信します。

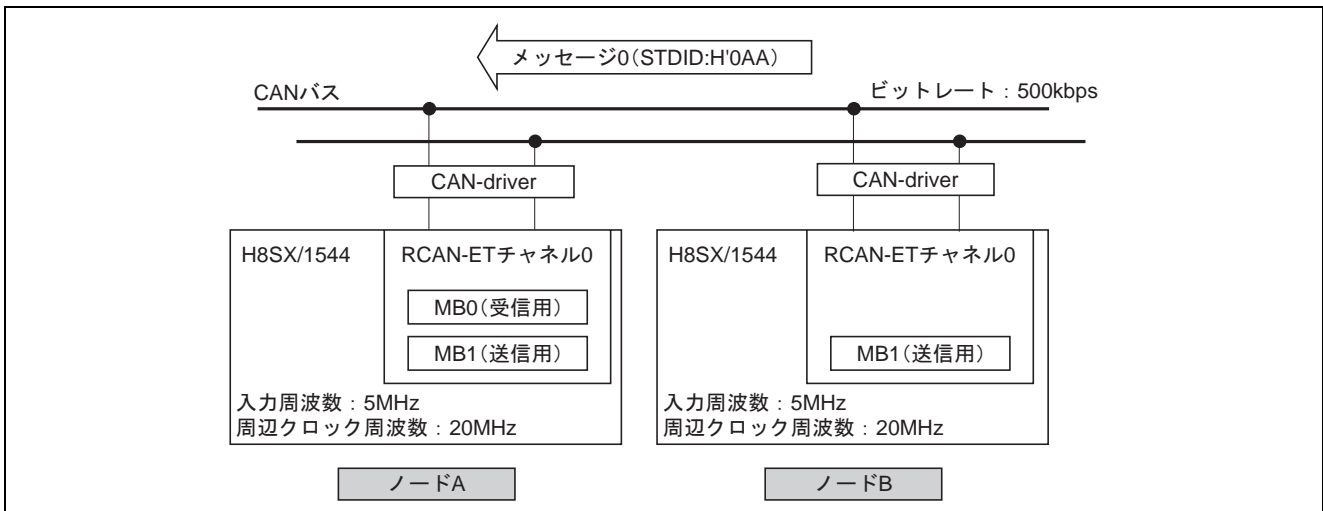


図 2.11.2 通信仕様 (2)

## 2.11.2 ソフトウェア説明

## (1) モジュール説明

表 2.11.1 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
H8SX/1544 割り込み設定	set_1544_INTC_init		
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
CAN メッセージ 送信	RCAN0_Tx	CAN メッセージの送信。	
メールボックス 再設定	set_RCAN0MB_change	メールボックスを受信用から送信用に再設定。	
メールボックス エンプティ 割り込み	SLE0_0	CAN メッセージの送信完了フラグクリア。	
データフレーム 受信割り込み (メールボックス 0)	RM0_0	データフレーム受信完了フラグのクリア。 受信データを RAM に格納。	
データフレーム 受信割り込み* (メールボックス 1)	RM1_0	データフレーム受信完了フラグのクリア。 メールボックスの再設定とメッセージ送信。 メッセージ送信完了フラグのクリア。	
受信バッファ 初期化	Clear_MBbuff	CAN メッセージ格納用内蔵 RAM エリアの初期化。	-

【注】 \* メールボックス 0 以外のメールボックスは、すべての割り込み要因 (RM1,SLE,OVR,ERS) に対して、共通の割り込みベクタアドレスを使用します。したがって、本動作例では、データフレーム受信割り込み (IRR1) とデータフレーム送信完了割り込み (IRR8) を同じ割り込み関数で処理します。

## (2) 使用定数説明

表 2.11.2 使用定数説明

ラベル名	設定値	機能
ID_0	0x0AA	CAN メッセージ ID
ID_1	0x1BB	

## (3) 使用レジスタ説明

## (a) 送信側

表 2.11.3 使用レジスタ説明 (送信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFD	メールボックスエンプティ割り込み (IRR8) およびデータフレーム受信割り込み (IRR1) を許可に設定。(IMR8=0,IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFC	メールボックス 0,1 を割り込み許可に設定。
	RCANET0.MB[0].CTRL0.WORD.H	(ID_0<<2)	メールボックス 0 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID および IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.MB[1].CTRL0.WORD.H	(ID_1<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0] ~ RCANET0.MB[1].MSG_DATA[7]	0x01 × 8	送信データを設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。
メールボックス エンプティ 割り込み	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。 (クリア条件: 1 ライト)
データフレーム 受信割り込み (メールボックス 0)	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。 (クリア条件: 1 ライト)

【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

2. ID\_0, ID\_1 は「2.11.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

## (b) 受信側

表 2.11.4 使用レジスタ説明 (受信側)

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	「2.1 初期設定」参照		
RCAN 初期設定			
RCAN メールボックス 初期設定	RCANET0.IMR.WORD	0xFEFD	メールボックスエンプティ割り込み (IRR8) およびデータフレーム受信割り込み (IRR1) を許可に設定。(IMR8=0,IMR1=0)
	RCANET0.MBIMR0.WORD	0xFFFD	メールボックス 1 を割り込み許可に設定。
	RCANET0.MB[1].CTRL0.WORD.H	(ID_1<<2)	メールボックス 1 をスタンダードフォーマット、データフレームに設定。 また、スタンダード ID を設定。
	RCANET0.MB[1].LAFM.WORD.H	0x0000	メールボックス 1 のスタンダード ID、IDE ビットのフィルタマスクを設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x02	メールボックス 1 を受信用に設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq),TSEG2=3(4tq), SJW=0,BSP=0,BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
H8SX/1544 割り込み設定			
データフレーム 受信割り込み (メールボックス 1)	RCANET0.RXPR0.WORD	0x0002	メールボックス 1 の受信完了フラグのクリア。 (クリア条件: 1 ライト)
	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。 (クリア条件: 1 ライト)
メールボックス 再設定	RCANET0.MCR.WORD	0x0002	RCAN-ET をホルトモードに設定。 (MCR1=1)
	RCANET0.GSR.WORD	-	RCAN-ET がホルトモードに遷移したことを確認。 (GSR4=1 を確認)
	RCANET0.IRR.WORD	-	RCAN-ET がホルトモードに遷移したことを確認。 (IRR0=1 を確認)
	RCANET0.IRR.WORD	0x0001	リセット / ホルト / スリープ割り込みフラグ(IRR0)をクリア。 (クリア条件: 1 ライト)
	RCANET0.MB[1].CTRL0.WORD.H	(ID_0<<2)	メールボックス 1 のスタンダード ID の設定を変更。 (ID_1 ID_0 に変更)

モジュール名	レジスタ名	設定値	機能
メールボックス 再設定	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 の MBC 設定を変更。(受信用 送信用に変更)
	RCANET0.MB[1].CTRL1.BYTE.L	0x08	データ長 (8byte) を設定。
	RCANET0.MB[1].MSG_DATA[0] ~ RCANET0.MB[1].MSG_DATA[7]	0x02 × 8	送信データを設定。
	RCANET0.MCR.WORD	0xFFFFD	RCAN-ET のホルトモードを解除。 (MCR1=0)
	RCANET0.GSR.WORD	-	RCAN-ET のホルトモードが解除されたことを確認。 (GSR4=0を確認)
CAN メッセージ 送信	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。

- 【注】 1. 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。  
 2. ID\_0, ID\_1 は「2.11.2 ソフトウェア説明 (2) 使用定数説明」のラベル名を示しています。

#### (4) 使用 RAM 説明

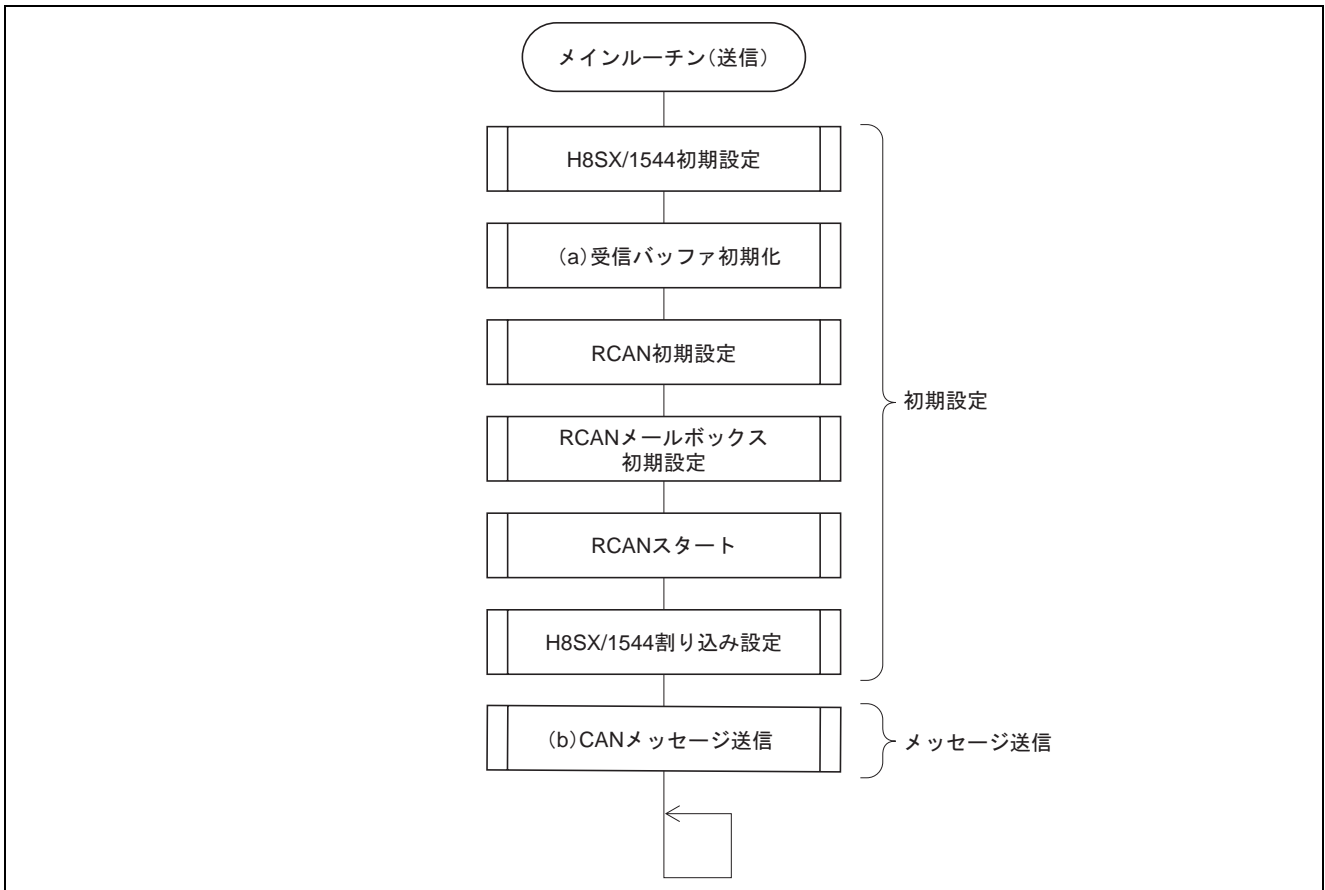
表 2.11.5 使用 RAM 説明

モジュール名	ラベル名	機能
受信バッファ初期化  データフレーム 受信割り込み	Mbbuff.ID.LONG MBbuff.ID.WORD.H MBbuff.ID.WORD.L	受信 ID を格納します。
	MBbuff.DATA.LONG[0] ~ [1] MBbuff.DATA.WORD[0] ~ [3] MBbuff.DATA.BYTE[0] ~ [7]	受信データを格納します。

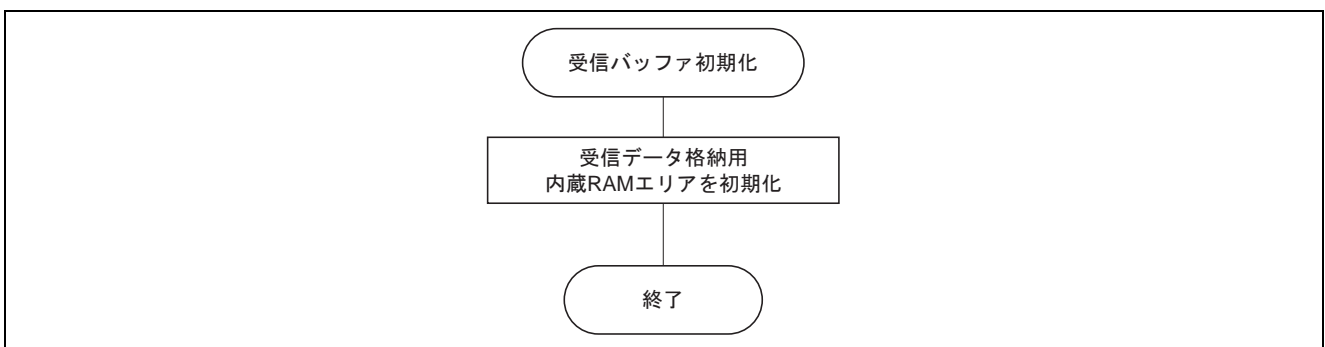
2.11.3 フローチャート

(1) 送信側フローチャート

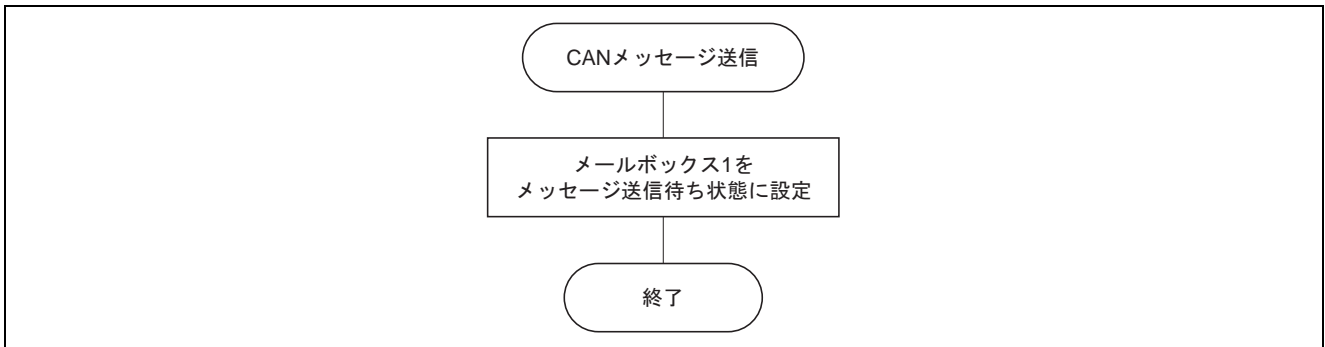
メインルーチン



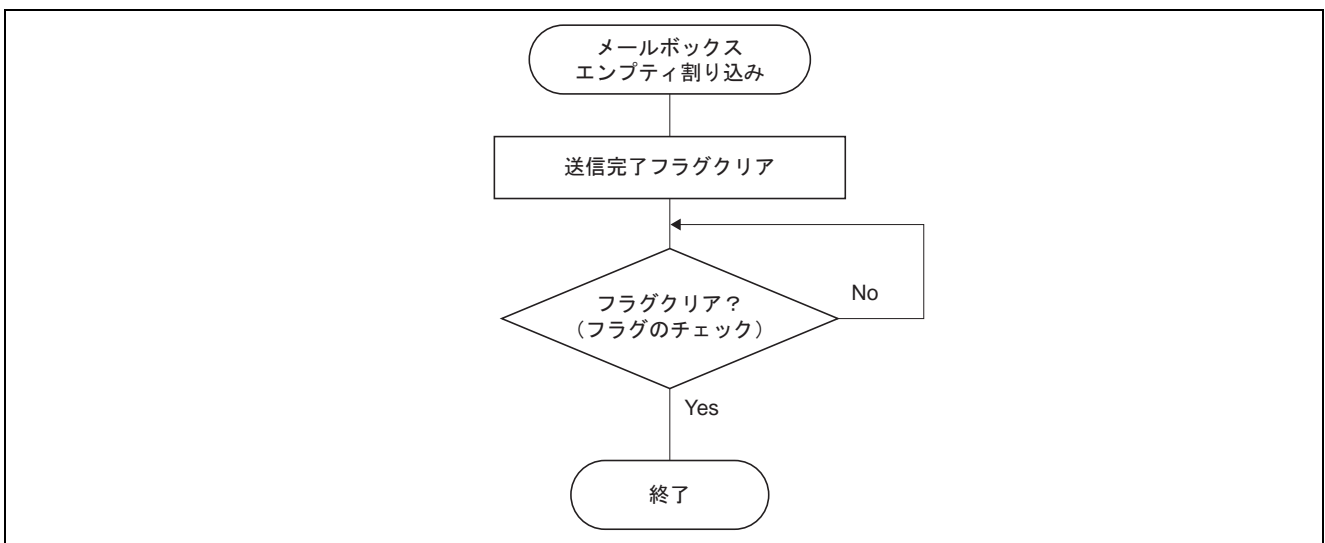
(a) 受信バッファ初期化ルーチン (デバイス個別設定)



(b) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)

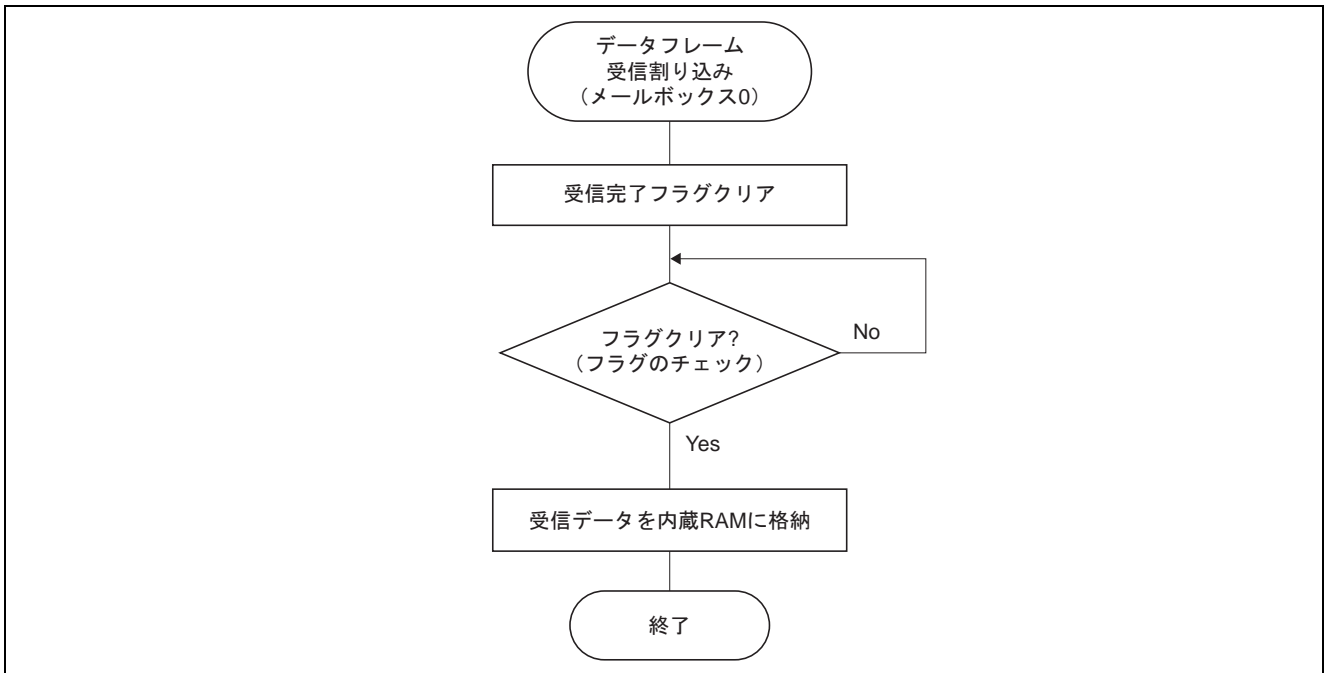


(c) メールボックスエンプティ割り込みルーチン (RCAN-ET 共通設定)



【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(d) データフレーム受信割り込みルーチン ( メールボックス 0, RCAN-ET 共通設定 )

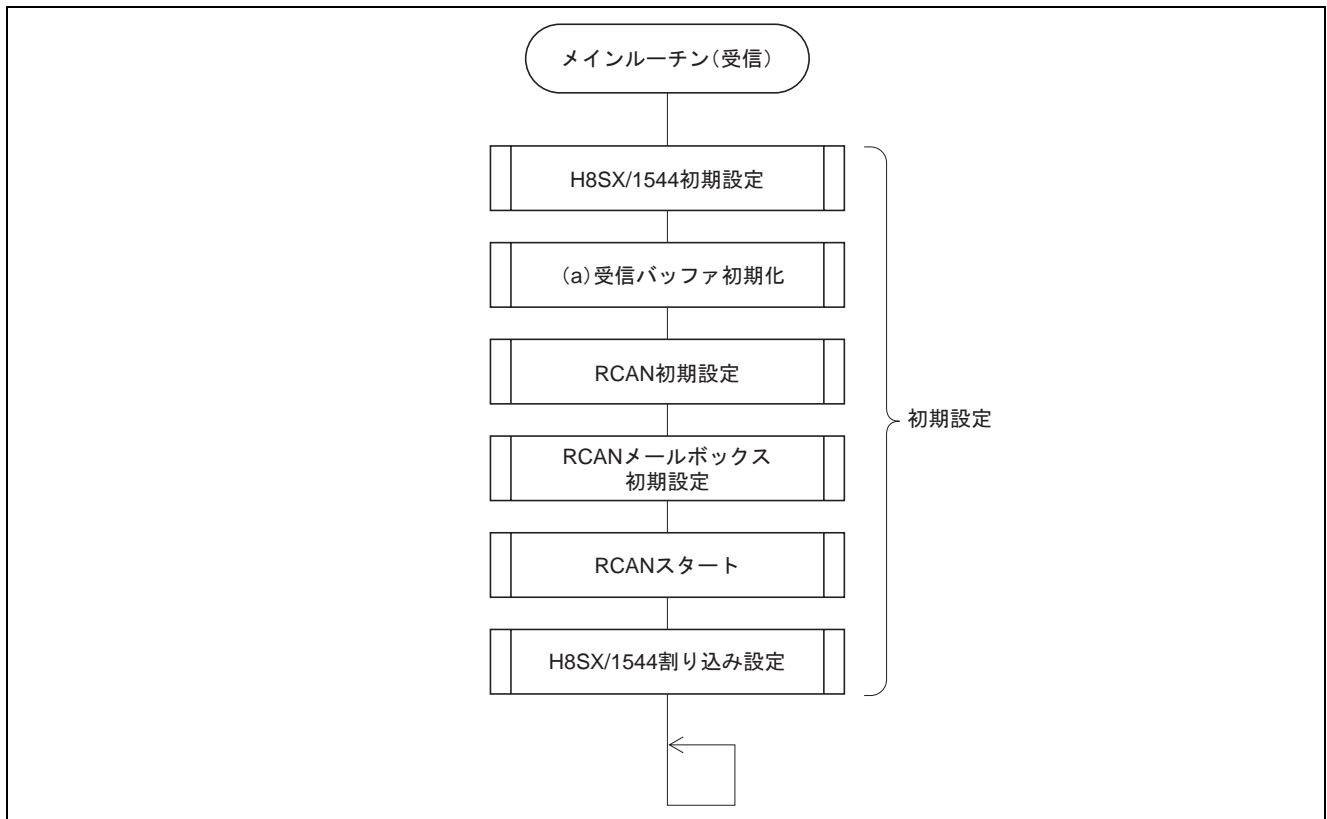


【注】 クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

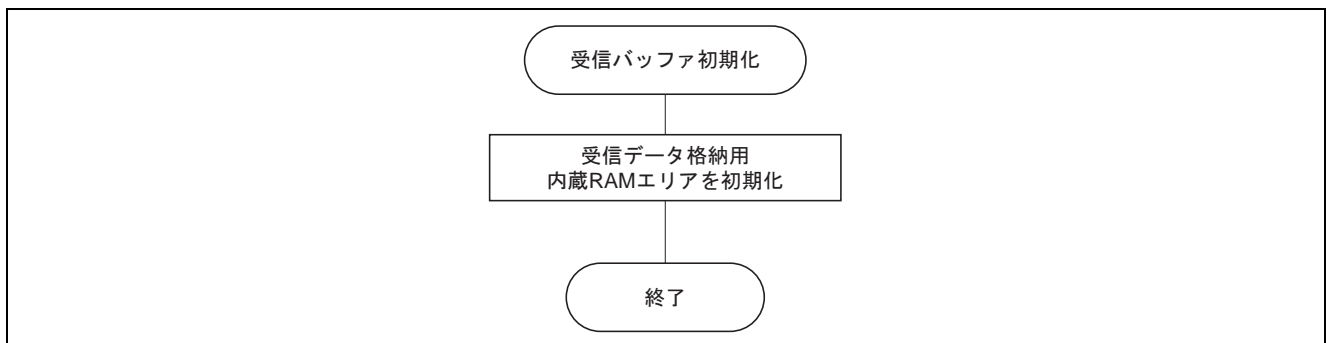


(2) 受信側フローチャート

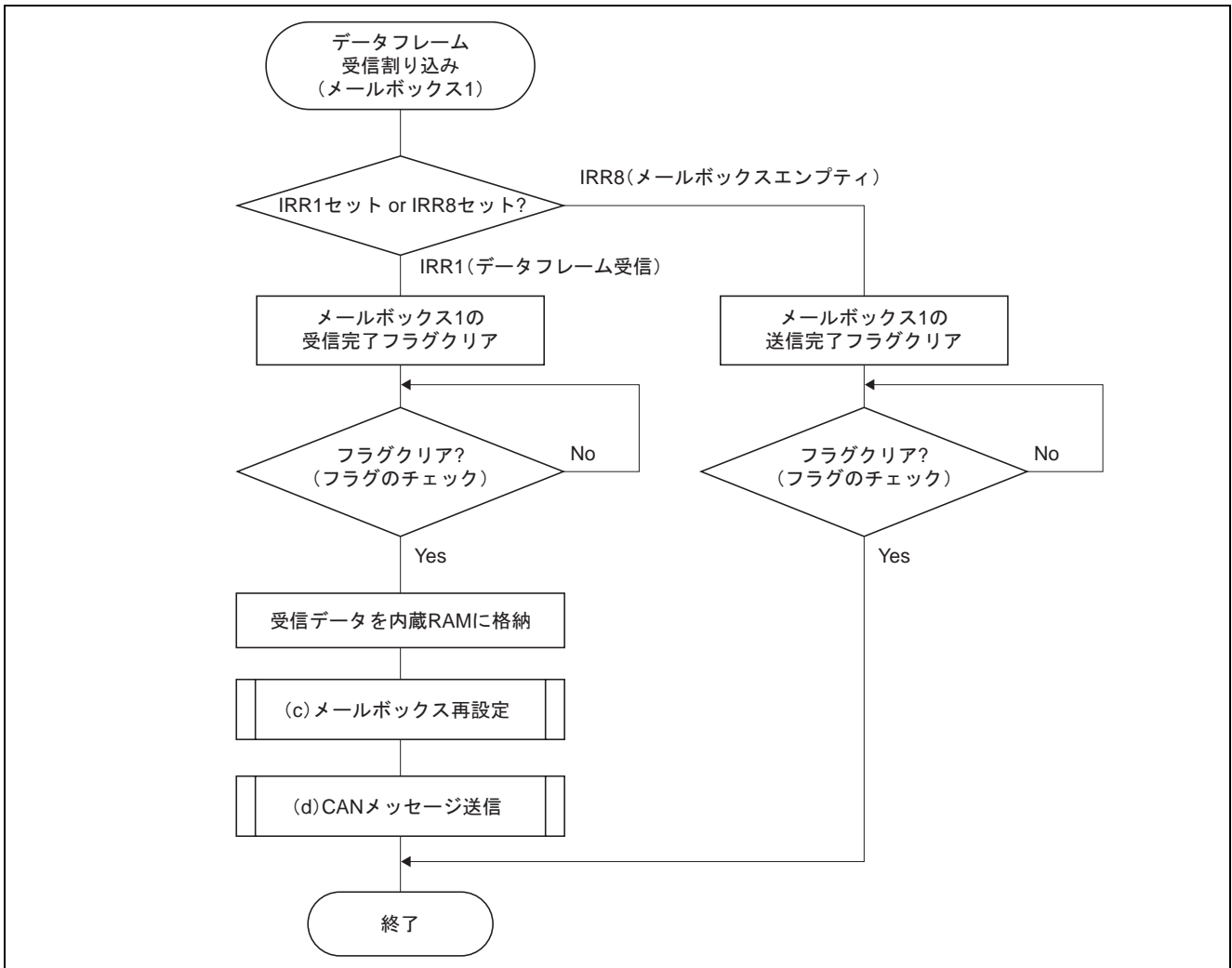
メインルーチン



(a) 受信バッファ初期化ルーチン (デバイス個別設定)

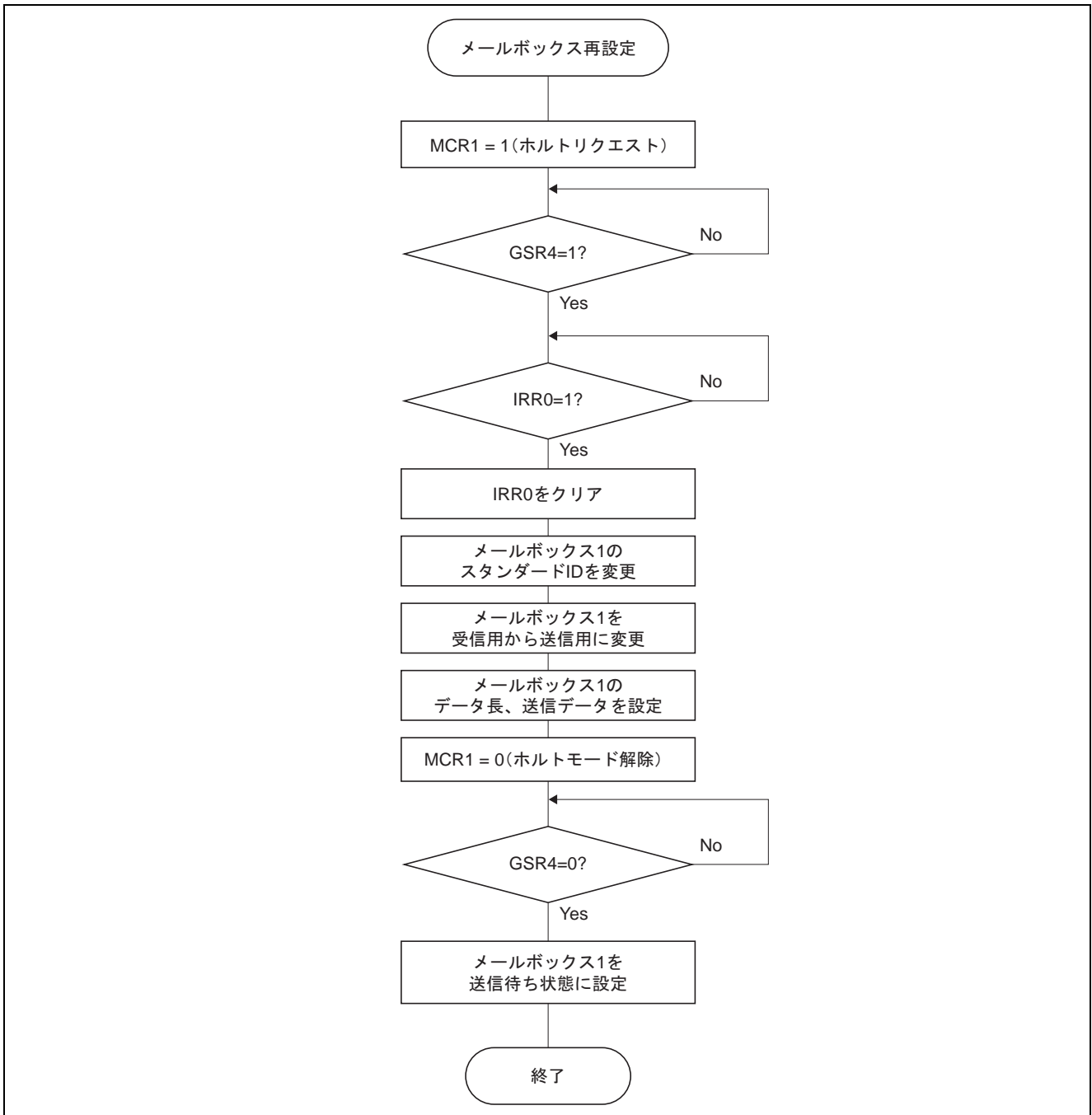


(b) データフレーム受信割り込みルーチン (メールボックス 1, RCAN-ET 共通設定)

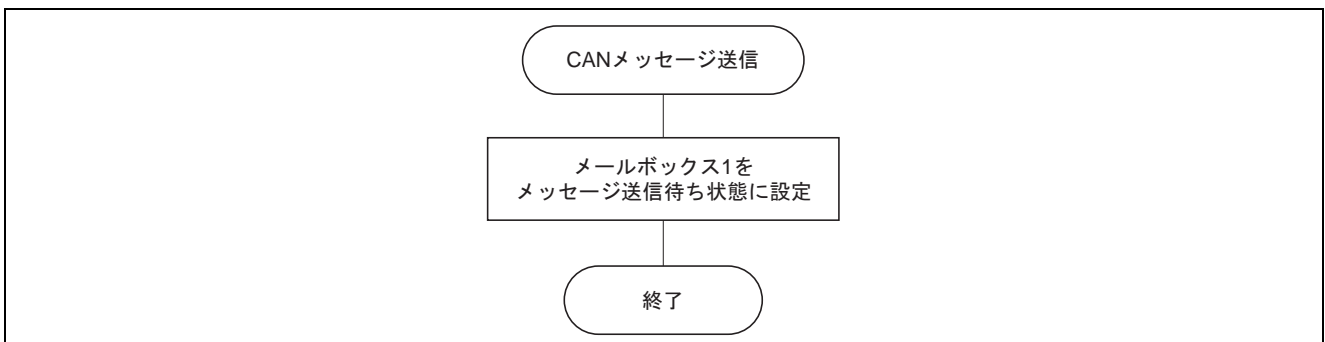


- 【注】
1. メールボックス 0 以外のメールボックスは、すべての割り込み要因 (RM1, SLE, OVR, ERS) に対して、共通の割り込みベクタアドレスを使用します。したがって、本動作例では、データフレーム受信割り込み (IRR1) とデータフレーム送信完了割り込み (IRR8) を、同じ割り込み関数で処理します。
  2. クリアした割り込み要因を誤って再度受け付けないようにするため、割り込み要因フラグをクリア後にフラグをリードし、クリアされたことを確認した後、RTE 命令を実行してください。

(c) メールボックス再設定ルーチン (RCAN-ET 共通設定)



(d) CAN メッセージ送信ルーチン (RCAN-ET 共通設定)



### 2.11.4 プログラムリスト

#### (1) 送信側プログラムリスト

##### (a) main.c

```

/*****
/* Filename      :   main.c                               */
/* Written       :   '06/06/01  REV.1.00                 */
/* Purpose       :   for H8SX/1544  RCAN-ET               */
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void RCAN0_Tx(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program                                     */
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用 RAM 初期化     */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    RCAN0_Tx();             /* CAN メッセージ送信         */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine                     */
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定 (5MHz x 8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック (5MHz x 4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック (5MHz x 4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード         */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除: RCAN   */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル           */
    P6.ICR.BIT.B4 = 1;        /* P64 を CRx_0 (入力端子) に設定       */
}

```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      : rcan.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
/*****
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void RCAN0_Tx(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*-----*/
#define ID_0      0x0AA
#define ID_1      0x1BB
/*****
/*      RCAN Initialize routine
/*****
void set_RCAN0_init(void){
    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え:MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メルブルックス(RAM Iリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }
}

```

```

/*****
/*      RCAN Mailbox Initialize routine                                     */
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFD;          /* メールボックスレジスタ、データフレーム受信割り込み許可 */
    RCANET0.MBIMR0.WORD = 0xFFFC;      /* MB0-1 割り込み許可 */

    RCANET0.MB[0].CTRL0.WORD.H = (ID_0<<2); /* STDID 設定,スタンダードフォーマット,データフレーム */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000;    /* STD_LAFM, IDE_LAFM の設定 */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02;     /* メールボックス 0 を受信用に設定 */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダードフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;     /* メールボックス 1 を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;     /* データ長: 8byte */
    for(i = 0; i < 8; i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x01;  /* 送信データ */
    }

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, (pφ=20MHz) */
    RCANET0.BCR0.WORD = 0x0001; /* BRP=1 */

}

/*****
/*      RCAN start routine                                               */
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

/*****
/*      RCAN send message routine                                       */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002;     /* MB1 を送信待ち状態に設定 */

}

/*****
/*      RAM area Initialize routine                                     */
/*****

```

```

/*****/
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;                /* データ格納用 RAM 初期化 */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;

}

/*****/
/* Mailbox Empty Interrupt routine */
/*****/
#pragma interrupt(SLE0_0)
void SLE0_0(void)
{
    RCANET0.TXACK0.WORD = 0x0002;      /* 送信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.TXACK0.WORD & 0x0002); /* フラグのチェック */

}

/*****/
/* Data Frame Received Interrupt routine(MailBox0) */
/*****/
#pragma interrupt(RM0_0)
void RM0_0(void)
{
    unsigned int i;

    RCANET0.RXPR0.WORD = 0x0001;      /* 受信完了フラグクリア(クリア条件:1バイト) */
    while(RCANET0.RXPR0.WORD & 0x0001); /* フラグのチェック */

    for(i = 0;i < 8;i++){
        MBbuff.DATA.BYTE[i] = RCANET0.MB[0].MSG_DATA[i]; /* 受信データを RAM に格納 */
    }

}
/*****/

```



## (2) 受信側プログラムリスト

## (a) main.c

```

/*****
/* Filename      :   main.c
/* Written       :   '06/06/01   REV.1.00
/* Purpose       :   for H8SX/1544   RCAN-ET
/*****
#include <machine.h>
#include "iodefine.h"
/*-----*/
void main(void);
void set_1544_init(void);
void set_1544_INTC_init(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void Clear_MBbuff(void);
/*****
/*      Main program
/*****
#pragma entry main(sp=0xFFC000)
void main(void)
{
    set_1544_init();           /* H8SX/1544 初期化           */
    Clear_MBbuff();          /* データ格納用RAMクリア初期化 */
    set_RCAN0_init();        /* RCAN 初期化               */
    set_RCAN0MB_init();
    set_RCAN0_start();
    set_1544_INTC_init();    /* H8SX/1544 割り込み設定     */
    while(1);
}

/*****
/*      H8SX/1544 Initialize routine
/*****
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 0;       /* システムクロック設定(5MHz×8=40MHz) */
    SCKCR.BIT.PCK = 1;       /* 周辺クロック(5MHz×4=20MHz)         */
    SCKCR.BIT.BCK = 1;       /* 外部バスクロック(5MHz×4=20MHz)     */
    SBYCR.BIT.SSBY = 1;      /* ソフトウェアスタンバイモード       */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除:RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル         */
    P6.ICR.BIT.B4 = 1;         /* P64 を CRx_0(入力端子)に設定     */
}
    
```

```

/*****/
/*      H8SX/1544 INTC Initialize routine      */
/*****/
void set_1544_INTC_init(void){

    INTC.INTCR.BIT.INTM = 2;          /* 割り込み制御モード 2          */
    INTC.IPRQ.BIT._RCAN01 = 7;      /* 割り込み優先レベル設定：RCAN */
    set_imask_exr(0);              /* 割り込みマスクレベル 0      */

}
/*****/

```

## (b) rcan.c

```

/*****
/* Filename      :   rcan.c
/* Written       :   '06/06/01  REV.1.00
/* Purpose       :   for H8SX/1544  RCAN-ET
*****/
#include"iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
void set_RCAN0MB_change(void);
void RCAN0_Tx(void);
void Clear_MBbuff(void);
/*-----*/
struct {
    union {
        unsigned long LONG;
        struct{
            unsigned short H;
            unsigned short L;
        }WORD;
    }ID;
    union {
        unsigned char BYTE[8];
        unsigned short WORD[4];
        unsigned long LONG[2];
    }DATA;
} MBbuff;
/*-----*/
#define ID_0      0x0AA
#define ID_1      0x1BB
/*****
/*      RCAN Initialize routine
*****/
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリクエスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件：1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え：MCR15=1(初期値)に設定 */

```

```

for(i = 0;i < 16;i++){
    RCANET0.MB[i].CTRL0.WORD.H = 0;
    RCANET0.MB[i].CTRL0.WORD.L = 0;
    RCANET0.MB[i].LAFM.WORD.H = 0;
    RCANET0.MB[i].LAFM.WORD.L = 0;
    for(j = 0;j < 8;j++){
        RCANET0.MB[i].MSG_DATA[j] = 0;
    }
}

}

/*****
/*      RCAN Mailbox Initialize routine
/*****
void set_RCAN0MB_init(void){

    unsigned int i;

    RCANET0.IMR.WORD &= 0xFEFD;          /* メールボックスインポート、データレム受信割り込み許可 */
    RCANET0.MBIMR0.WORD &= 0xFFFD;      /* MB1 割り込み許可 */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_1<<2); /* STDID 設定,スタンダートフォーマット,データレム*/
    RCANET0.MB[1].LAFM.WORD.H = 0x0000;   /* STD_LAFM,IDE_LAFM の設定 */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x02;    /* メールボックス1を受信用に設定 */

    /*ビットレート=500kbps*/
    RCANET0.BCR1.WORD = 0x4300; /* TSEG1=4(5tq),TSEG2=3(4tq),SJW=0,BSP=0,(pφ=20MHz)*/
    RCANET0.BCR0.WORD = 0x0001;          /* BRP=1 */

}

/*****
/*      RCAN start routine
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE;          /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}

```

```

/*****
/*      RCAN Mailbox Change routine      */
/*****
void set_RCAN0MB_change(void){

    unsigned int i;

    /* メールボックス再設定 */
    RCANET0.MCR.WORD |= 0x0002;          /* MCR1 セット(ホトリクエスト)      */
    while((RCANET0.GSR.WORD & 0x0010) != 0x0010); /* GSR4=1?(ホトリート 遷移)      */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001); /* IRR0=1?(リセット/ホトリート/スリーブ 割り込み) */
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */

    RCANET0.MB[1].CTRL0.WORD.H = (ID_0<<2); /* STDID 設定変更      */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00;      /* MBC 設定変更(送信用に設定) */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x08;      /* データ長:8byte      */
    for(i = 0;i < 8;i++){
        RCANET0.MB[1].MSG_DATA[i] = 0x02; /* 送信データ      */
    }

    RCANET0.MCR.WORD &= 0xFFFD;          /* MCR1 クリア(ホトリート 解除要求) */
    while ((RCANET0.GSR.WORD & 0x0010) != 0x0000); /* GSR4=0?(ホトリート 解除)      */

}

/*****
/*      RCAN send message routine      */
/*****
void RCAN0_Tx(void){

    RCANET0.TXPR0.LONG = 0x00000002;      /* MB1 を送信待ち状態に設定      */

}

/*****
/*      RAM area Initialize routine      */
/*****
void Clear_MBbuff(void){

    MBbuff.ID.LONG = 0;                    /* データ格納用 RAM 初期化      */
    MBbuff.DATA.LONG[0] = 0;
    MBbuff.DATA.LONG[1] = 0;

}

```

```

/*****
/*      Data Frame Received Interrupt routine(MailBox1-15)      */
/*****
#pragma interrupt(RM1_0)
void RM1_0(void)
{
    unsigned int i;

    if(RCANET0.RXPR0.WORD){
        RCANET0.RXPR0.WORD = 0x0002;
        while(RCANET0.RXPR0.WORD & 0x0002);

        MBbuff.ID.WORD.H = RCANET0.MB[1].CTRL0.WORD.H;
        for(i = 0;i < 8;i++){
            MBbuff.DATA.BYTE[i] = RCANET0.MB[1].MSG_DATA[i];
        }

        set_RCAN0MB_change();

        RCAN0_Tx();
    }
    else if(RCANET0.TXACK0.WORD){
        RCANET0.TXACK0.WORD = 0x0002;
        while(RCANET0.TXACK0.WORD & 0x0002);
    }
}
/*****

```

### 2.12 CAN 経由 FLASH 書き込み

#### 2.12.1 仕様

図 2.12.1 に示すように、マスタから CAN 経由で転送されたデータをスレーブ側 (H8SX/1544) で受信し、スレーブ側の内蔵フラッシュメモリの全面書き換え (512Kbyte) を行います。フラッシュメモリの書き換えには、ユーザープログラムモードを使用します。

【注】 本動作例では、スレーブ側の動作について説明します。

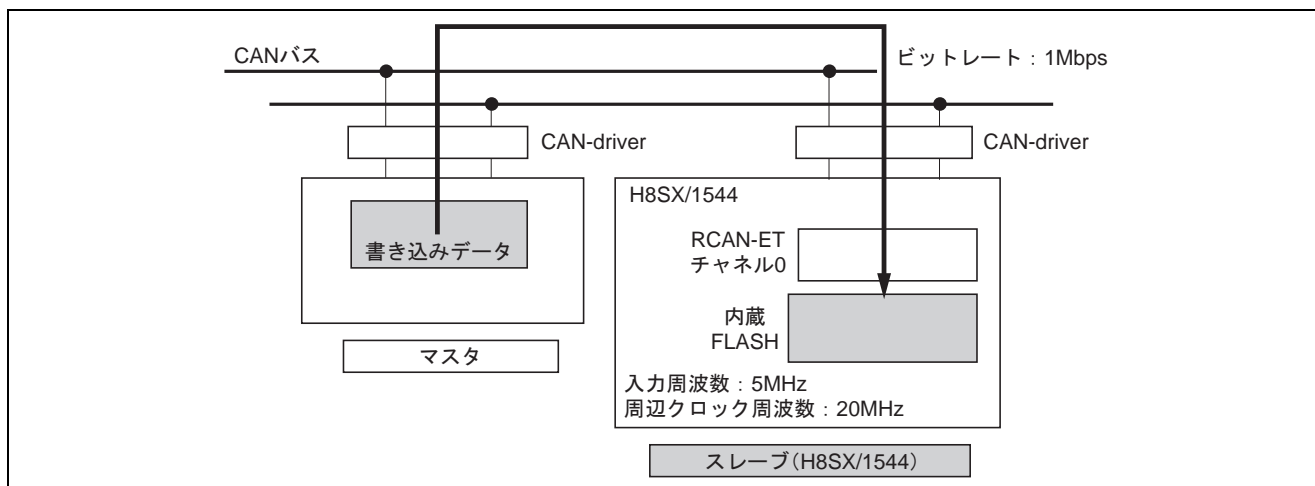


図 2.12.1 仕様

#### RCAN-ET の設定

- ビットレート : 1Mbps (システムクロック=20MHz、周辺モジュールクロック=20MHz)
- 割り込み : 使用しない
- 使用するメールボックス数 : 2 個
- メールボックスの仕様 : 表 2.12.1 参照

表 2.12.1 メールボックス仕様

メールボックス	スタンダード ID	送受信方向	機能
メールボックス 0	H'2AA	受信	コマンド、データの受信
メールボックス 1	H'555	送信	コマンドの送信

【注】 本動作例では、コマンド (4byte 固定データ) を使用して消去、書き込みの制御を行います。各コマンドの機能については、「2.12.3 ソフトウェア説明 (5) 使用定数説明」を参照してください。

2.12.2 動作説明

(1) 消去・書き込み手順

図 2.12.2 ~ 図 2.12.7 に、本動作例でのフラッシュメモリの消去、書き込み手順を示します。

ユーザープログラムモードでアプリケーションプログラムを起動します。

(書き換え制御プログラムはスレーブ側に、書き換えデータはマスタ側に予め準備します。)

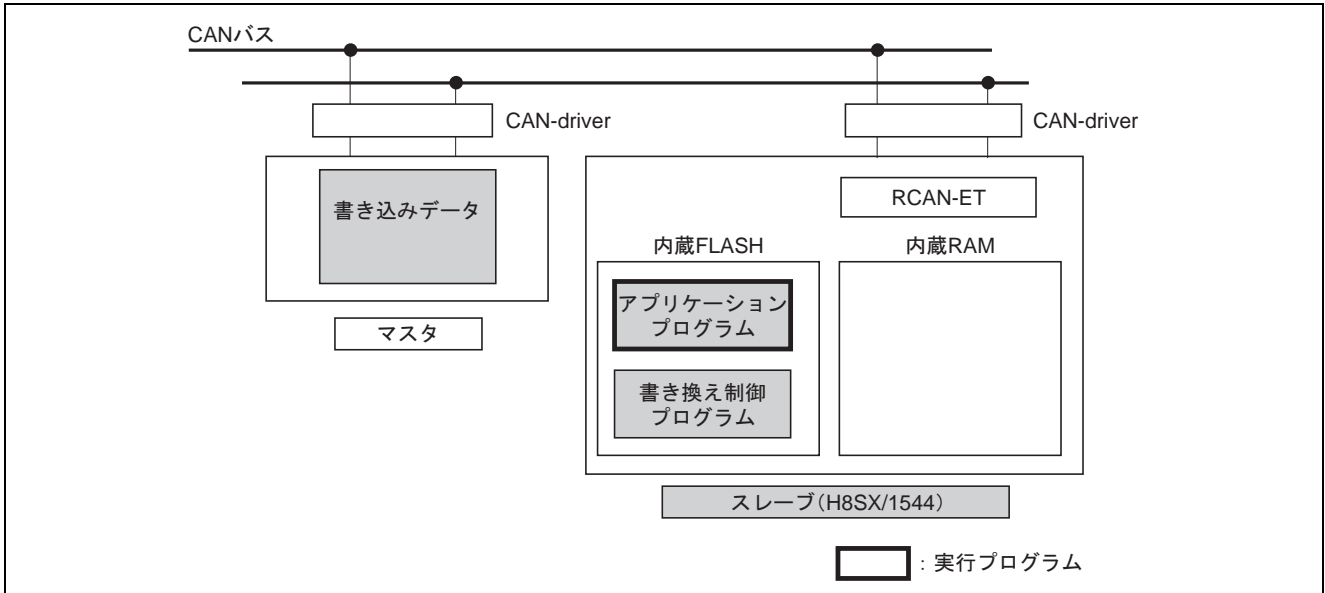


図 2.12.2 消去・書き込み手順 (1)

書き換え制御プログラムを内蔵 RAM に転送します。

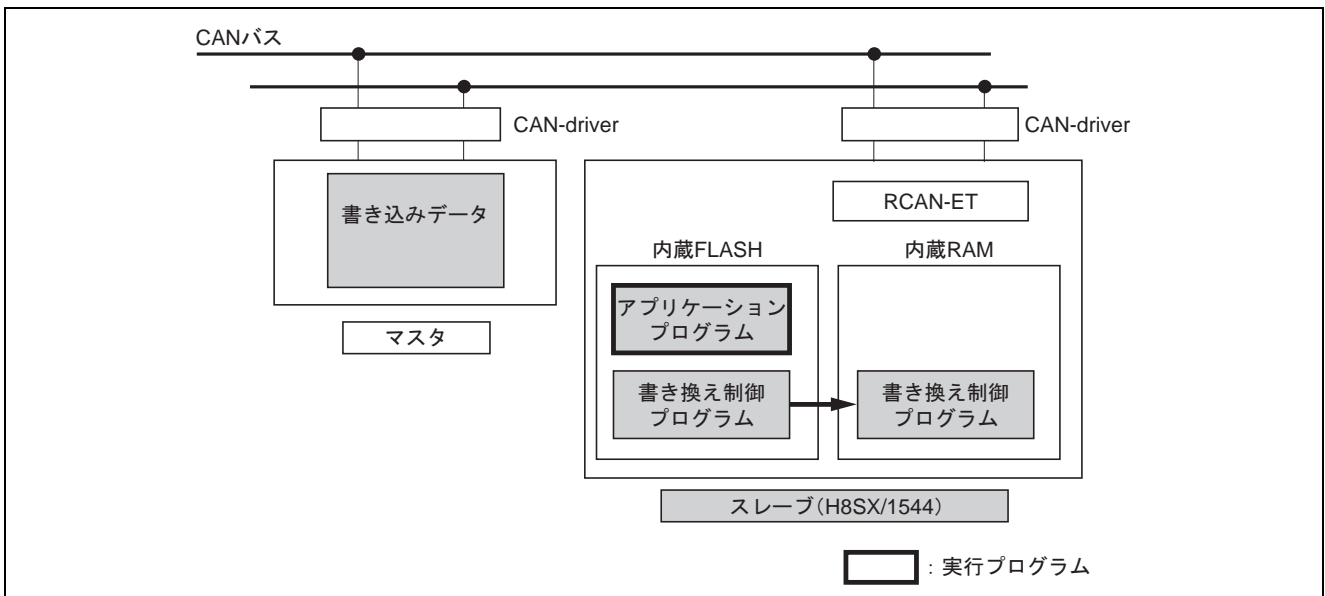


図 2.12.3 消去・書き込み手順 (2)



内蔵 RAM 上の書き換え制御プログラムを実行し、内蔵 FLASH のプログラムを全面消去します。

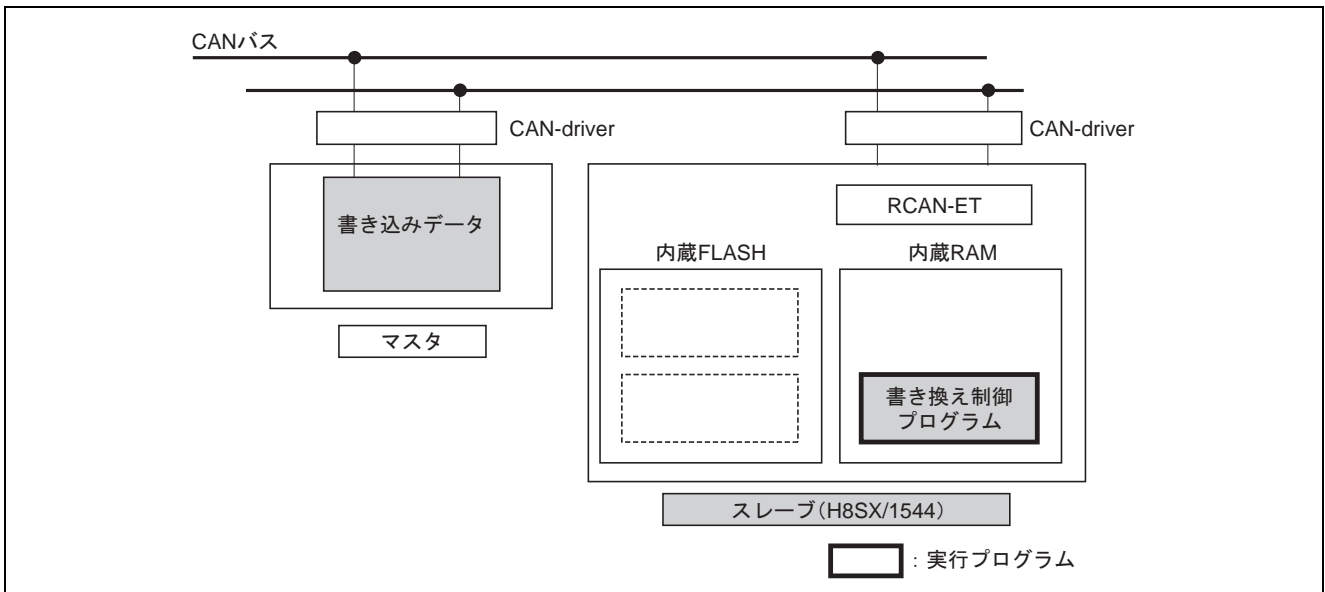


図 2.12.4 消去・書き込み手順 (3)

マスタからスレーブに対して書き込みデータ (128byte) を CAN 経由で転送し、内蔵 RAM 上に格納します。

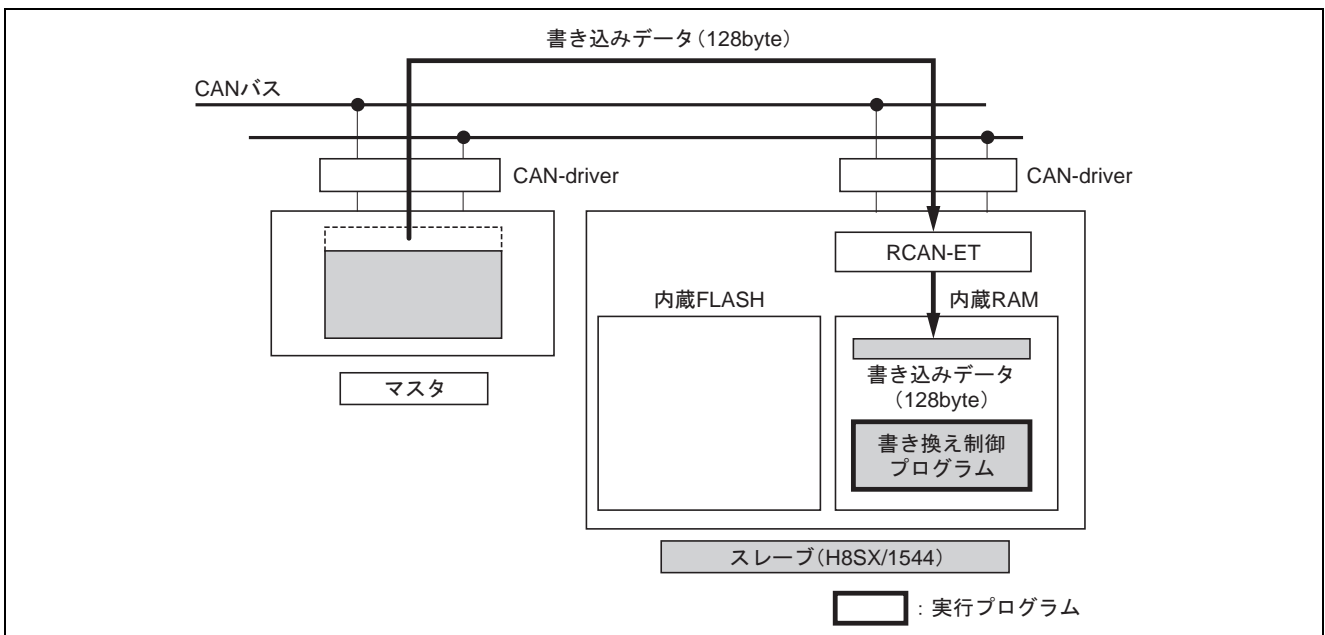


図 2.12.5 消去・書き込み手順 (4)

内蔵 RAM 上の書き込みデータ (128byte) を内蔵 FLASH に書き込みます。

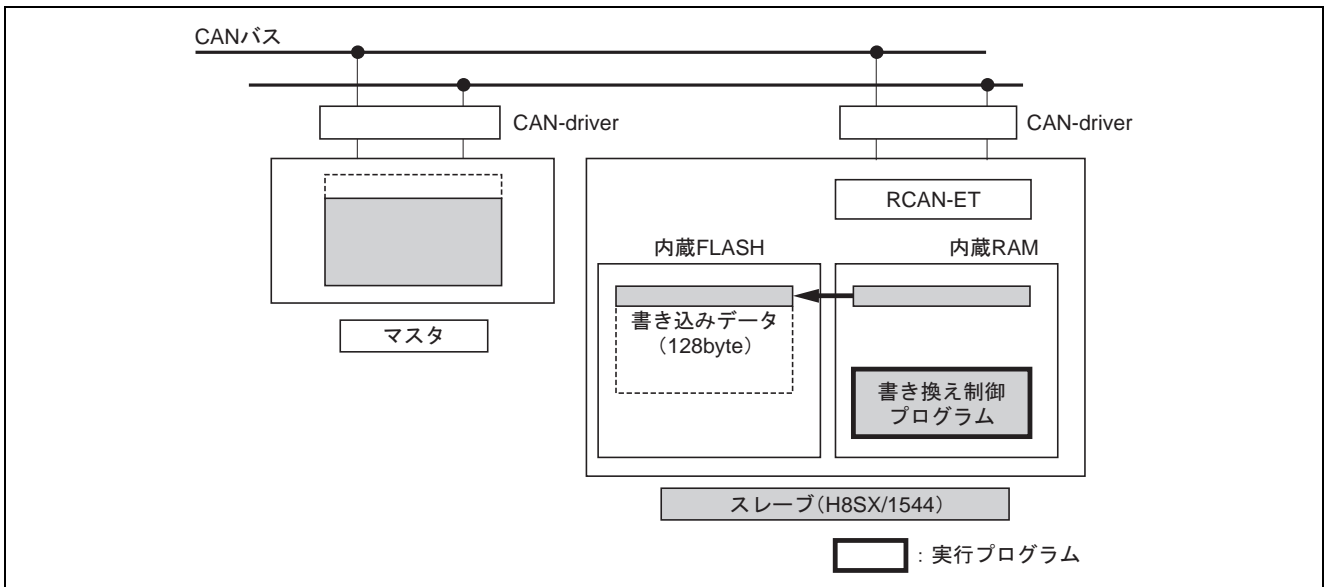


図 2.12.6 消去・書き込み手順 (5)

、を繰り返し、内蔵 FLASH を全面書き込みます。

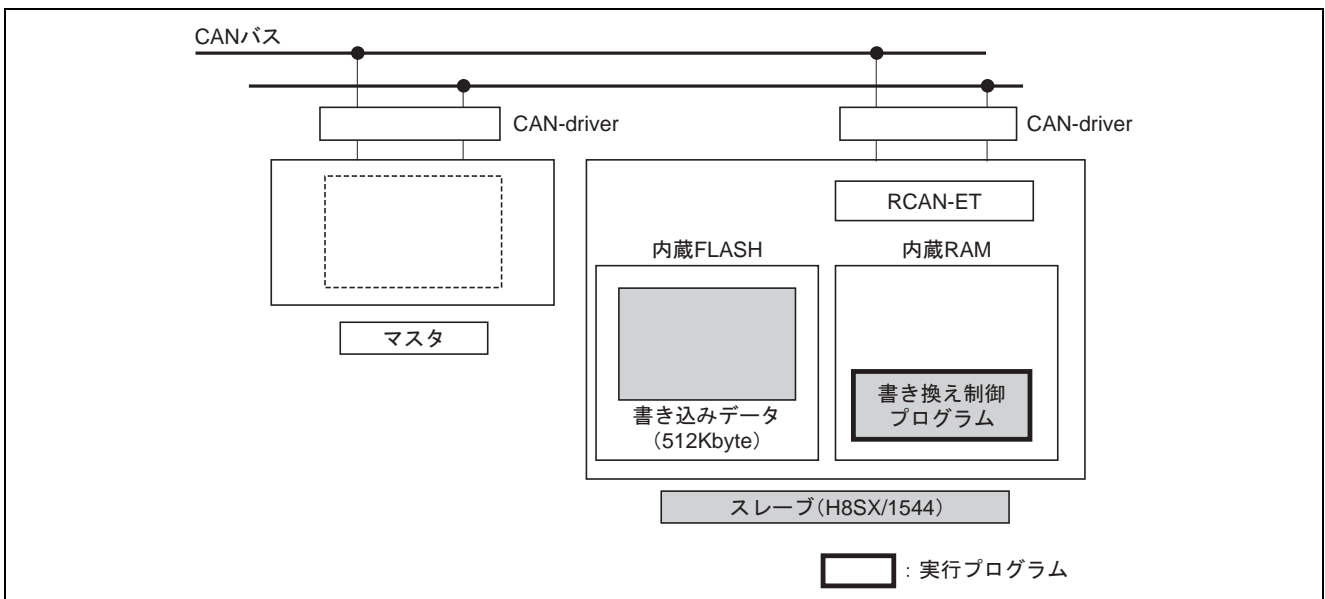


図 2.12.7 消去・書き込み手順 (6)

(2) 消去・書き込み時のコマンド/データ通信内容

図 2.12.8～図 2.12.9 に、フラッシュメモリの消去、書き込み時のコマンド/データ通信内容について示します。スレーブ側の仕様に合わせてマスタ側のプログラムを作成していただくことで、スレーブ側のフラッシュ書き換えが可能となります。

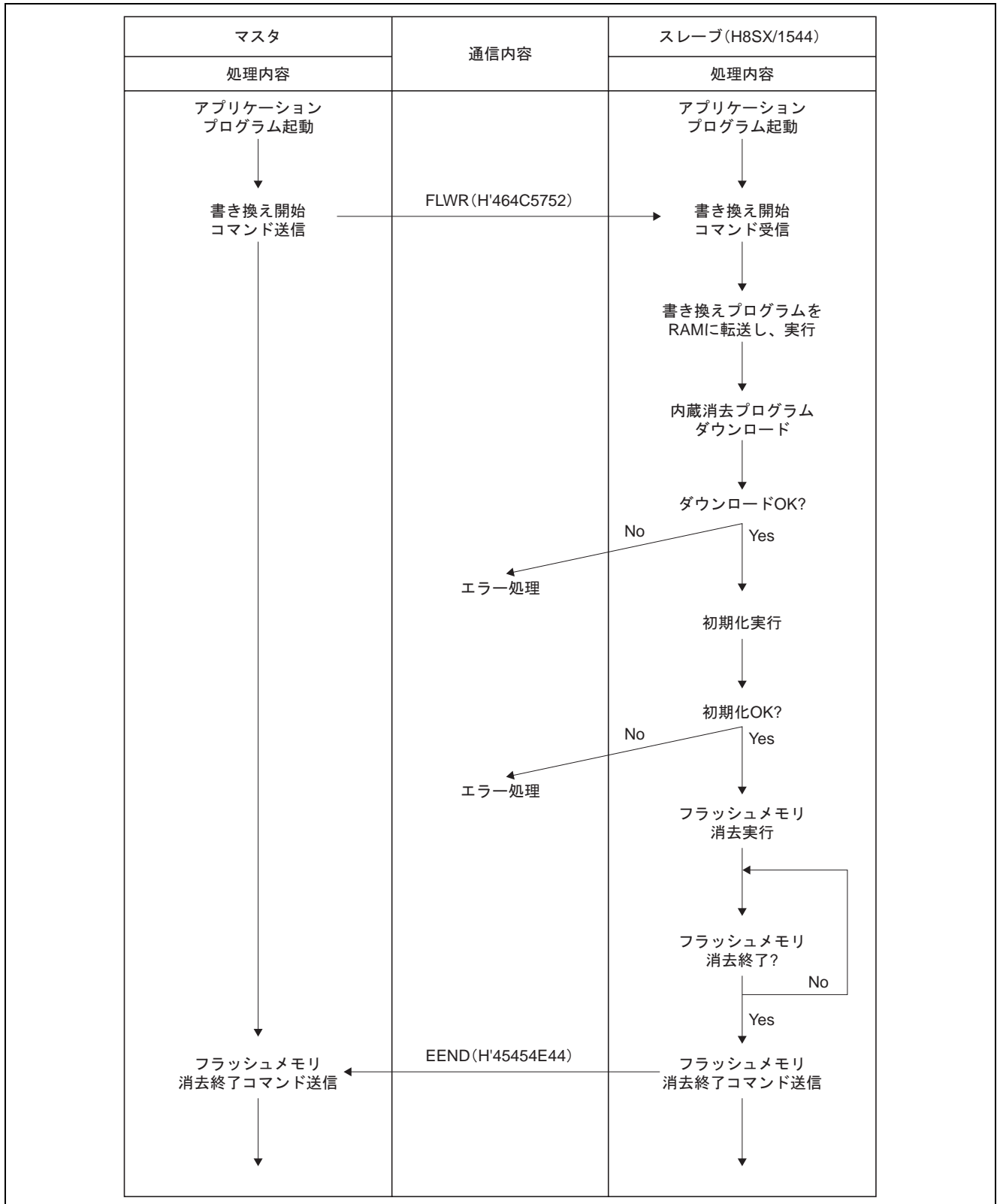


図 2.12.8 通信内容

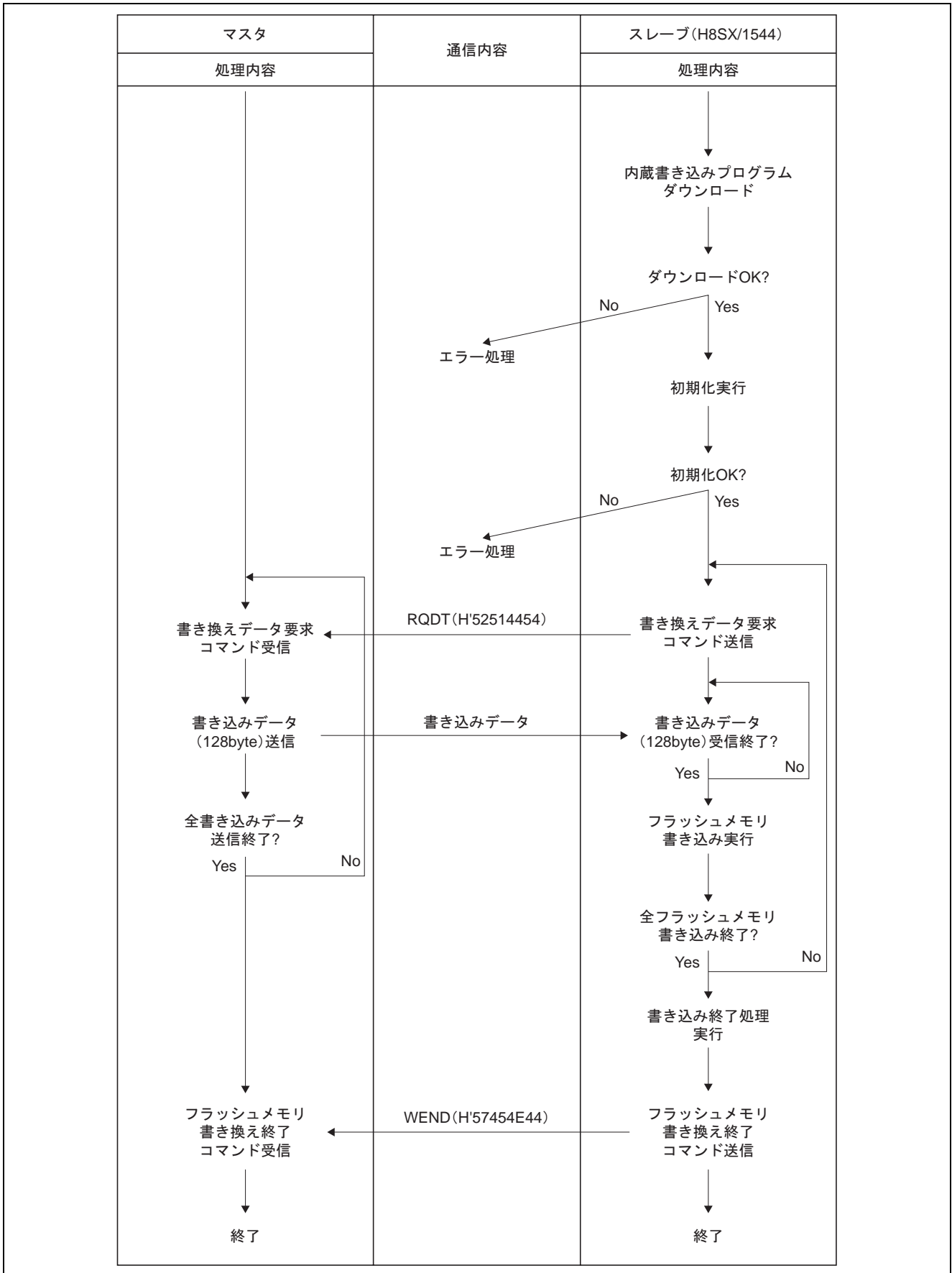


図 2.12.8 通信内容 ( 続き )

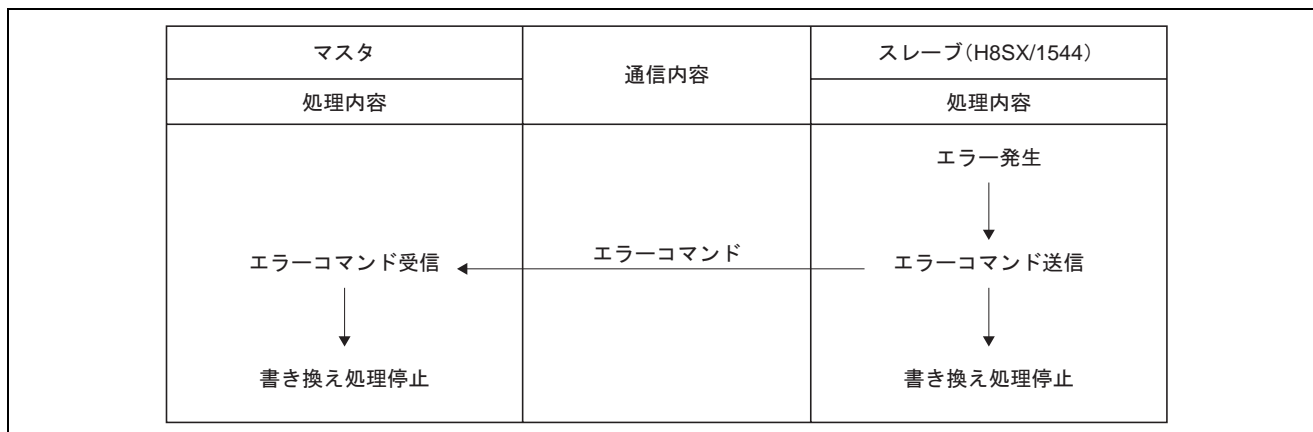


図 2.12.9 通信内容 (エラー処理)

## 2.12.3 ソフトウェア説明

## (1) モジュール説明

表 2.12.2 モジュール説明

モジュール名	ラベル名	機能	設定内容
H8SX/1544 初期設定	set_1544_init	「2.1 初期設定」参照	デバイス 個別
RCAN 初期設定	set_RCAN0_init		RCAN-ET 共通
RCAN メールボックス 初期設定	set_RCAN0MB_init		
RCAN スタート	set_RCAN0_start		
書き換え開始コマンド 受信ルーチン	CAN_RV_MB0	フラッシュメモリの書き換え開始コマンド を CAN 経由で受信。	
書き込みデータ 受信ルーチン	CAN_RV_MB0_128B	フラッシュメモリの書き込みデータを CAN 経由で受信し、内蔵 RAM に格納。	
書き換え制御コマンド 送信ルーチン	CAN_TR_MB1	フラッシュメモリの書き換え制御用コマ ンドを CAN 経由で送信。	
フラッシュメモリ 消去・書き込み処理 ルーチン	procedure	フラッシュメモリの消去、書き込み。	
消去・書き込み初期化用 ダミールーチン	FLASH_INIT	内蔵初期化プログラムへジャンプするた めのダミールーチン。	デバイス 個別
消去用ダミールーチン	FLASH_ERASE	内蔵消去プログラムへジャンプするた めのダミールーチン。	
書き込み用 ダミールーチン	FLASH_WRITE	内蔵書き込みプログラムへジャンプする ためのダミールーチン。	

## (2) 引数の説明

表 2.12.3 引数の説明

ラベル名	機能	サイズ	使用モジュール
FPEFEQ	CPU の動作周波数の設定。	unsigned long	FLASH_INIT
FEBS	消去ブロック番号の指定。	unsigned long	FLASH_ERASE
FMPAR	書き込み先の先頭アドレス。	unsigned long	FLASH_WRITE
FMPDR	データ格納領域（書き込み元）の先頭アドレス。	unsigned long	
work	送信用コマンドの設定。	unsigned long	CAN_TR_MB1

## (3) 戻り値の説明

表 2.12.4 戻り値の説明

ラベル名	機能	サイズ	使用モジュール
ERR_CHECK	消去・書き込みの初期化、消去処理、書き込み処理 結果の確認用パラメータ。	unsigned char	FLASH_INIT FLASH_ERASE FLASH_WRITE

## (4) 使用変数の説明

表 2.12.5 使用変数の説明

ラベル名	機能	サイズ	使用モジュール
_BMAIN_BGN	フラッシュメモリ書き換え制御ルーチンの先頭アドレス。	unsigned char*	main
_BMAIN_END	フラッシュメモリ書き換え制御ルーチンの最終アドレス。	unsigned char*	
_BMAIN_RAM	フラッシュメモリ書き換え制御ルーチンの転送先アドレス。	unsigned char*	
Command	受信コマンドの格納。	unsigned long	main procedure
ERASE_block	消去ブロックの設定。	unsigned short	

## (5) 使用定数の説明

表 2.12.6 使用定数の説明

ラベル名	機能	設定値	使用モジュール
RAM90	内蔵消去・書き込みプログラムのダウンロード先先頭アドレス (FTDAR レジスタの設定値)。	0x00	procedure
RAM90_TOP	内蔵消去・書き込みプログラムのダウンロード先先頭エリア。	0xFF9000	
FLWR	フラッシュメモリの書き換え開始コマンド	0x464C5752	main procedure
EDER	内蔵消去プログラムのダウンロードエラーコマンド	0x45444552	
EIER	消去初期設定時のエラーコマンド	0x45494552	
ERER	消去実行時のエラーコマンド	0x45524552	
EEND	フラッシュメモリの消去完了コマンド	0x45454E44	
WDER	内蔵書き込みプログラムのダウンロードエラーコマンド	0x57444552	
WIER	書き込み初期設定時のエラーコマンド	0x57494552	
WRER	書き込み実行時のエラーコマンド	0x57524552	
RQDT	書き込みデータ要求コマンド	0x52514454	
WEND	書き込みの完了コマンド	0x57454E44	

## (6) 使用レジスタ説明

表 2.12.7 使用レジスタ説明

モジュール名	レジスタ名	設定値	機能
H8SX/1544 初期設定	SCKCR.BIT.ICK	1	システムクロックの周波数を20MHz(入力クロック:5MHz×4)に設定。
	SCKCR.BIT.BCK	1	周辺モジュールクロックの周波数を20MHz(入力クロック:5MHz×4)に設定。
	SCKCR.BIT.PCK	1	外部バスクロックの周波数を20MHz(入力クロック:5MHz×4)に設定。
	SBYCR.BIT.SSBY	1	Sleep 命令実行後、ソフトウェアスタンバイモードに遷移するように設定。
	MSTP.CRC.BIT._RCAN01	0	RCAN-ET のモジュールストップモードを解除。
	RCANET0.RCANMON.BYTE	0x20	RCAN-ET の送受信端子をイネーブルに設定。
	P6.ICR.BIT.B4	1	P64 (67Pin) を RCAN-ET チャンネル 0 の受信端子 (CRx_0) に設定。
RCAN 初期設定	「2.1 初期設定」参照		
RCAN メール ボックス 初期設定	RCANET0.MB[0].CTRL0.WORD.H	0x0AA8	メールボックス 0 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID (H'2AA) を設定。
	RCANET0.MB[0].LAFM.WORD.H	0x0000	メールボックス 0 のスタンダード ID および IDE ビットのフィルタマスクを設定。
	RCANET0.MB[0].CTRL1.BYTE.H	0x02	メールボックス 0 を受信用に設定。
	RCANET0.MB[1].CTRL0.WORD.H	0x1554	メールボックス 1 をスタンダードフォーマット、データフレームに設定。また、スタンダード ID (H'555) を設定。
	RCANET0.MB[1].CTRL1.BYTE.H	0x00	メールボックス 1 を送信用に設定。
	RCANET0.MB[1].CTRL1.BYTE.L	0x04	データ長 (4byte) を設定。
	RCANET0.BCR1.WORD	0x4300	Pφ=20MHz 時 500kbps に設定。 (TSEG1=4(5tq), TSEG2=3(4tq), SJW=0, BSP=0, BRP=1)
	RCANET0.BCR0.WORD	0x0001	
RCAN スタート	「2.1 初期設定」参照		
書き換え開始 コマンド 受信ルーチン	RCANET0.RXPR0.WORD	-	メールボックス 0 の受信完了待ち。 (RXPR0 の bit0 が 1 にセットされるまでフラグポーリング)
	RCANET0.RXPR0.WORD	0x0001	メールボックス 0 の受信完了フラグのクリア。(クリア条件: 1 ライト)



モジュール名	レジスタ名	設定値	機能	
フラッシュ 消去・ 書き込み 処理ルーチン	FLASH.FTDAR.BYTE	RAM90 (=0x00)	内蔵プログラムのダウンロード先の先頭アドレスを H'FF9000 に設定。	
	FLASH.FECS.BYTE	0x01	ダウンロードする消去プログラムを選択。(EPVB ビットセット)	
	FLASH.FPCS.BYTE	0x01	ダウンロードする書き込みプログラムを選択。(PPVS ビットセット)	
	FLASH.FCCS.BYTE	0x01	消去・書き込みプログラムのダウンロードを許可。(SCO ビットセット)	
	FLASH.FKEY		0xA5	FCCS の SCO ビットの書き込みを許可。
			0x5A	フラッシュメモリの消去・書き込みを許可。
			0x00	ソフトウェアプロテクト。
	RAM90_TOP(=DPFR)	-	内蔵消去・書き込みプログラムのダウンロードエラーの有無の確認用パラメータ。(0:正常動作、0以外:エラー)	
	FPEFEQ	0x000007D0	CPU の動作周波数の設定用パラメータ。(動作周波数を 20MHz に設定)	
	ERR_CHECK(=FPFR)	-	消去・書き込みの初期化、消去処理、書き込み処理結果の確認用パラメータ。(0:正常動作、0以外:エラー)	
	FMPAR	0x00000000	書き込み先の先頭アドレス設定用パラメータ。(先頭アドレスを H'0 に設定)	
	FMPDR	0xFFB100	書き込み元の先頭アドレス設定用パラメータ。 (先頭アドレスを H'FFB100 に設定)	
FEBS	0x00000000	消去ブロック番号の設定用パラメータ。		
書き換え制御 コマンド 送信ルーチン	RCANET0.TXPR0.LONG	0x00000002	メールボックス 1 を送信待ち状態に設定。	
	RCANET0.TXACK0.WORD	-	メールボックス 1 の送信完了待ち。 (TXACK0 の bit1 が 1 にセットされるまでフラグポーリング)	
	RCANET0.TXACK0.WORD	0x0002	メールボックス 1 の送信完了フラグのクリア。(クリア条件: 1 ライト)	

【注】 表中のビット名は、ハードウェアマニュアルのビット名を使用しています。

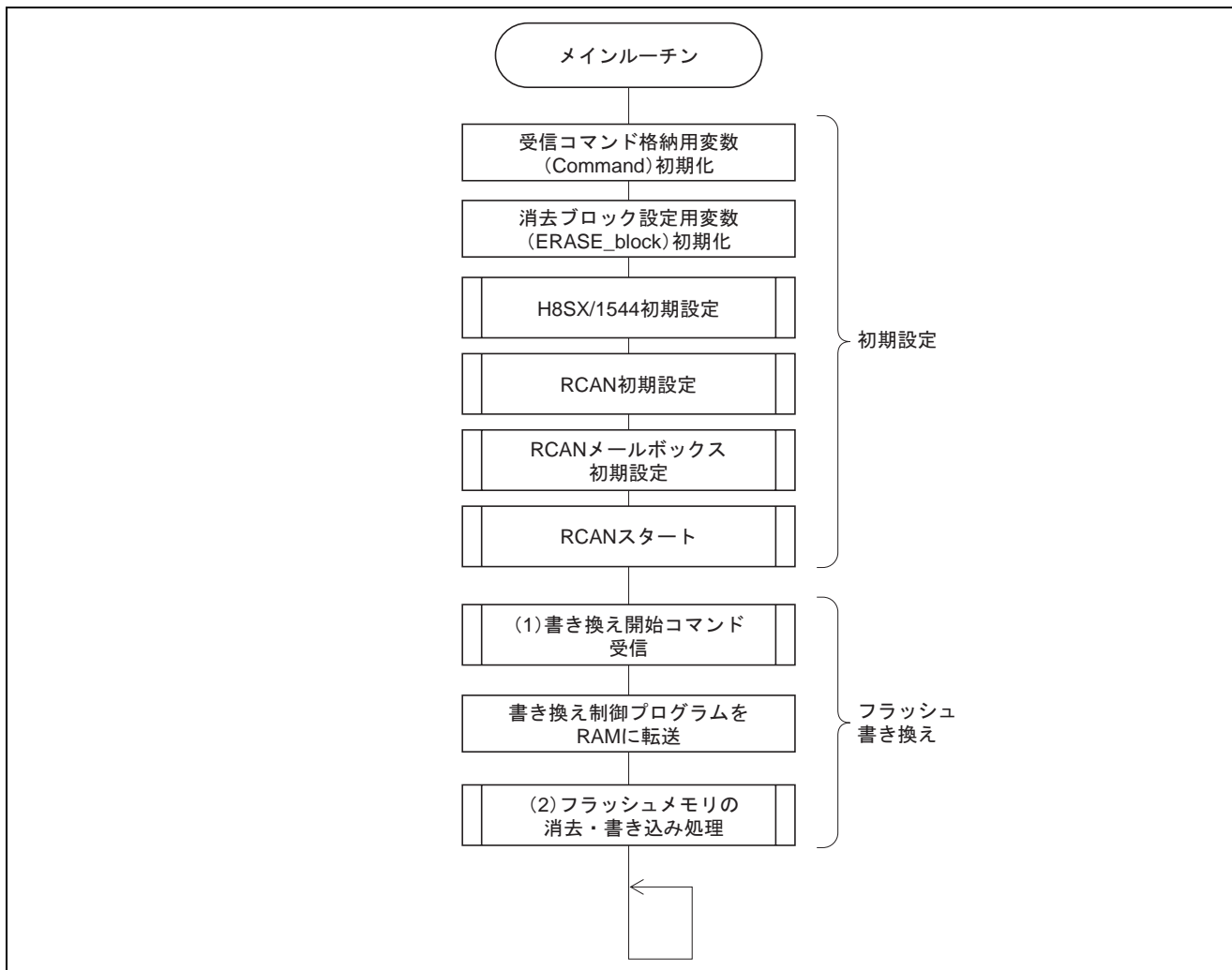
(7) メモリマップ

図 2.12.10 に、本動作例のメモリマップを示します。

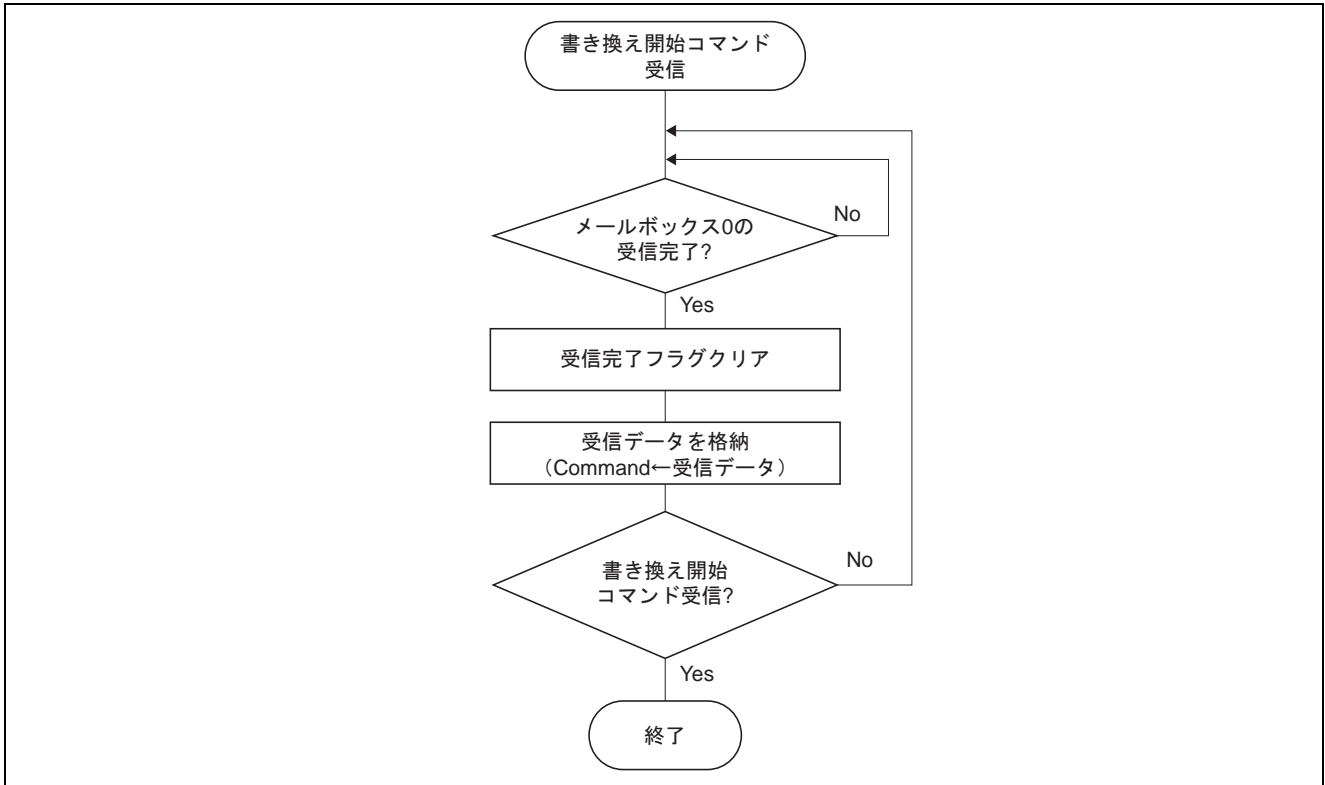
H'000000	ベクタテーブル
H'000800	メインプログラム
H'001000	P、Cセクション
H'0010FA	書き換え制御プログラム (P_BMAINセクション)
H'0012CA	消去用ダミープログラム (P_ERASEセクション)
H'0012CC	書き込み用ダミープログラム (P_WRITEセクション)
H'0012CE	消去・書き込み初期化用ダミープログラム (P_EWINIセクション)
H'FF6000	Bセクション
H'FF7000	P_BMAINのオーバーラップエリア
H'FF9010	P_ERASE/P_WRITEの オーバーラップエリア
H'FF9020	P_EWINIのオーバーラップエリア
H'FFB100	書き込みデータ格納エリア(128byte)

図 2.12.10 メモリマップ

2.12.4 フローチャート  
メインルーチン

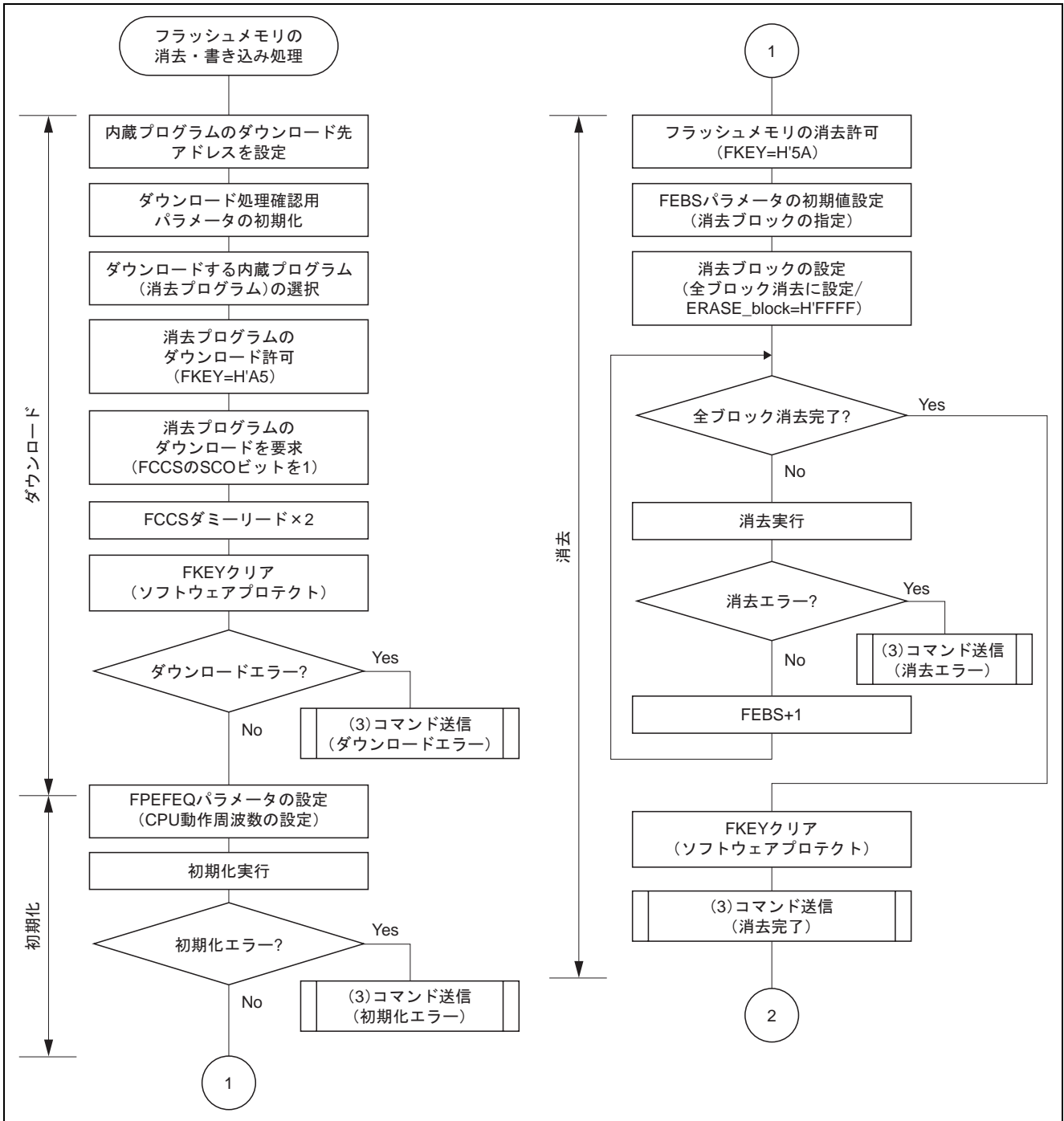


(1) 書き換え開始コマンド受信ルーチン (RCAN-ET 共通設定)

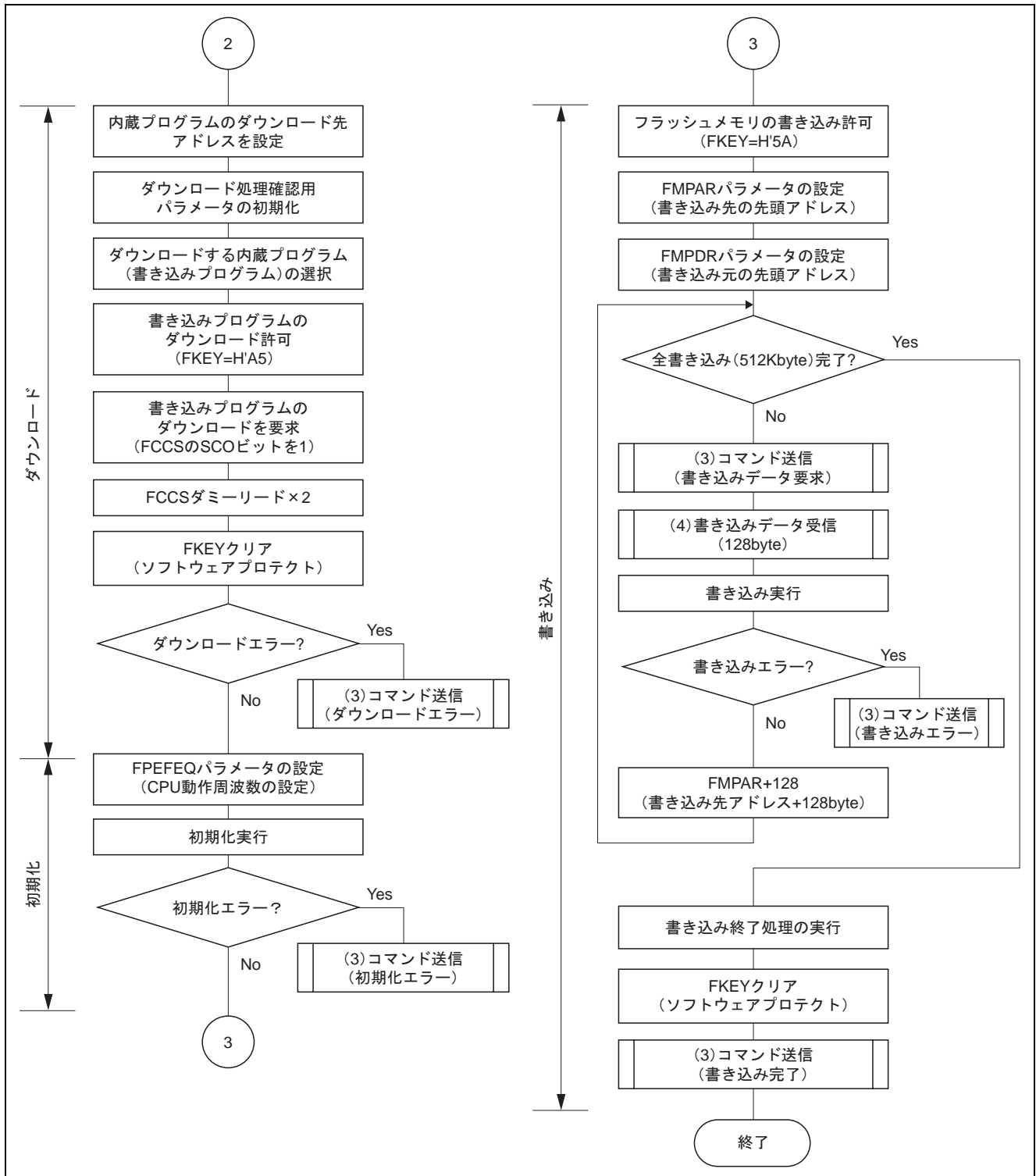


(2) フラッシュメモリ消去・書き込み処理ルーチン (デバイス個別設定)

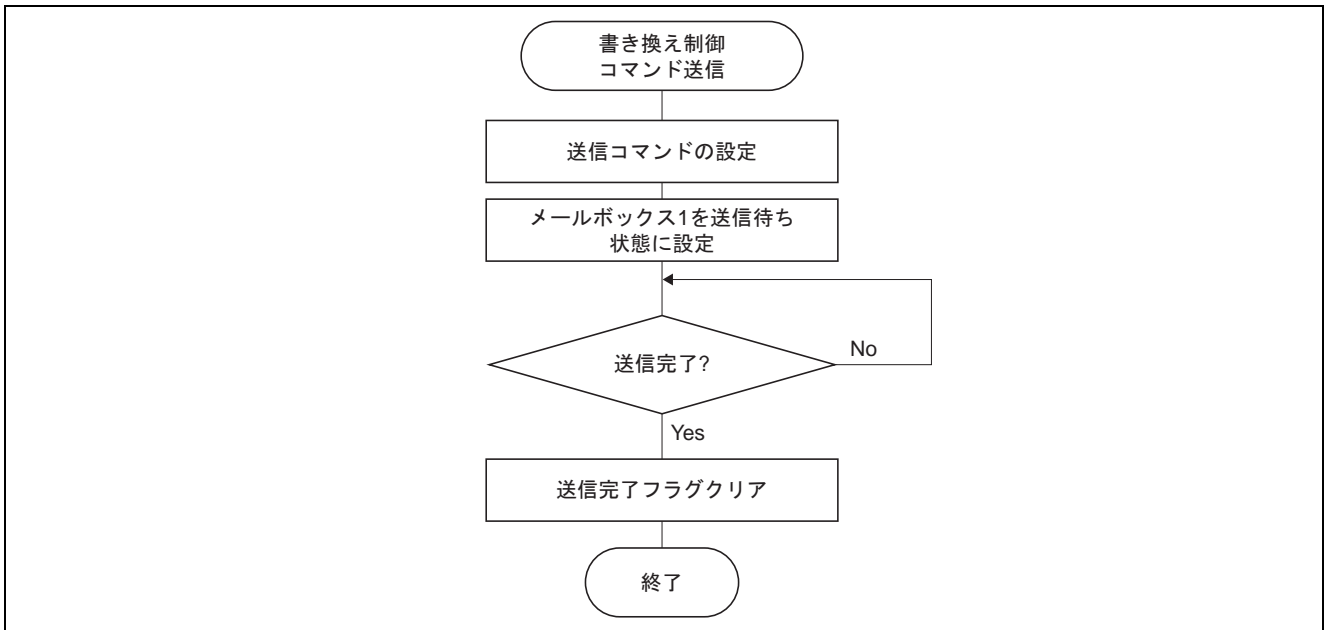
(a) 消去



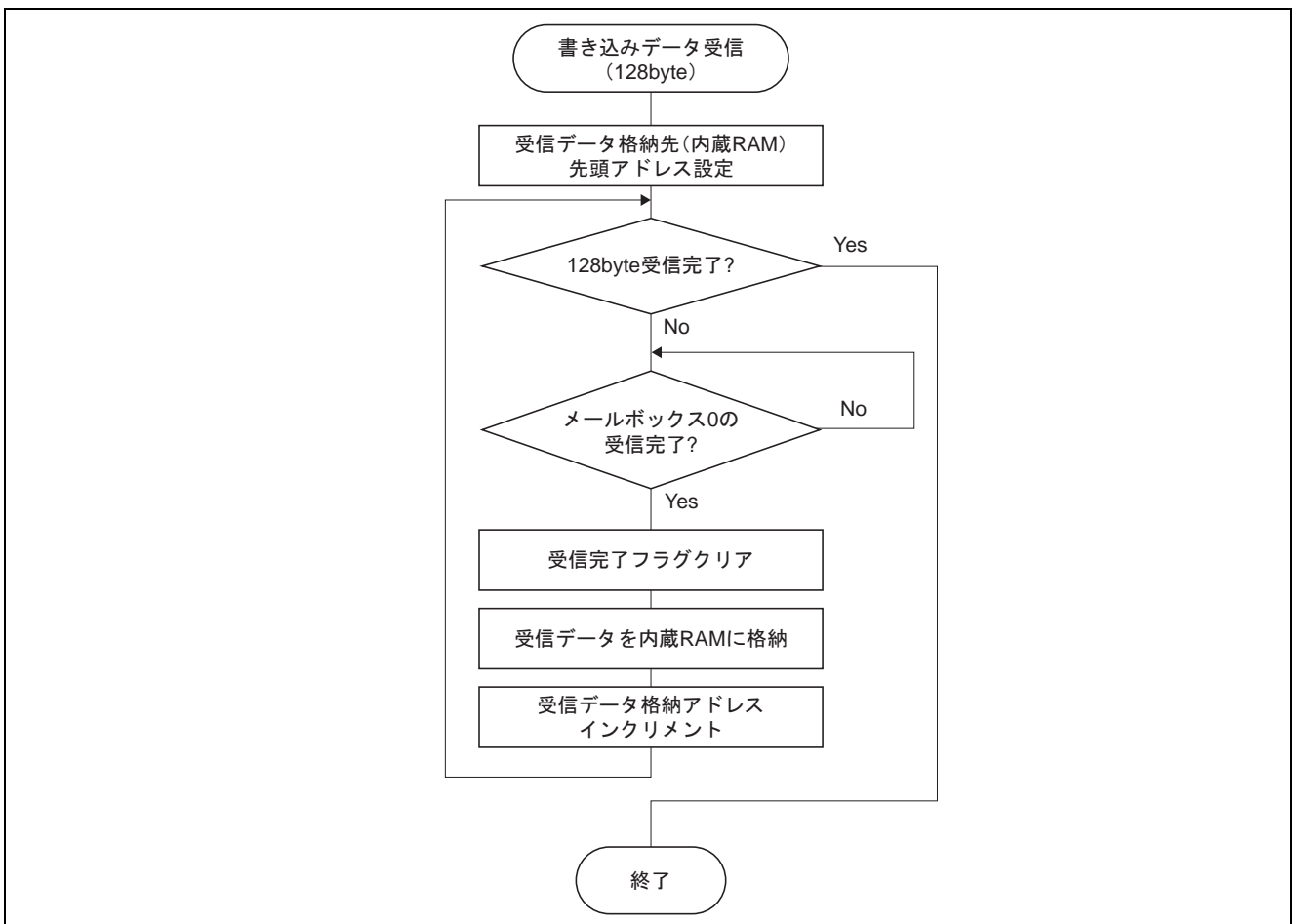
### (b) 書き込み



(3) 書き換え制御コマンド送信ルーチン (RCAN-ET 共通設定)



(4) 書き込みデータ受信ルーチン (RCAN-ET 共通設定)



## 2.12.5 プログラムリスト

### (1) main.c

```

/*****
/* Filename      :   main.c
/* Written       :   '06/06/01 REV.1.00
/* Purpose       :   for H8SX/1544 RCAN-ET
/*****
#include <machine.h>
#include "iodefine.h"
#include "reg.h"
/*-----*/
void main(void);
void set_1544_init(void);
void CAN_RV_MB0(void);
extern void set_RCAN0_init(void);
extern void set_RCAN0MB_init(void);
extern void set_RCAN0_start(void);
extern void procedure(void);
/*-----*/
unsigned long Command;
unsigned short ERASE_block;
extern unsigned char *_BMAIN_BGN;
extern unsigned char *_BMAIN_END;
extern unsigned char *_BMAIN_RAM;
/*****
/*      Main program
/*****
#pragma section _MAIN
#pragma entry main(sp=0xFFC000)
void main(void)
{

    unsigned int i;
    unsigned char *src, *dst;

    /*** INIT Value ***/
    Command = 0;
    ERASE_block = 0;

    /*** H8SX/1544 init ***/
    set_1544_init();

    /*** RCAN init ***/
    set_RCAN0_init();
    set_RCAN0MB_init();
    set_RCAN0_start();

    /*** 書換え開始コマンド受信 ***/
    CAN_RV_MB0();

```



```

    /*** RAM 転送 ***/
    for(src=_BMAIN_BGN, dst=_BMAIN_RAM; src<=_BMAIN_END; src++, dst++) {
        *dst = *src; /* 書換え手続きプログラムのRAMに転送 */
    }

    /*** FLASH 書込み ***/
    procedure(); /* 書込み制御プログラムの実行 */

    while(1);
}

/*****
/*      H8SX/1544 Initialize routine
*****/
void set_1544_init(void){

    /*** SYSTEM ***/
    SCKCR.BIT.ICK = 1; /* システムクロック設定 (5MHz × 4=20MHz) */
    SCKCR.BIT.PCK = 1; /* 周辺クロック (5MHz × 4=20MHz) */
    SCKCR.BIT.BCK = 1; /* 外部バスクロック (5MHz × 4=20MHz) */
    SBYCR.BIT.SSBY = 1; /* ソフトウェアスタンバイモード */
    MSTP.CRC.BIT._RCAN01 = 0; /* モジュールストップモード解除: RCAN */

    /*** IO ***/
    RCANET0.RCANMON.BYTE = 0x20; /* RCAN-ET 送受信端子レベル */
    P6.ICR.BIT.B4 = 1; /* P64 を CRx_0 (入力端子) に設定 */
}

/*****
/* receive command routine
*****/
void CAN_RV_MB0(void){

    unsigned int i;

    do {
        while ((RCANET0.RXPR0.WORD & 0x0001) != 0x0001); /* 受信完了待ち */
        RCANET0.RXPR0.WORD = 0x0001; /* 受信完了フラグクリア (クリア条件: 1 ライト) */
        for(i = 0; i < 4; i++){
            Command <<= 8;
            Command |= RCANET0.MB[0].MSG_DATA[i]; /* 受信データを格納 */
        }
    } while(Command != FLWR); /* 書換え開始コマンド (FLWR コマンド) 受信? */
}
/*****

```

## (2) rcan.c

```

/*****
/* Filename      : rcan.c                               */
/* Written       : '06/06/01 REV.1.00                   */
/* Purpose       : for H8SX/1544 RCAN-ET                 */
/*****
#include "iodefine.h"
/*-----*/
void set_RCAN0_init(void);
void set_RCAN0MB_init(void);
void set_RCAN0_start(void);
/*****
/*      RCAN Initialize routine                          */
/*****
void set_RCAN0_init(void){

    unsigned int i,j;

    RCANET0.MCR.WORD |= 0x0001;          /* リセットリスト(HW リセット時は自動的にセット) */
    while((RCANET0.GSR.WORD & 0x0008) != 0x0008);/* GSR3=1?(RCAN-ET リセット状態) */
    while((RCANET0.IRR.WORD & 0x0001) != 0x0001);/* IRR0=1?(リセット/ホルト/スリープ 割り込み)*/
    RCANET0.IRR.WORD = 0x0001;          /* IRR0 クリア(クリア条件:1 ライト) */
    RCANET0.MCR.WORD |= 0x8000;          /* ID 並び替え: MCR15=1(初期値)に設定 */

    for(i = 0;i < 16;i++){              /* メールボックス(RAMエリア)初期化 */
        RCANET0.MB[i].CTRL0.WORD.H = 0;
        RCANET0.MB[i].CTRL0.WORD.L = 0;
        RCANET0.MB[i].LAFM.WORD.H = 0;
        RCANET0.MB[i].LAFM.WORD.L = 0;
        for(j = 0;j < 8;j++){
            RCANET0.MB[i].MSG_DATA[j] = 0;
        }
    }

}
/*****
/*      RCAN Mailbox Initialize routine                  */
/*****
void set_RCAN0MB_init(void){

    RCANET0.MB[0].CTRL0.WORD.H = 0x0AA8; /* ID:H'2AA,スタンダードフォーマット,データフレーム */
    RCANET0.MB[0].LAFM.WORD.H = 0x0000; /* STD_LAFM, IDE_LAFM の設定 */
    RCANET0.MB[0].CTRL1.BYTE.H = 0x02; /* メールボックス0を受信用に設定 */

    RCANET0.MB[1].CTRL0.WORD.H = 0x1554; /* ID:H'555,スタンダードフォーマット,データフレーム */
    RCANET0.MB[1].CTRL1.BYTE.H = 0x00; /* メールボックス1を送信用に設定 */
    RCANET0.MB[1].CTRL1.BYTE.L = 0x04; /* データ長:4byte */

```

```

/* ビットレート: 1Mbps */
RCANET0.BCR1.WORD = 0x4300; /* TSEG1=3(4tq), TSEG2=2(3tq), SJW=0, BSP=0, (pφ=20MHz) */
RCANET0.BCR0.WORD = 0x0000; /* BRP=0 */

}

/*****
/*      RCAN start routine
/*****
void set_RCAN0_start(void){

    RCANET0.MCR.WORD &= 0xFFFE; /* MCR0 クリア */
    while ((RCANET0.GSR.WORD & 0x0008) != 0x0000); /* GSR3=0? */

}
/*****

```

## (3) procedure.c

```

/*****
/* Filename      : procedure.c
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
*****/
#include "iodefine.h"
#include "reg.h"
/*-----*/
void procedure(void); /* 書き込み制御プログラム */
void CAN_TR_MB1(unsigned long work); /* 書き込み・消去制御コマンド送信 */
void CAN_RV_MB0_128B(void); /* 書き込みデータ(128byte)受信 */
extern unsigned char FLASH_INIT(unsigned long FPEFEQ); /* タミ関数 */
extern unsigned char FLASH_ERASE(unsigned long FEBS); /* タミ関数 */
extern unsigned char FLASH_WRITE(unsigned long FMPDR,unsigned long FMPAR); /* タミ関数 */
/*-----*/
extern unsigned long Command; /* 受信コマンド格納用変数 */
extern unsigned short ERASE_block; /* 消去ブロック設定用変数 */
/*****
/* procedure program routine
*****/
#pragma section _BMAIN
void procedure(void){

    unsigned long FPEFEQ; /* 書き込み・消去アドレス */
    unsigned long FEBS; /* 書き込み・消去アドレス */
    unsigned long FMPAR; /* 書き込み・消去アドレス */
    unsigned long FMPDR; /* 書き込み・消去アドレス */
    unsigned char ERR_CHECK; /* 書き込み・消去アドレス */
    unsigned char buff; /* FCCSタミリット用変数 */

/**** 消去 ****/
    /* タミロード処理 */
    FLASH.FTDAR.BYTE = RAM90; /* タミロード先アドレス設定(H'FF9000) */
    RAM90_TOP = 0xFF; /* タミロード処理確認用アドレスクリア */
    FLASH.FECS.BYTE = 0x01; /* 消去プログラム選択(EPVBビットセット) */
    FLASH.FKEY = 0xA5; /* 消去プログラムタミロード許可 */
    FLASH.FCCS.BYTE |= 0x01; /* タミロード要求(SCOビットセット) */
    buff = FLASH.FCCS.BYTE; /* FCCSタミリット×2 */
    buff = FLASH.FCCS.BYTE;
    FLASH.FKEY = 0x00; /* FKEYクリア(ソフトウェアリセット) */
    if(RAM90_TOP != 0x00){ /* タミロードエラー? */
        CAN_TR_MB1(EDER); /* タミロードエラーコマンド(EDERコマンド)送信 */
        while(1);
    }
    /* 初期設定 */
    FPEFEQ = 0x000007D0; /* CPU動作周波数設定:20MHz(H'07D0) */
    ERR_CHECK = FLASH_INIT(FPEFEQ); /* 初期化実行 */
    if(ERR_CHECK != 0){ /* 初期設定エラー? */
        CAN_TR_MB1(EIER); /* 初期設定エラーコマンド(EIERコマンド)送信 */
        while(1);
    }
}

```

```

/* 消去処理 */
FLASH.FKEY = 0x5A; /* フラッシュメモリ消去許可 */
FEBS = 0x00000000; /* 消去ブロック指定用ハーフワード設定 */
ERASE_block = 0xFFFF; /* 消去ブロック設定：全ブロック消去 */
while(ERASE_block != 0x0000){ /* 全ブロック消去完了? */
    if((ERASE_block & 0x0001)==0x0001){
        ERR_CHECK = FLASH_ERASE(FEBS); /* 消去実行 */
        if(ERR_CHECK != 0){ /* 消去エラー? */
            CAN_TR_MB1(ERER); /* 消去エラーコマンド (ERER コマンド)送信 */
            while(1);
        }
    }
    ERASE_block = (ERASE_block>>1);
    FEBS++;
}
FLASH.FKEY = 0x00; /* FKEY クリア(ソフトウェアリセット) */
CAN_TR_MB1(EEND); /* 消去完了コマンド (EEND コマンド)送信 */
/**** 書込み ****/
/* ダウンロード処理 */
FLASH.FTDAR.BYTE = RAM90; /* ダウンロード先アドレス設定 (H'FF9000) */
RAM90_TOP = 0xFF; /* ダウンロード処理確認用ハーフワードクリア */
FLASH.FPCS.BYTE = 0x01; /* 書込みブロックサイズ選択 (PPVSビットセット) */
FLASH.FKEY = 0xA5; /* 書込みブロックサイズダウンロード許可 */
FLASH.FCCS.BYTE |= 0x01; /* ダウンロード要求 (SCOビットセット) */
buff = FLASH.FCCS.BYTE; /* FCCS デミリット × 2 */
buff = FLASH.FCCS.BYTE;
FLASH.FKEY = 0x00; /* FKEY クリア(ソフトウェアリセット) */
if(RAM90_TOP != 0x00){ /* ダウンロードエラー? */
    CAN_TR_MB1(WDER); /* ダウンロードエラーコマンド (WDER コマンド)送信 */
    while(1);
}
/* 初期設定 */
FPEFEQ = 0x000007D0; /* CPU 動作周波数設定：20MHz (H'07D0) */
ERR_CHECK = FLASH_INIT(FPEFEQ); /* 初期化実行 */
if(ERR_CHECK != 0){ /* 初期設定エラー? */
    CAN_TR_MB1(WIER); /* 初期設定エラーコマンド (WIER コマンド)送信 */
    while(1);
}
/* 書込み処理 */
FLASH.FKEY = 0x5A; /* フラッシュメモリ書き込み許可 */
FMPAR = 0x00000000; /* 書込み先アドレス */
FMPDR = 0xFFB100; /* 書込み元先アドレス */
while(FMPAR < 0x00080000){ /* 全書込み完了? (512kbyte) */
    CAN_TR_MB1(RQDT); /* 書込みデータ要求コマンド (RQDT コマンド)送信 */
    CAN_RV_MB0_128B(); /* 書込みデータ (128byte) 受信 */
    ERR_CHECK = FLASH_WRITE(FMPDR, FMPAR); /* 書込み実行 */
    if(ERR_CHECK != 0){ /* 書込みエラー? */
        CAN_TR_MB1(WRER); /* 書込みエラーコマンド (WRER コマンド)送信 */
        while(1);
    }
    FMPAR += 0x80;
}

```

```

FLASH_WRITE(0xF0F0F0F0,0x0F0F0F0F); /* 書き込み終了処理の実行 */
FLASH.FKEY = 0x00; /* FKEY クリア(ソフトウェアリセット) */

CAN_TR_MB1(WEND); /* 書き込み完了コマンド (WEND コマンド)送信 */

while(1);
}

/*****
/* send command routine */
/*****
void CAN_TR_MB1(unsigned long work){

    unsigned int i;

    for(i = 4;i > 0;i--){ /* 送信データ設定 */
        RCANET0.MB[1].MSG_DATA[i-1] = (unsigned char)(work&0x000000FF);
        work >>= 8;
    }
    RCANET0.TXPR0.LONG = 0x00000002; /* MB1 を送信待ち状態に設定 */
    while ((RCANET0.TXACK0.WORD & 0x0002) != 0x0002); /* 送信完了待ち */
    RCANET0.TXACK0.WORD = RCANET0.TXACK0.WORD; /* 送信完了フラグクリア(クリア条件：1バイト) */

}

/*****
/* receive write data routine */
/*****
void CAN_RV_MB0_128B(void){

    unsigned char *buff;
    unsigned int i,j;

    buff = (unsigned char*)0xFFB100; /* 受信データ格納先アドレス */
    for(i = 0;i < 16; i++){ /* 8byte×16=128byte */
        while (RCANET0.RXPR0.WORD != 0x0001); /* 受信完了待ち */
        RCANET0.RXPR0.WORD = 0x0001; /* 受信完了フラグクリア(クリア条件：1バイト) */
        for(j = 0;j < 8; j++){
            *buff = RCANET0.MB[0].MSG_DATA[j]; /* 受信データ格納 */
            buff++;
        }
    }

}

/*****

```

## (4) flash.c

```

/*****/
/* Filename      : flash.c                               */
/* Written       : '06/06/01 REV.1.00                   */
/* Purpose       : for H8SX/1544 RCAN-ET                 */
/*****/
unsigned char FLASH_INIT(unsigned long FPEFEQ); /* 消去・書込み初期化用ダミー関数 */
unsigned char FLASH_ERASE(unsigned long FEBS); /* 消去用ダミー関数 */
unsigned char FLASH_WRITE(unsigned long FMPDR,unsigned long FMPAR);/* 書込みダミー関数 */
/*****/
/* flash initialize dummy routine                       */
/*****/
#pragma section _EWINI
unsigned char FLASH_INIT(unsigned long FPEFEQ){

}

/*****/
/* flash erase dummy routine                           */
/*****/
#pragma section _ERASE
unsigned char FLASH_ERASE(unsigned long FEBS){

}

/*****/
/* flash write dummy routine                           */
/*****/
#pragma section _WRITE
unsigned char FLASH_WRITE(unsigned long FMPDR,unsigned long FMPAR){

}
/*****/

```

## (5) reg.h

```

/*****
/* Filename      : reg.h
/* Written       : '06/06/01 REV.1.00
/* Purpose       : for H8SX/1544 RCAN-ET
*****/

/**** 書き込み・消去制御マント ****/
#define FLWR  0x464C5752
#define EDER  0x45444452
#define EIER  0x45494552
#define ERER  0x45524552
#define EEND  0x45454E44
#define WDER  0x57444452
#define WIER  0x57494552
#define WRER  0x57524552
#define RQDT  0x52514454
#define WEND  0x57454E44

/**** 内蔵プログラムのダウンロード先アドレス ****/
#define RAM90  0x00
#define RAM90_TOP (*(volatile unsigned char *)0xFF9000)
#define RAMA0  0x01
#define RAMA0_TOP (*(volatile unsigned char *)0xFFA000)
#define RAMB0  0x02
#define RAMB0_TOP (*(volatile unsigned char *)0xFFB000)
    
```



ホームページとサポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>

[csc@renesas.com](mailto:csc@renesas.com)

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2007.06.07	—	新規作成

## 本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の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、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊社営業窓口で最新の情報をご確認頂きますとともに、弊社ホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意下さい。
5. 本資料に記載した情報は、正確を期すため慎重に制作したものです。万一本資料の記述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、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会下さい。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないで下さい。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
  - 1) 生命維持装置。
  - 2) 人体に埋め込み使用するもの。
  - 3) 治療行為（患部切り出し、薬剤投与等）を行なうもの。
  - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて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、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に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。弊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、社会的損害など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、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエンジニアリング処理等、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願い致します。
  11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
  12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断り致します。
  13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会下さい。