

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ひ申し上げます。

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、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
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。

# SH7763 グループ

## イーサネット受信設定例

### 要旨

本アプリケーションノートは、SH7763 グループに搭載のイーサネット機能の RMII (Reduced Media Independent Interface) インタフェース 1 ポートを使用した受信設定例を示します。

### 動作確認デバイス

SH7763

### 目次

1. はじめに.....	2
2. 応用例の説明.....	3
3. 参考プログラムリスト.....	18
4. 参考ドキュメント.....	29

## 1. はじめに

### 1.1 仕様

- 本応用例ではイーサネットフレームを 10 フレーム受信します。受信のたびにフレーム受信割り込みを使用し、1 フレームずつユーザバッファにコピーします。

### 1.2 使用機能

- ギガビットイーサネットコントローラ (GETHER)
- 割り込みコントローラ (INTC)
- 汎用入出力ポート (GPIO)

### 1.3 適用条件

- 評価ボード: 日立超 LSI システムズ製 SH7763 Solution Engine 型番 MS7763SE02  
イーサネット PHY: Micrel 社 KSZ8721BL
- マイコン: SH7763 (R5S77631AY266BGV)
- 動作周波数: CPU クロック: 266.66MHz  
ローカルバスクロック: 66.66MHz  
DDR-SDRAM I/F クロック: 133.33MHz  
周辺バスクロック 0: 66.67MHz
- ツールチェーン: ルネサス テクノロジ製 SuperH RISC engine Standard Toolchain Ver.9.3.0.0
- コンパイルオプション: High-performance Embedded Workshop でのデフォルト設定  
(-cpu=sh4a -include="\$(PROJDIR)¥inc"  
-object="\$(CONFIGDIR)¥\$(FILELEAF).obj" -debug -gbr=auto -chgincpath  
-errorpath -global\_volatile=0 -opt\_range=all -infinite\_loop=0  
-del\_vacant\_loop=0 -struct\_alloc=1 -nologo)

### 1.4 関連アプリケーションノート

本資料の参考プログラムは、「SH7763 グループ アプリケーションノート SH7763 初期設定例 (RJJ06B1050)」の設定条件で動作確認をしています。そちらも合わせてご参照ください。

また、以下のアプリケーションノートもご参照ください。

「SH7763 グループ アプリケーションノート イーサネット送信設定例 (RJJ06B1052)」

「SH7670 グループ アプリケーションノート イーサネット PHY-LSI 自動交渉設定例 (RJJ06B0887)」

## 2. 応用例の説明

本参考プログラムでは、ギガビットイーサネットコントローラ (GETHER) の MAC 層インタフェース 2 系統のうち 1 系統 (E-MAC-0) を使用してイーサネットの受信動作を行います。インタフェースは評価ボードの仕様に合わせて RMII を選択し、100BASE-T のイーサネット通信を行います。

受信の際は GETHER が内蔵する専用のダイレクトメモリアクセスコントローラ (E-DMAC) により、イーサネットフレームをメモリ上の受信バッファとの間で転送を行います。

### 2.1 使用機能の動作概要

GETHER は、以下の 3 つの機能から構成されています。

#### (1) DMA 転送制御部 (E-DMAC): メモリ上の送受信バッファと送受信 FIFO 間の DMA 転送

E-DMAC は E-DMAC に内蔵されているダイレクトメモリアクセス (DMA) 機能を使用し、ユーザが指定するイーサネットフレームデータの格納先 (アクセス可能なメモリ空間: 送信バッファ/受信バッファ) と送信/受 FIFO との間でフレームデータの転送を行います。

E-DMAC が DMA 転送を行うためには、ディスクリプタと呼ぶ送信/受信データの格納アドレス等が書かれた情報 (データ) が必要になります。E-DMAC は、ディスクリプタに書かれた情報に従って送信データを送信バッファから読み出す、あるいは受信データを受信バッファへ書き込みます。このディスクリプタは、読み書き可能なメモリ空間に配置されます。ディスクリプタを複数個並べ、ディスクリプタ列 (リスト) 化することによって、複数のイーサネットフレームデータの送信/受信を連続的に行うことができます。

E-DMAC は 0 系と 1 系の 2 系統あります。送信、受信とも、0 系と 1 系それぞれ独立して動作します。

#### (2) MAC 制御部 (E-MAC): 送信/受信 FIFO と GMII/MII/RMII 間の送信/受信処理

E-MAC は、送信 FIFO に書き込まれたデータからイーサネットフレームを構成し、GMII/MII/RMII へ送信します。また、GMII/MII/RMII から受信したイーサネットフレームの CRC チェックを行った後、分解し受信 FIFO に書き込みます。本 LSI の外部に接続する PHI-LSI とのインタフェースフォーマットとして、MII, GMII, RMII の 3 種類をサポートしています。

E-MAC は、0 系と 1 系の 2 系統あります。それぞれ E-DMAC の 0 系と 1 系と対応しています。

#### (3) 転送制御部 (TSU: Transfer Switching Unit): E-MAC の 1 系と 0 系間の転送処理および CAM 処理

TSU は、0 系の E-MAC と 1 系の E-MAC 間でイーサネットフレームデータの転送を行います。TSU は、E-DMAC と E-MAC の間に位置し、CAM エントリテーブルを参照しながら、E-MAC に入力されたイーサネットフレームの DA (宛先アドレス) を元に、以下のいずれかの処理を選択します。

1. 受信し受信 FIFO に書き込む
2. 転送のみ実行し転送 FIFO に書き込む
3. 受信し受信 FIFO に書き込むとともに転送 FIFO にも書き込む
4. 破棄する

TSU は、0 系から 1 系、1 系から 0 系への転送をそれぞれ別々に処理します。

表 1 に GETHER の概要を示します。

GETHER の機能の詳細については、「SH7763 グループ ハードウェアマニュアル (RJJ09B0260) 23 章 ギガビットイーサネットコントローラ (GETHER)」を参照ください。

表 1 GETHER の概要

項目	内容
E-MAC 機能	<ul style="list-style-type: none"> <li>• データフレームの組み立て/分解 (IEEE802.3,2000Edition 準拠フレーム形式)</li> <li>• 10Mbps , 100Mbps , および 1000Mbps 転送に対応</li> <li>• 全二重モードおよび半二重モード対応</li> <li>• 2 チャンネル内蔵 (GETHER0, GETHER1)</li> <li>• IEEE802.3x 準拠のフロー制御が可能</li> <li>• 3 種の IEEE802.3 準拠 PHY インタフェースをサポート                         <ul style="list-style-type: none"> <li>— GMII (Gigabit Media Independent Interface)</li> <li>— MII (Media Independent Interface)</li> <li>— RMII (Reduced Media Independent Interface)</li> </ul> </li> <li>• 上位プロトコルサポート (サムチェック) 機能</li> </ul>
E-DMAC 機能	<ul style="list-style-type: none"> <li>• GETHER と外部/内部メモリ間転送可能</li> <li>• 4 チャンネル内蔵</li> <li>• 32 バイトバースト転送可能</li> <li>• 1 フレーム/1 ディスクリプタ , 1 フレーム/複数ディスクリプタ方式 (マルチバッファ) 対応可能</li> <li>• 転送データ幅: 32 ビット</li> <li>• 送受信 FIFO 搭載 (送信用: 2K バイト , 受信用: 8K バイト)</li> </ul>
TSU 機能	<ul style="list-style-type: none"> <li>• チャンネル間転送用スイッチングユニット搭載 (転送用 FIFO: 6K バイト)</li> </ul>

## 2.2 使用機能の設定手順

ここでは、イーサネット受信するための基本的な設定例について説明します。図 1、図 2 にイーサネット受信設定フロー例を示します。

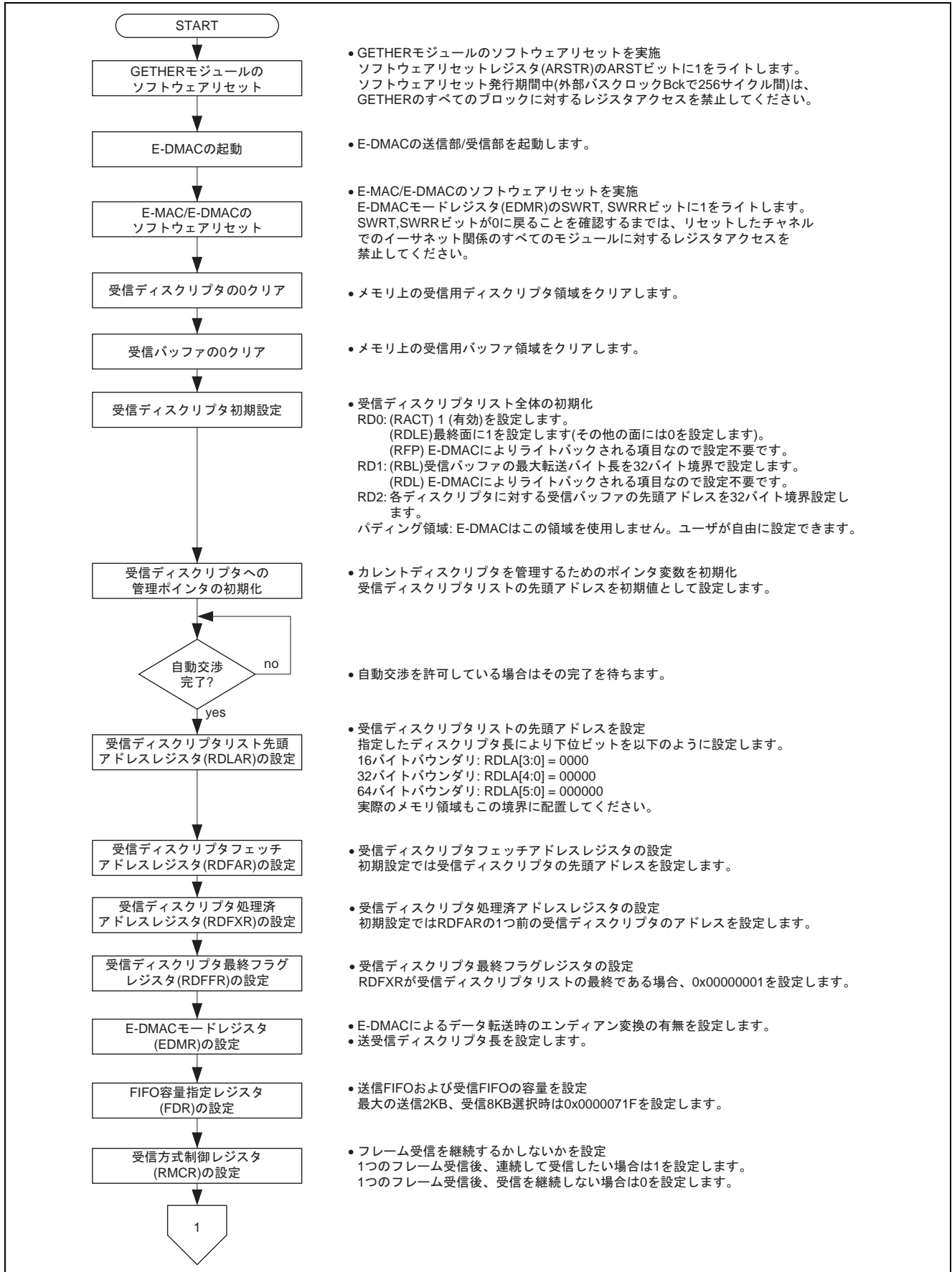


図1 イーサネット受信設定フロー例 (1)

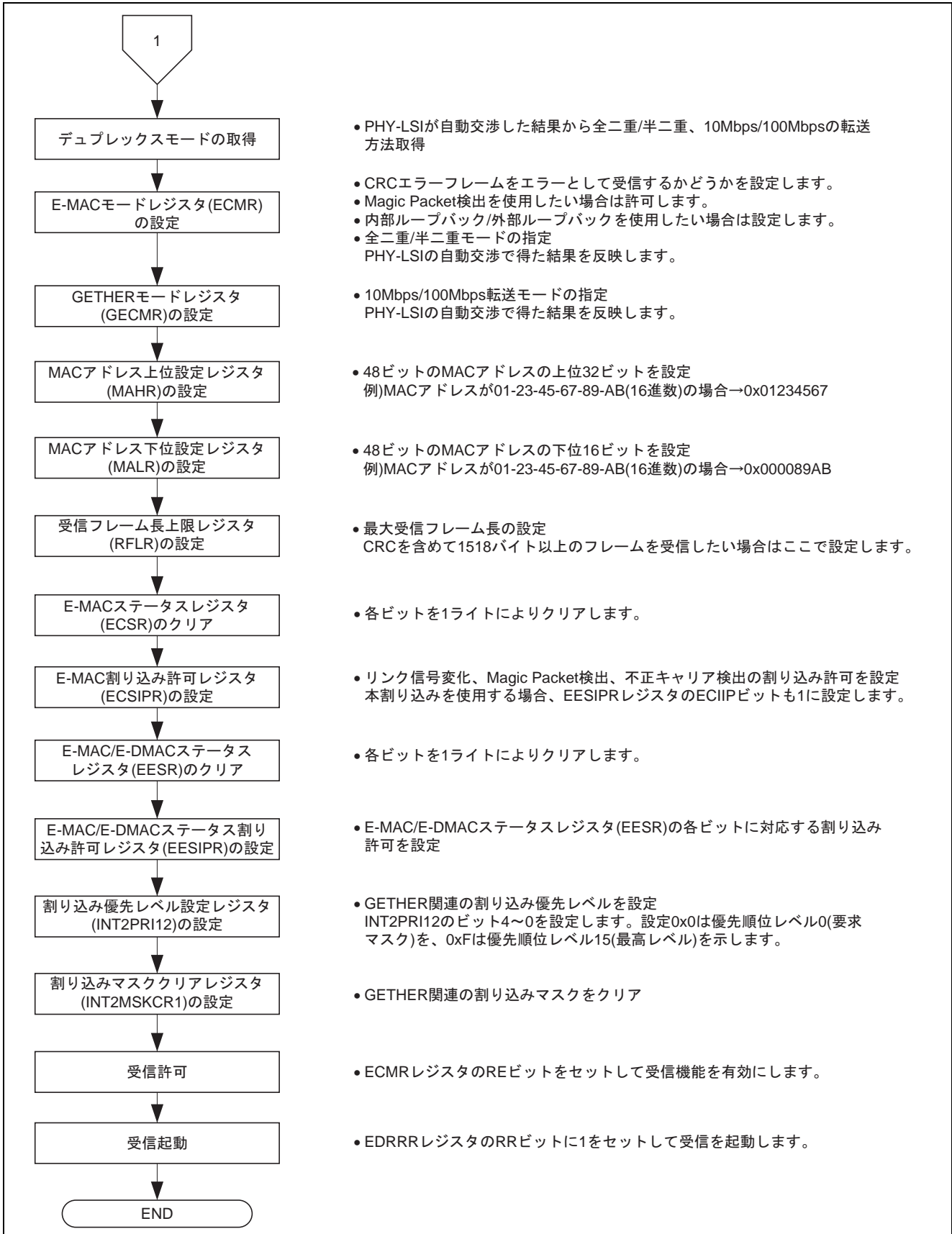


図2 イーサネット受信設定フロー例 (2)



### 2.3 参考プログラムの動作

参考プログラムでは、E-MAC-0 系および E-DMAC0 系を使用し、対向ホストからイーサネットフレームを 10 フレーム受信します。受信ディスクリプタと 1520 バイトの受信バッファを 4 面用意しています。受信方式制御レジスタ (RMCR) 内の受信コントロールビット (RNC) に 1 を設定し、連続受信方式にしています。フレーム受信割り込み (FR) 等受信に関連する割り込みが発生するたびに受信ディスクリプタの RFE ビット (RD0 のビット 27) をチェックし、エラーがなければ (RFE = 0 の場合) 受信バッファにある 1 フレーム分のデータをユーザバッファにコピーします。その後当該ディスクリプタを初期化し次回の受信に備えます。エラーがあれば (RFE = 1 の場合)、ユーザバッファへのコピーは行わず当該ディスクリプタを初期化するだけにします。

なお、受信バッファにはイーサネットフレームのうちプリアンブル、SFD、および CRC を除いた部分が転送されます。

図 3 に参考プログラムの動作環境を、図 4 にイーサネットフレームフォーマットを示します。

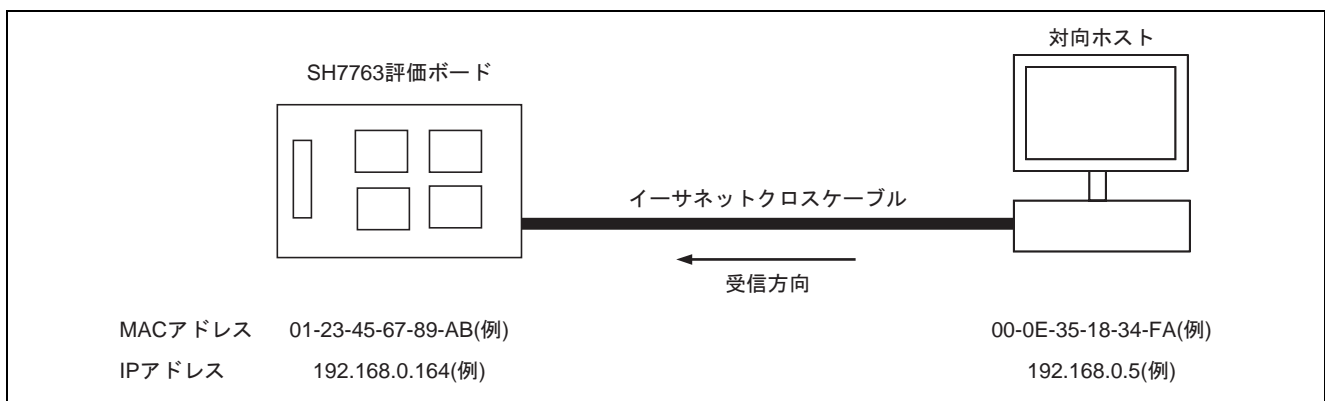


図 3 参考プログラムの動作環境

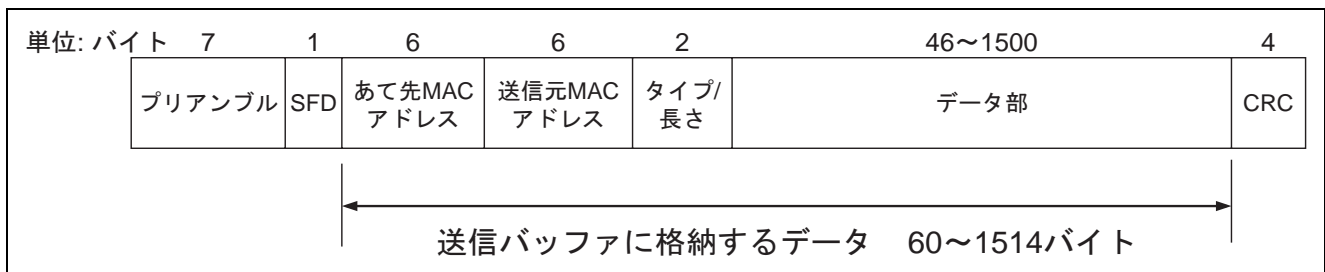


図 4 イーサネットフレームフォーマット

## 2.4 参考プログラムのディスクリプタ定義

E-DMAC ではディスクリプタのパディング領域を使用しません。ユーザが自由に使用できます。本プログラムではこの領域に次のディスクリプタの先頭アドレスを設定し、ソフトウェアにてリング構造を実現しています。図 5 に参考プログラムでの受信ディスクリプタ構造体の定義と受信ディスクリプタ列の使用例を示します。

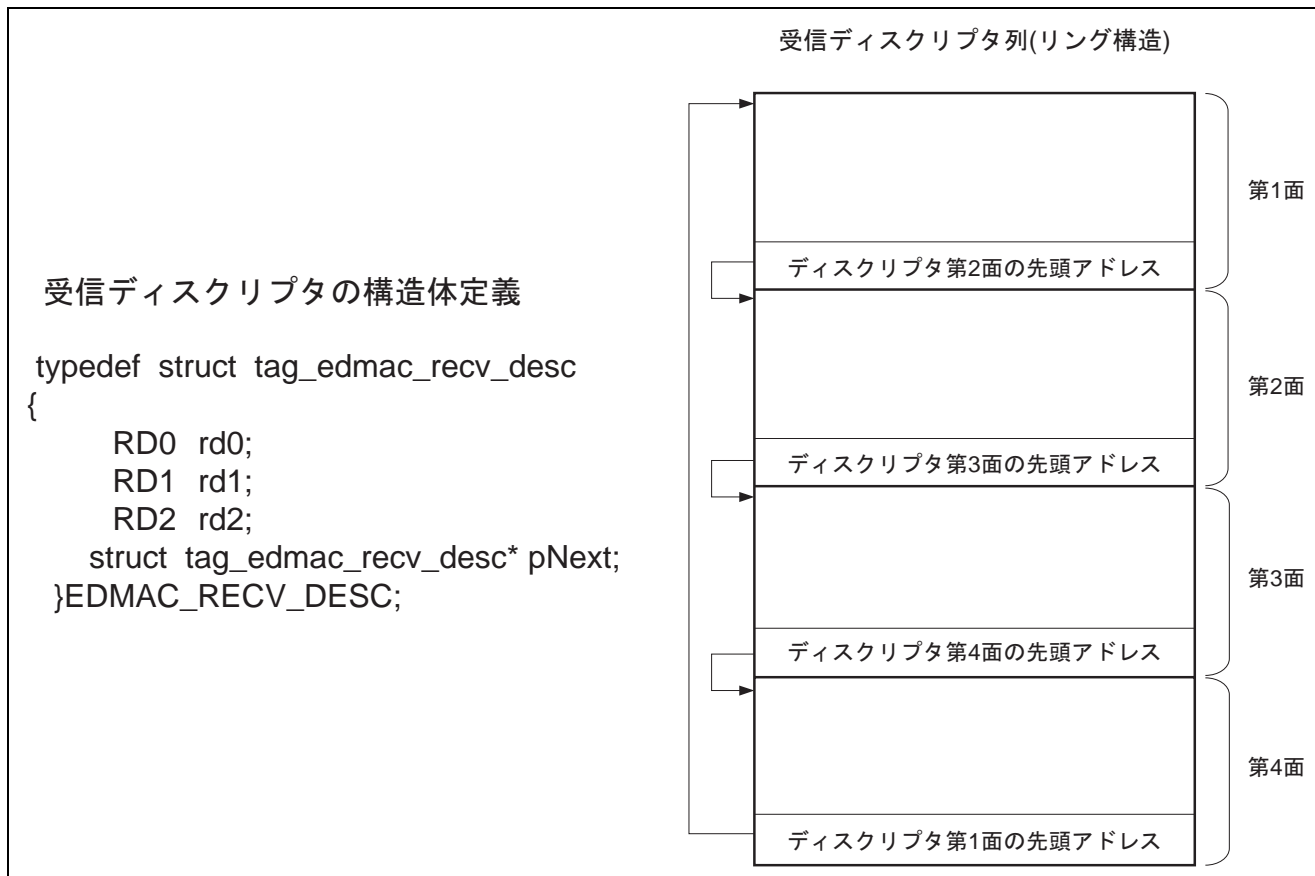


図 5 受信ディスクリプタの構造体定義と受信ディスクリプタ列使用例

2.5 参考プログラムの処理手順

図 6～図 10 に参考プログラムの処理フローを示します。なお、E-MAC/E-DMAC の各種レジスタおよびディスクリプタの初期設定では送信の設定も行っていますが、送信処理自体は行っていません。

main.c, ether.c 以外のプログラムは送信設定例と同様です。

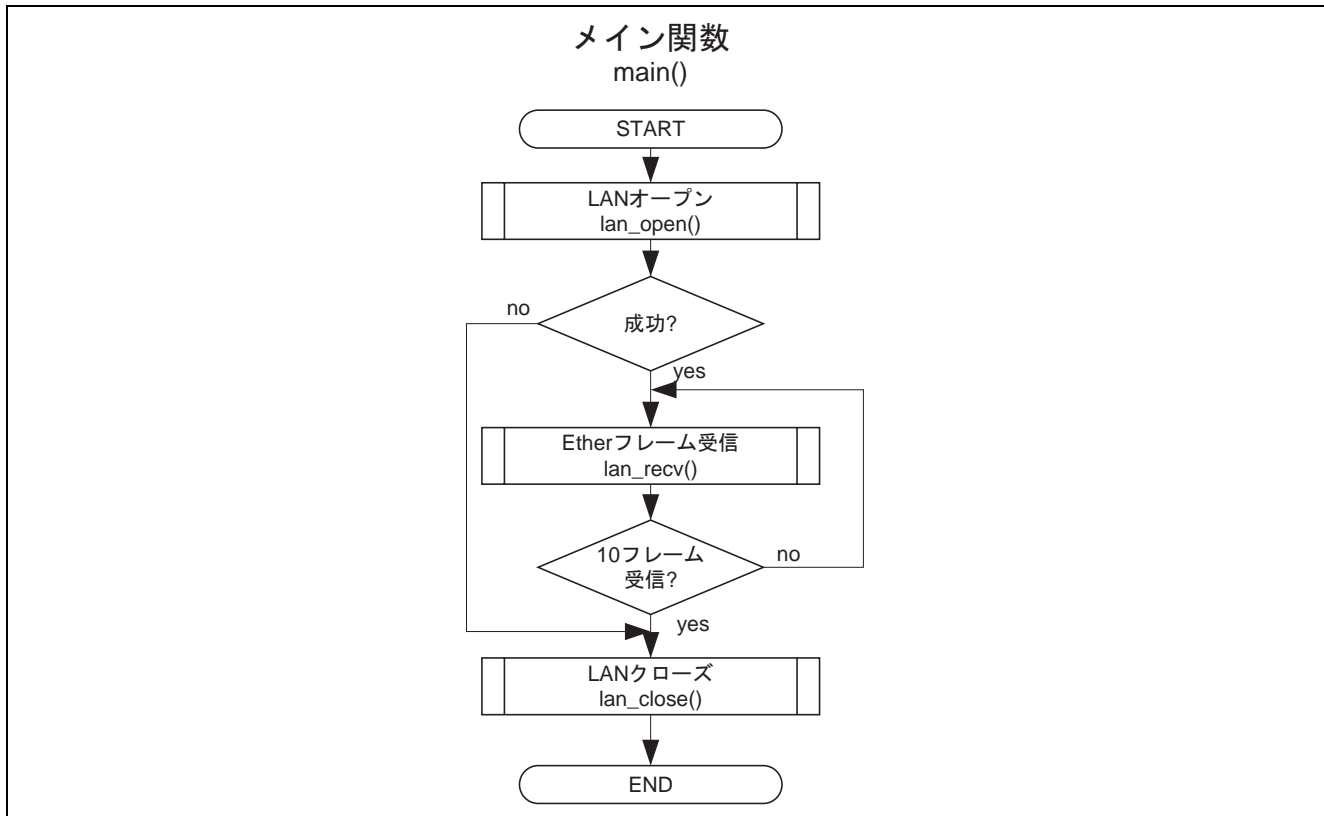


図 6 参考プログラムの処理フロー例 (1)

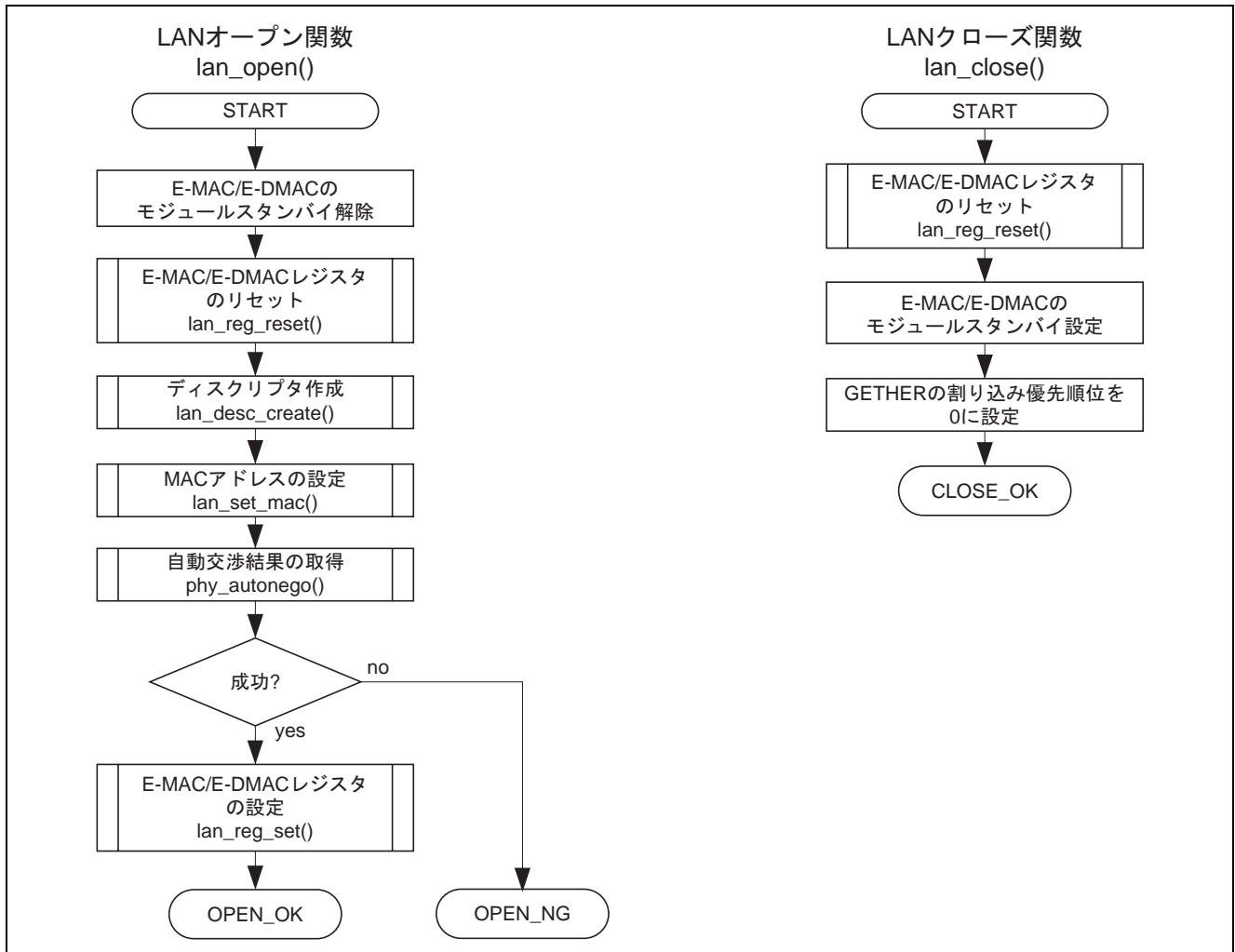


図7 参考プログラムの処理フロー例 (2)

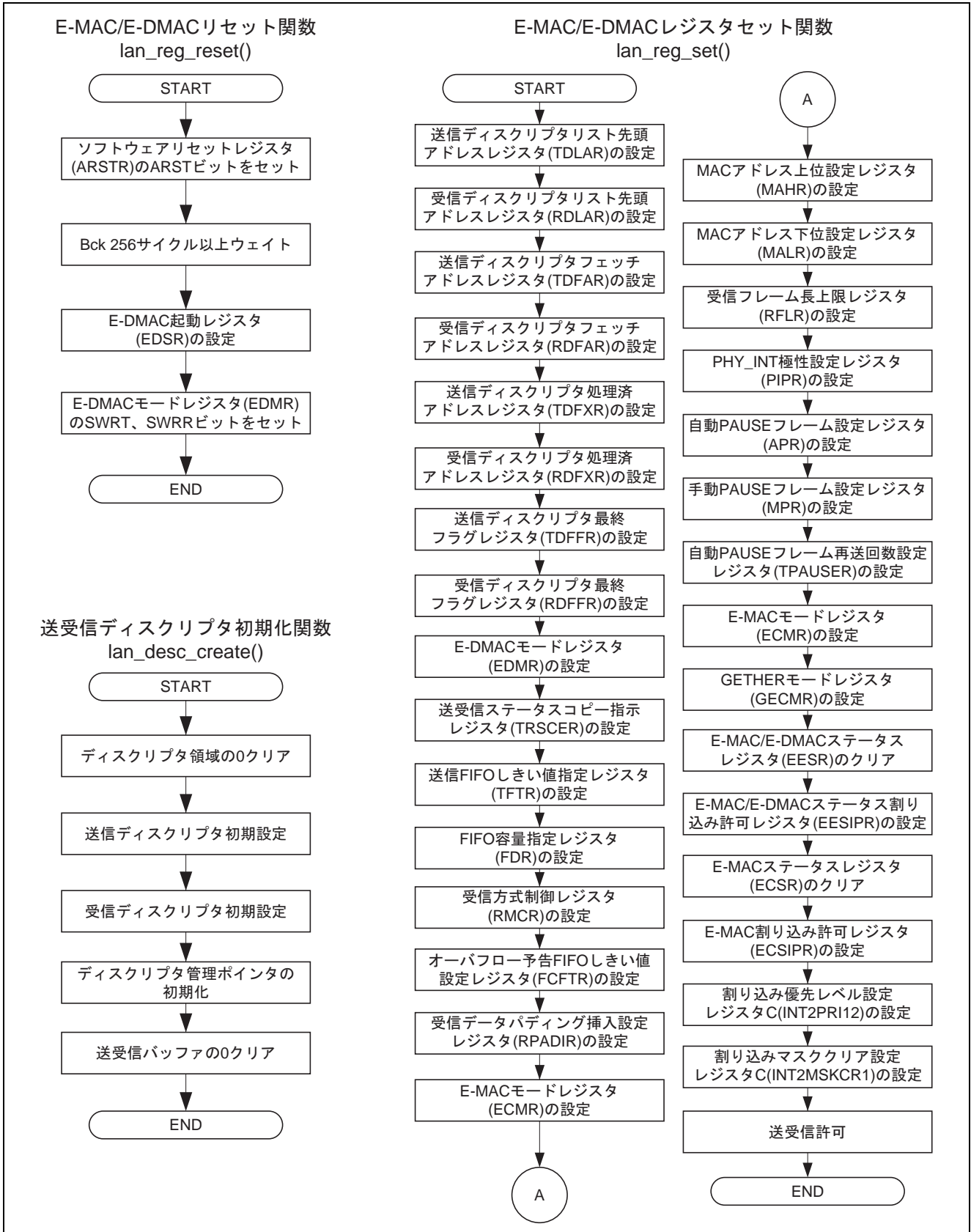


図8 参考プログラムの処理フロー例 (3)

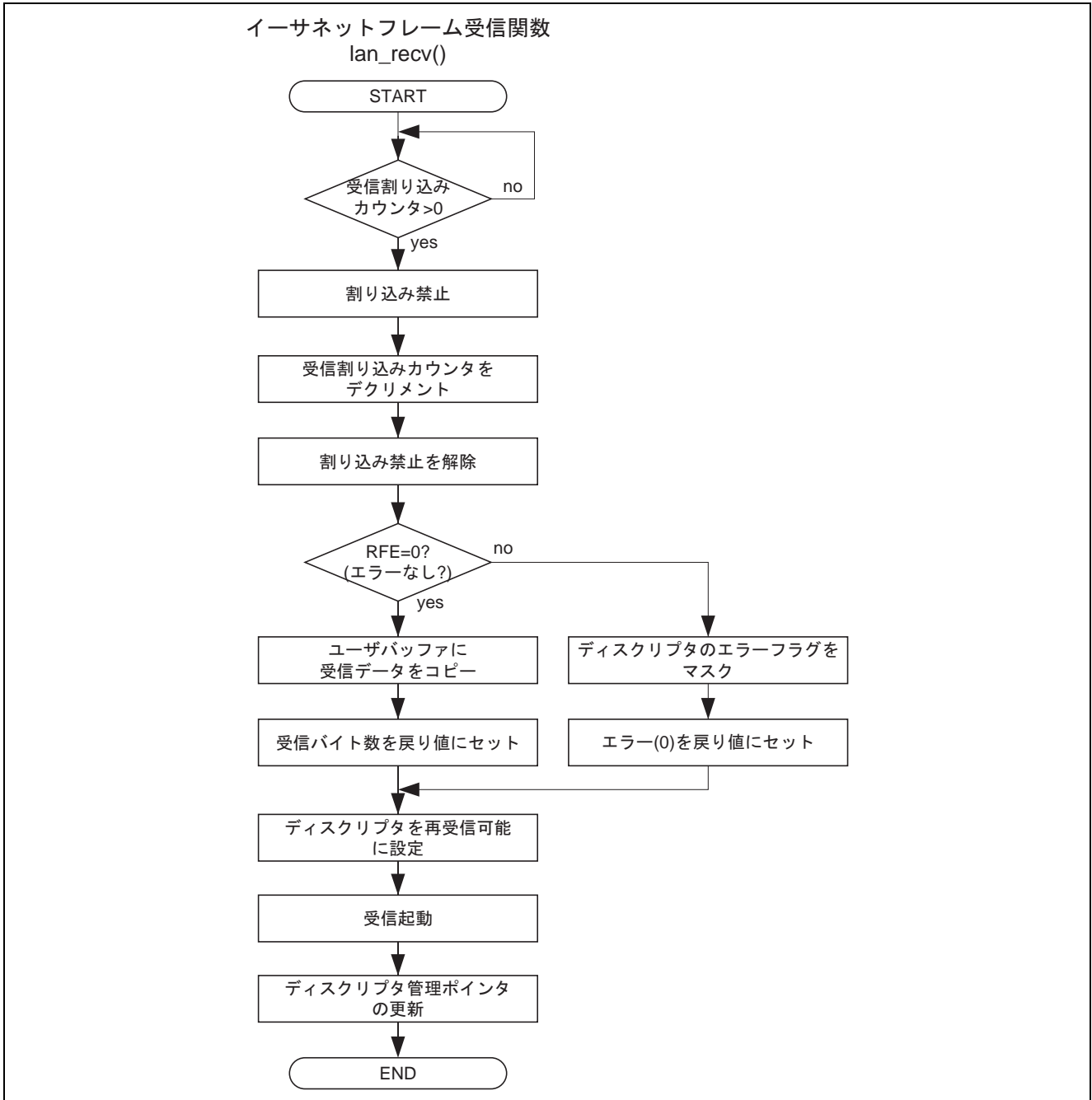


図9 参考プログラムの処理フロー例 (4)

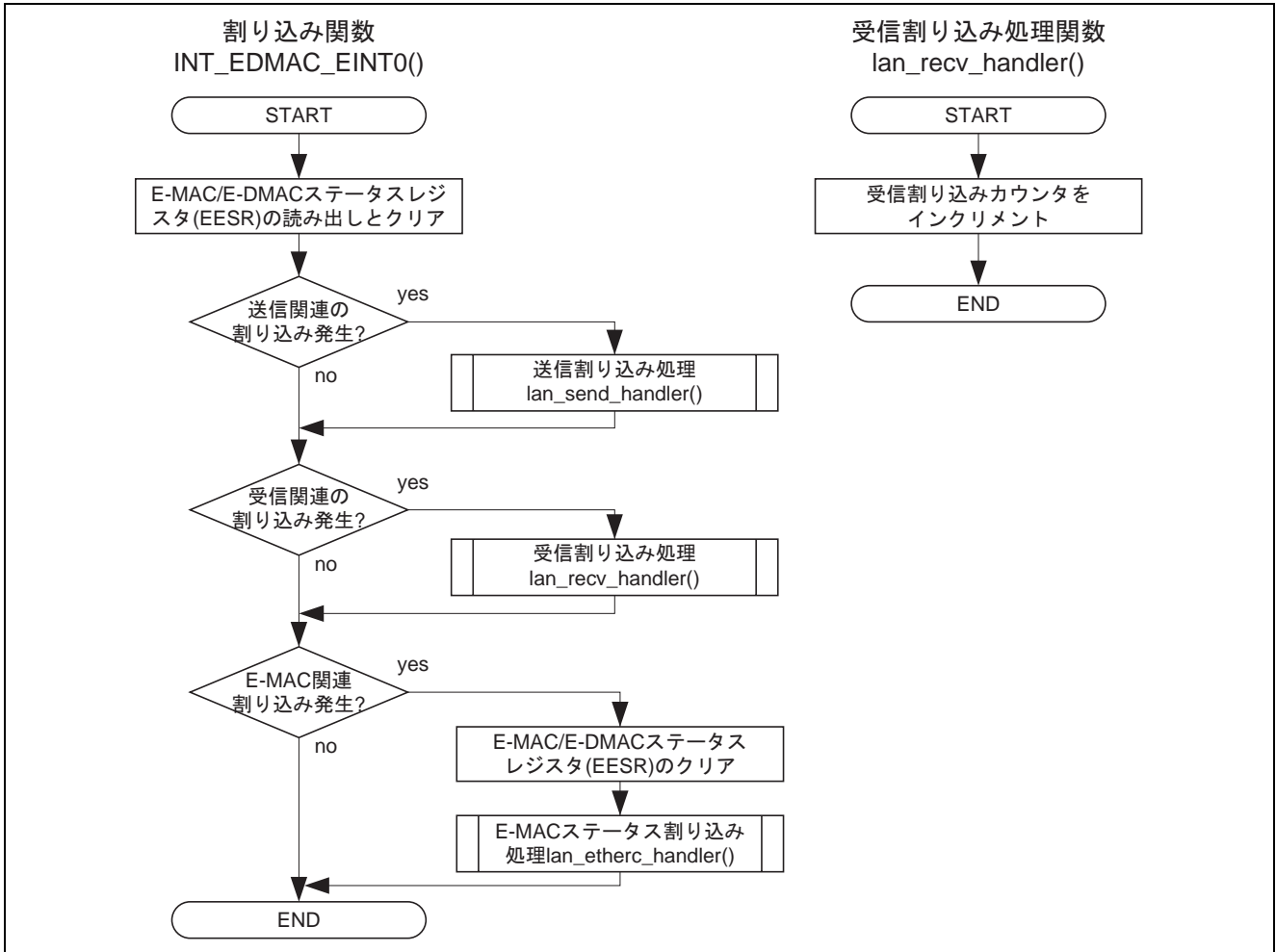


図 10 参考プログラムの処理フロー例 (5)

## 2.6 セクション配置

参考プログラムでは、送受信ディスクリプタとバッファの領域設定をセクション配置によって行っています。表 2 にセクション配置を示します。

表 2 セクション配置

セクション名	セクション用途	領域	配置アドレス (仮想アドレス)	
P	プログラム領域 (指定なしの場合)	ROM	0x00002000	P0 領域 (キャッシング可能, MMU アドレス変換可能)
C	定数領域	ROM		
C\$BSEC	未初期化データ領域用アドレス構造体	ROM		
C\$DSEC	初期化データ領域用アドレス構造体	ROM		
D	初期化データ (初期値)	ROM		
RINTTBL	初期化データ領域	RAM	0x08000000	
B	未初期化データ領域	RAM		
R	初期化データ領域	RAM		
RP	プログラム転送領域	RAM		
RC	定数転送領域	RAM		
S	スタック領域	RAM	0x0FFFF9F0	
INTHandler	例外/割り込みハンドラ	ROM	0x80000800	P1 領域 (キャッシング可能, MMU アドレス変換不可)
VECTTBL	リセットベクタテーブル 割り込みベクタテーブル	ROM		
INTTBL	割り込みマスクテーブル	ROM		
PIntPRG	割り込み関数	ROM		
SP_S	TLB ミスハンドラ専用スタック	RAM		
RSTHandler	リセットハンドラ	ROM	0xA0000000	P2 領域 (キャッシング不可, MMU アドレス変換不可)
PResetPRG	リセットプログラム	ROM		
DINTTBL	初期化データ領域	ROM		
PnonCACHE	プログラム領域 (キャッシュ無効アクセス)	ROM		
BETH_DESC	ディスクリプタ領域	RAM		
BETH_BUFF	バッファ領域	RAM	0xAF001000	

## 2.7 PHY-LSI 自動交渉設定

PHY-LSI との自動交渉結果の取得に関する参考プログラムの処理フローを図 11 ~ 図 15 に示します。

PHY 部インタフェースレジスタ (PIR) を経由して PHY-LSI 内の MII レジスタにアクセスして、物理層の自動交渉の結果を取得します。

本参考例では RMII インタフェースの PHY-LSI を使用していますが、基本的な考え方は MII インタフェースの PHY-LSI と同様です。

詳しくは、「SH7670 グループ アプリケーションノート イーサネット PHY-LSI 自動交渉設定例 (RJJ06B0887)」と PHY-LSI のデータシートをご参照ください。

ether.h, intprg.c, phy.c, phy.h のサンプルコードはイーサネット送信設定例と同一です。



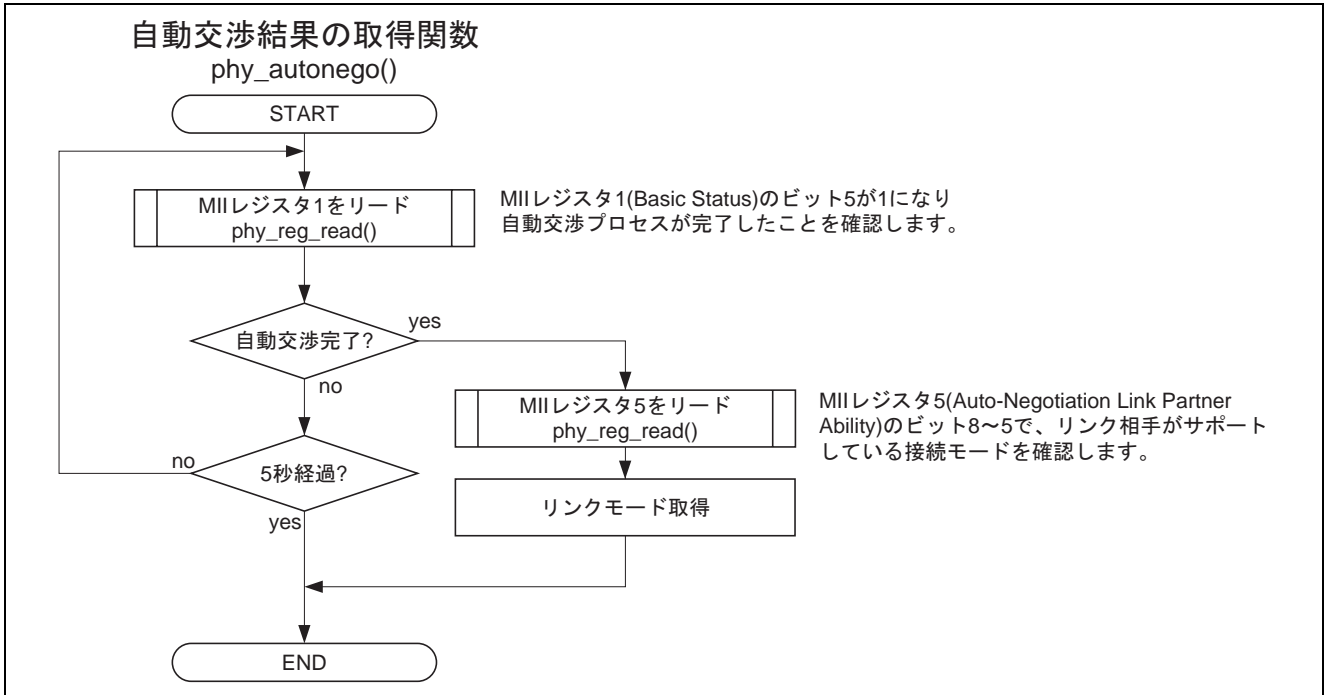


図 11 参考プログラムの処理フロー例 (6)

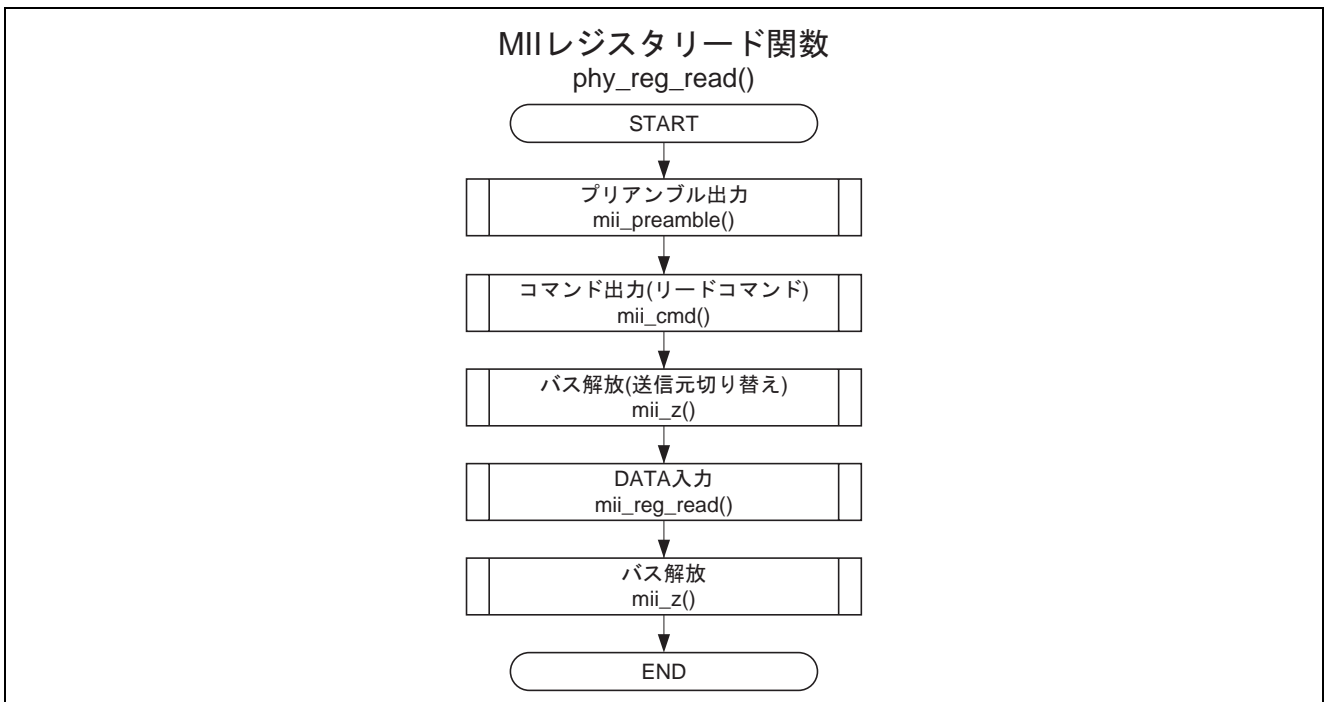


図 12 参考プログラムの処理フロー例 (7)

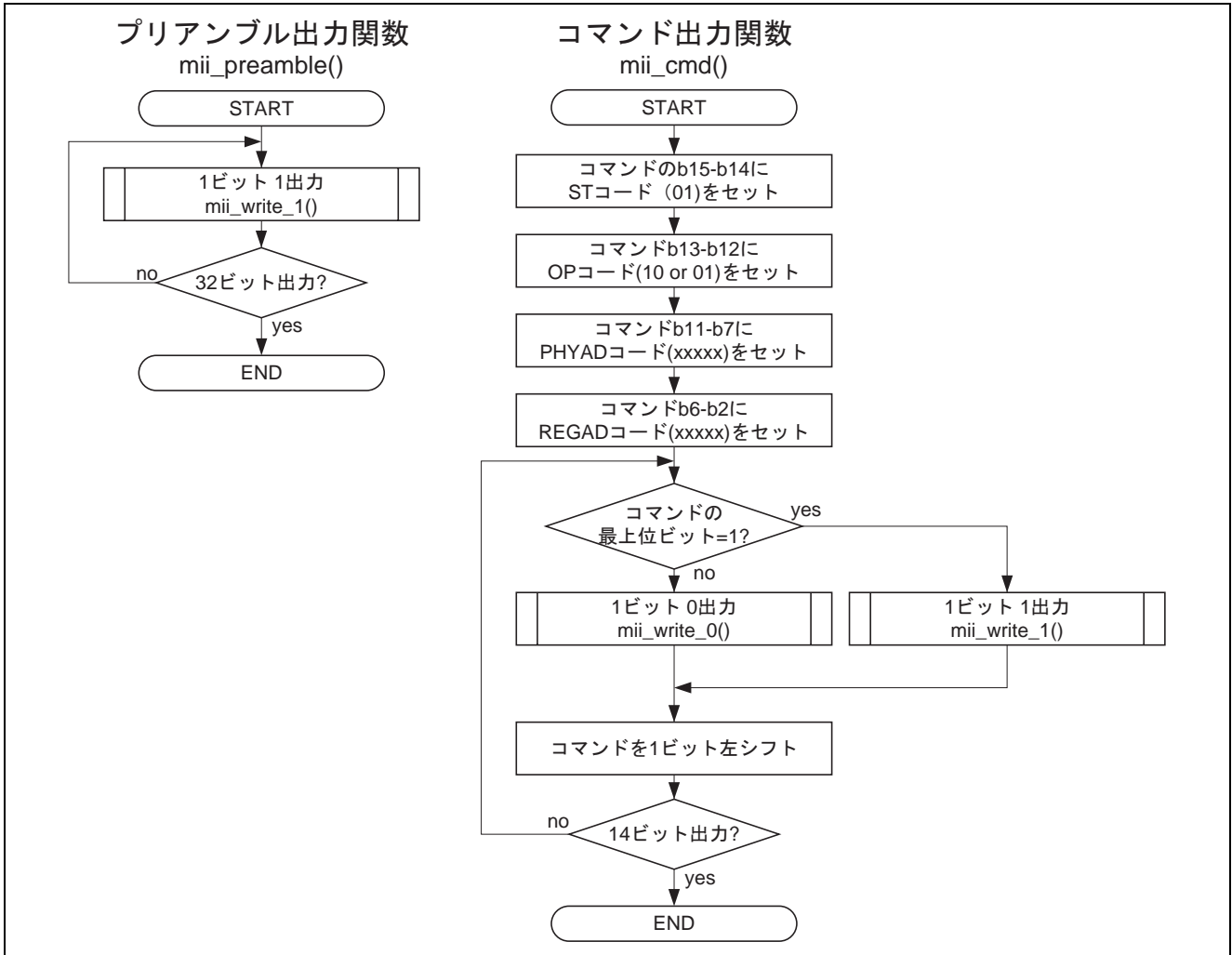


図 13 参考プログラムの処理フロー例 (8)

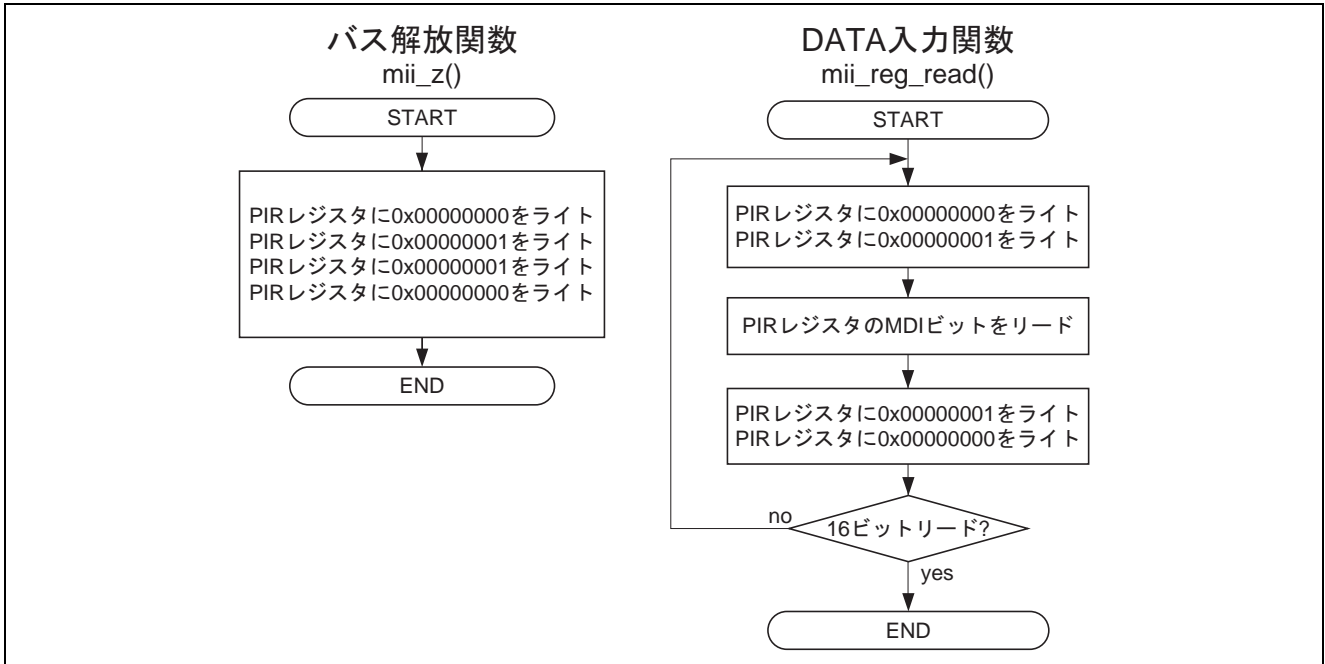


図 14 参考プログラムの処理フロー例 (9)

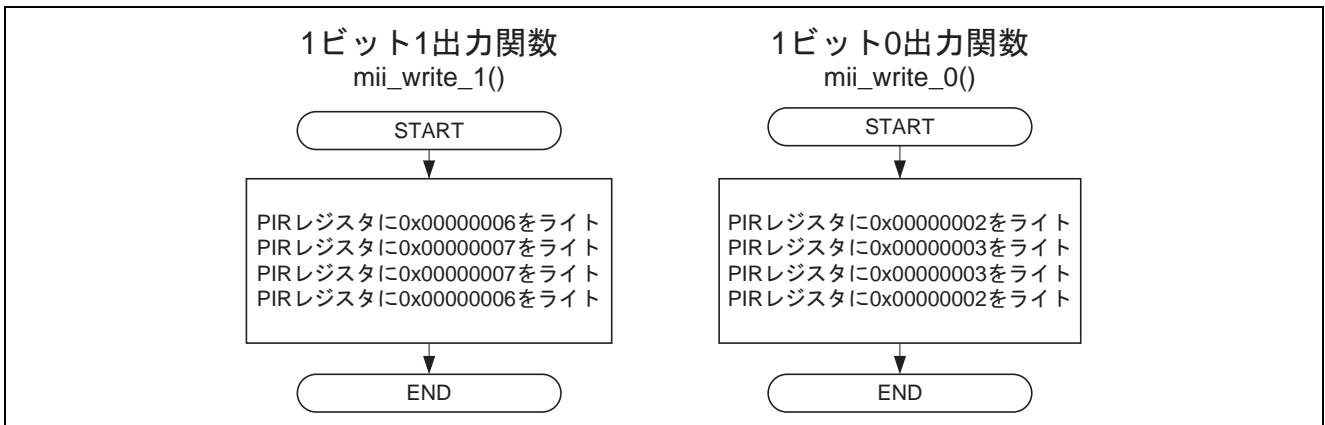


図 15 参考プログラムの処理フロー例 (10)

### 3. 参考プログラムリスト

#### 3.1 サンプルプログラムリスト "main.c"(1)

```

1  /*****
2  * DISCLAIMER
3
4  * This software is supplied by Renesas Technology Corp. and is only
5  * intended for use with Renesas products. No other uses are authorized.
6
7  * This software is owned by Renesas Technology Corp. and is protected under
8  * all applicable laws, including copyright laws.
9
10 * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 * DISCLAIMED.
15
16 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 * TECHNOLOGY CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21
22 * Renesas reserves the right, without notice, to make changes to this
23 * software and to discontinue the availability of this software.
24 * By using this software, you agree to the additional terms and
25 * conditions found by accessing the following link:
26 * http://www.renesas.com/disclaimer
27 *****/
28 /* Copyright (C) 2009. Renesas Technology Corp., All Rights Reserved. */
29 /** "FILE COMMENT" ***** Technical reference data *****
30 * System Name : SH7763 Sample Program
31 * File Name : main.c
32 * Abstract : Sample Program for Reception of Ethernet Frames
33 * Version : Ver 1.00
34 * Device : SH7763
35 * Tool-Chain : High-performance Embedded Workshop (Version 4.05.01.001)
36 * : C/C++ Compiler Package for SuperH Family (V.9.03 release00)
37 * OS : None
38 * H/W Platform : MS7763SE02
39 * Description : Sample Program for Reception of Ethernet Frames
40 * :
41 * Operation :
42 * Limitation :
43 * :
44 *****/
45 * History : 31.July.2009 Ver. 1.00 First Release
46 * "FILE COMMENT END" *****
47
48 #include "iodefine.h"
49 #include "ether.h"
50

```

### 3.2 サンプルプログラムリスト "main.c"(2)

```

51  /* **** プロトタイプ宣言 **** */
52  void main(void);
53
54  /* **** 変数宣言 **** */
55  #pragma section ETH_BUFFER          /* 大容量のため SDRAM に配置 */
56  typedef struct{
57      unsigned char frame[SIZE_OF_BUFFER];
58      int len;
59      unsigned char wk[12];
60  }USER_BUFFER;
61  static USER_BUFFER recv[10];
62  #pragma section
63
64  /*"FUNC COMMENT"*****
65  * ID          :
66  * Outline     : メイン関数
67  * Include     : #include "iodefine.h"
68  * Declaration : void main(void)
69  * Description : イーサネットフレームを受信します。
70  *
71  * Argument    : none
72  * Return Value : none
73  * Calling Functions :
74  *"FUNC COMMENT END"*****/
75  void main(void)
76  {
77      int i,j;
78      int ret;
79
80      /* ==== イーサネット初期設定 ==== */
81      ret = lan_open();
82      if( ret == OPEN_OK ){
83          /* ==== 10 フレーム受信開始 ==== */
84          for(i=0; i<10; i++){
85              /* ---- 受信 ---- */
86              recv[i].len = lan_recv( recv[i].frame );
87              if( recv[i].len == 0 ){
88                  i--;
89              }
90          }
91      }
92      /* ==== イーサネット送受信停止 ==== */
93      lan_close();
94  }
95  /* End of file */

```

### 3.3 サンプルプログラムリスト "ether.c"(1)

```

1  /*****
2  * DISCLAIMER
3
4  * This software is supplied by Renesas Technology Corp. and is only
5  * intended for use with Renesas products. No other uses are authorized.
6
7  * This software is owned by Renesas Technology Corp. and is protected under
8  * all applicable laws, including copyright laws.
9
10 * THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 * REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 * INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 * PARTICULAR PURPOSE AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY
14 * DISCLAIMED.
15
16 * TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 * TECHNOLOGY CORP. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 * FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 * FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 * AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21
22 * Renesas reserves the right, without notice, to make changes to this
23 * software and to discontinue the availability of this software.
24 * By using this software, you agree to the additional terms and
25 * conditions found by accessing the following link:
26 * http://www.renesas.com/disclaimer
27 *****/
28 /* Copyright (C) 2009. Renesas Technology Corp., All Rights Reserved. */
29 /*"FILE COMMENT"***** Technical reference data *****/
30 * System Name : SH7763 Sample Program
31 * File Name : ether.c
32 * Abstract : Sample Program for Reception of Ethernet Frames
33 * Version : Ver 1.00
34 * Device : SH7763
35 * Tool-Chain : High-performance Embedded Workshop (Version 4.05.01.001)
36 * : C/C++ Compiler Package for SuperH Family (V.9.03 release00)
37 * OS : None
38 * H/W Platform : MS7763SE02
39 * Description : Sample Program for Reception of Ethernet Frames
40 * :
41 * Operation :
42 * Limitation :
43 * :
44 *****/
45 * History : 31.July.2009 Ver. 1.00 First Release
46 /*"FILE COMMENT END"*****/
47
48 #include "machine.h"
49 #include "string.h"
50 #include "iodefine.h"

```

### 3.4 サンプルプログラムリスト "ether.c"(2)

```

51  #include "phy.h"
52  #include "ether.h"
53
54  /* **** マクロ宣言 **** */
55  #define DEFAULT_MAC_H    0x00010203          /* デバッグ用 */
56  #define DEFAULT_MAC_L    0x00000405
57  #define MACSET_OK        0
58  #define MACSET_NG        -1
59  #define FPGA_ETON        (*(volatile unsigned short *)0xBB00001A) /* FPGA Ether ON address */
60
61  /* **** プロトタイプ宣言 **** */
62  void main(void);
63  void lan_send_handler( unsigned long status );
64  static void lan_desc_create( void );
65  static void lan_reg_reset( void );
66  static void lan_reg_set( int link );
67
68  /* **** 変数宣言 **** */
69  /* ---- ディスクリプタ ---- */
70  #pragma section ETH_DESC          /* 16 バイト境界へ配置 */
71  static volatile TXRX_DESCRIPTOR_SET desc; /* ディスクリプタ領域 */
72  #pragma section
73  /* ---- バッファ ---- */
74  #pragma section ETH_BUFFER        /* 16 バイト境界へ配置 */
75  static volatile TXRX_BUFFER_SET buf; /* 送受信バッファ領域 */
76  #pragma section
77  /* ---- MAC アドレス ---- */
78  static unsigned long my_macaddr_h;
79  static unsigned long my_macaddr_l;
80  /* ---- その他 ---- */
81  static volatile int c_recv = 0; /* 受信フレームカウンタ */
82
83  /*"FUNC COMMENT"*****
84  * ID          :
85  * Outline     : イーサネットオープン関数
86  * Include     : #include "iodefine.h"
87  *             : #include "phy.h"
88  *             : #include "ether.h"
89  * Declaration : int lan_open(void)
90  * Description : E-DMAC, E-MAC, PHY, バッファメモリの初期化
91  *             : を行います。
92  *             :
93  * Argument    : none
94  * Return Value : OPEN_OK(0) :オープン成功
95  *             : OPEN_NG(-1):オープン失敗
96  * Calling Functions :
97  *"FUNC COMMENT END"*****/
98  int lan_open(void)
99  {
100     int link;

```

### 3.5 サンプルプログラムリスト "ether.c"(3)

```

101
102     /* ==== PFC の設定 ==== */
103     GPIO.PSEL2.BIT.PTSEL23 = 0;
104     GPIO.PSEL3.BIT.PTSEL33 = 0;
105     GPIO.PSEL4.BIT.PTSEL4A = 3;
106     GPIO.PSEL4.BIT.PTSEL43 = 2;
107     GPIO.PJCR.WORD = 0xFFFC;
108     GPIO.PMCR.WORD = 0x0000;
109     GPIO.POCR.WORD = 0x0000;
110     /* ==== FPGA の設定 ==== */
111     FPGA_ETON = 0x0001;                /* FPGA:Ether ポート使用可能 */
112     /* ==== E-MAC,E-DMAC の停止 === */
113     lan_reg_reset();
114     /* ==== バッファメモリの初期化 ==== */
115     lan_desc_create();
116     /* ==== E-MAC,E-DMAC の設定 ==== */
117     link = phy_autonego();             /* デュプレックスモード確認 */
118     if( link == NEGO_FAIL ){
119         return OPEN_NG;                /* OPEN 失敗 */
120     }
121     else{
122         lan_reg_set(link);
123     }
124     return OPEN_OK;
125 }
126
127 /*"FUNC COMMENT"*****
128 * ID          :
129 * Outline     : イーサネットクローズ関数
130 * Include     : #include "iodefine.h"
131 *             : #include "ether.h"
132 * Declaration : int lan_close(void)
133 * Description : EDMAC/E-MAC を停止する。
134 *             :
135 * Argument    : none
136 * Return Value : none
137 * Calling Functions :
138 *"FUNC COMMENT END"*****/
139 int lan_close( void )
140 {
141     int i;
142
143     /* ==== E-MAC,E-DMAC のリセット === */
144     lan_reg_reset();
145     /* ==== E-DMAC 関連の割り込みを禁止 === */
146     INTC.INT2PRI12.BIT._GETHER = 0;
147
148     return CLOSE_OK;
149 }
150

```



### 3.6 サンプルプログラムリスト "ether.c"(4)

```

151  /*"FUNC COMMENT"*****
152  * ID          :
153  * Outline     : フレーム受信関数
154  * Include     : #include "iodefine.h"
155  *             : #include "ether.h"
156  * Declaration : int lan_rcv( unsigned char *addr )
157  * Description : 受信したフレームを指定されたバッファにコピーします。
158  *             : 受信フレームが存在しなければ本関数内で待ちます。
159  *             :
160  * Argument    : unsigned char *addr
161  * Return Value: int : 受信したフレームのバイト数 (エラー:0)
162  * Calling Functions :
163  *"FUNC COMMENT END"*****/
164  int lan_rcv( unsigned char *addr )
165  {
166      int i;
167      int pri;
168      int ret = 0;
169      EDMAC_RECV_DESC *p;
170
171      /* ==== 受信待ち ==== */
172      while( c_rcv <= 0 ){
173          /* wait */
174      }
175      /* ==== 割り込みカウンタのデクリメント ==== */
176      pri = INTC.INT2PRI12.BIT._GETHER;          /* 排他制御 (割り込み禁止) */
177      INTC.INT2PRI12.BIT._GETHER = 0;
178      --c_rcv;
179      INTC.INT2PRI12.BIT._GETHER = pri;
180
181      /* ==== 受信フレームをコピー ==== */
182      p = desc.pRecv_end;
183      if( p->rd0.BIT.RFE == 0 ){
184          memcpy(addr, p->rd2.RBA, p->rd1.RDL);
185          ret = p->rd1.RDL;
186      }
187      /* ---- 受信エラー ---- */
188      else{
189          p->rd0.LONG &= 0x70000000;          /* エラーフラグを処理 */
190          ret = 0;                          /* 受信エラー:0 */
191      }
192      /* ==== ディスクリプタを受信できる状態に戻す ==== */
193      p->rd0.BIT.RACT = 1;
194      /* ---- 受信起動 ---- */
195      if( EDMAC0.EDRRR.BIT.RR == 0 ){          /* 1 ライト前に必ず 0 をリード */
196          EDMAC0.EDRRR.BIT.RR = 1;
197      }
198      /* ==== カレントポインタの更新 ==== */
199      desc.pRecv_end = p->pNext;
200

```

### 3.7 サンプルプログラムリスト "ether.c"(5)

```

201     return ret;
202 }
203
204 /*"FUNC COMMENT"*****
205 * ID
206 * Outline      : ディスクリプタ構築関数
207 * Include      : #include "ether.h"
208 * Declaration  : static void lan_desc_create( void )
209 * Description  : イーサネットで必要な送受信バッファとディスクリプタの初期化を
210 *              : 行います。1 フレーム/1 バッファを前提としています。
211 *
212 * Argument     : none
213 * Return Value : none
214 * Calling Functions
215 *"FUNC COMMENT END"*****
216 static void lan_desc_create( void )
217 {
218     int i;
219     /* ==== ディスクリプタ領域の構築 ==== */
220     /* ---- メモリクリア ---- */
221     memset(&desc, 0, sizeof(desc) );
222     /* ---- 送信ディスクリプタ ---- */
223     for(i=0; i<NUM_OF_TX_DESCRIPTOR; i++){
224         desc.send[i].td2.TBA = buf.send[i];      /* TD2 */
225         desc.send[i].td1.TDL = 0;                /* TD1 */
226         desc.send[i].td0.LONG= 0x30000000;      /* TD0:1frame/1buf、送信無効*/
227         if( i != (NUM_OF_TX_DESCRIPTOR-1) ){ /* pNext */
228             desc.send[i].pNext = &desc.send[i+1];
229         }
230     }
231     desc.send[i-1].td0.BIT.TDLE = 1;
232     desc.send[i-1].pNext = &desc.send[0];
233     /* ---- 受信ディスクリプタ ---- */
234     for(i=0; i<NUM_OF_RX_DESCRIPTOR; i++){
235         desc.recv[i].rd2.RBA = buf.recv[i];      /* RD2 */
236         desc.recv[i].rd1.RBL = SIZE_OF_BUFFER;   /* RD1 */
237         desc.recv[i].rd0.LONG= 0xb0000000;      /* RD0:1frame/1buf、受信有効*/
238         if( i != (NUM_OF_RX_DESCRIPTOR-1) ){ /* pNext */
239             desc.recv[i].pNext = &desc.recv[i+1];
240         }
241     }
242     desc.recv[i-1].rd0.BIT.RDLE = 1;              /* 最終ディスクリプタの設定 */
243     desc.recv[i-1].pNext = &desc.recv[0];
244
245     /* ---- ディスクリプタ管理情報の初期化 ---- */
246     desc.pSend_top = &desc.send[0];
247     desc.pRecv_end = &desc.recv[0];
248
249     /* ==== バッファ領域の構築 ==== */
250     /* ---- 領域クリア ---- */
    
```

### 3.8 サンプルプログラムリスト "ether.c"(6)

```

251     memset(&buf, 0, sizeof(buf) );
252 }
253
254 /*"FUNC COMMENT"*****
255 * ID
256 * Outline      : E-MAC,E-DMAC レジスタ初期化関数
257 * Include      : #include "iodefine.h"
258 * Declaration  : static void lan_reg_reset( void )
259 * Description  : E-MAC と E-DMAC のレジスタをリセットします。
260 *
261 * Argument     : none
262 * Return Value : none
263 * Calling Functions
264 *"FUNC COMMENT END"*****/
265 static void lan_reg_reset( void )
266 {
267     volatile int j = 200;          /* B 256 サイクルウェイト */
268
269     /* ---- GETHER ソフトウェアリセット ---- */
270     GETHER.ARSTR.BIT.ARST = 1;    /* E-DMAC ソフトウェアリセット */
271     while(j--){
272         /* B 256 サイクル ウェイト */
273     }
274
275     /* ---- E-DMAC ソフトウェアリセット ---- */
276     EDMAC0.EDSR = 0x00000003;     /* E-DMAC 起動 */
277     EDMAC0.EDMR.LONG = 0x00000003; /* E-DMAC ソフトウェアリセット */
278
279     /* ---- ソフトウェアリセット解除の確認 ---- */
280     while(EDMAC0.EDMR.LONG != 0x00000000){
281         nop();
282         nop();
283     }
284 }
285
286 /*"FUNC COMMENT"*****
287 * ID
288 * Outline      : EhterC,E-DMAC レジスタ設定
289 * Include      : #include "iodefine.h"
290 *
291 *              : #include "ether.h"
292 *              : #include "PHY.h"
293 * Declaration  : void lan_reg_set(int link)
294 * Description  : E-DMAC, E-MAC の初期化を行います。
295 *
296 * Argument     : int link
297 * Return Value : none
298 * Calling Functions
299 *"FUNC COMMENT END"*****/
300 static void lan_reg_set( int link )
301 {

```

## 3.9 サンプルプログラムリスト "ether.c"(7)

```

301     /* ==== EDMAC ==== */
302     EDMAC0.TDLAR    = &desc.send[0];/* 送信ディスクリプタ先頭          */
303     EDMAC0.RDLAR    = &desc.recv[0];/* 受信ディスクリプタ先頭          */
304     EDMAC0.TDFAR    = &desc.send[0];/* 送信ディスクリプタフェッチアドレスレジスタ*/
305     EDMAC0.RDFAR    = &desc.recv[0];/* 受信ディスクリプタフェッチアドレスレジスタ*/
306     EDMAC0.TDFXR    = &desc.send[3];/* 送信ディスクリプタ処理アドレスレジスタ*/
307     EDMAC0.RDFXR    = &desc.recv[3];/* 受信ディスクリプタ処理アドレスレジスタ*/
308     EDMAC0.TDFFR    = 0x00000001; /* 送信ディスクリプタ最終フラグレジスタ */
309     EDMAC0.RDFFR    = 0x00000001; /* 受信ディスクリプタ最終フラグレジスタ */
310     EDMAC0.EDMR.LONG = 0x00000000; /* エンディアン変換なし(ビッグエンディアン) */
311                                     /* ディスクリプタ長 16 バイト          */
312     EDMAC0.TRSCER.LONG = 0x00000000; /* 全ステータスをディスクリプタにコピー */
313     EDMAC0.TFTR.LONG = 0x00000000; /* 送信 FIFO しきい値(ストア&フォワード) */
314     EDMAC0.FDR.BIT.TFD = 0x07;      /* 送信 FIFO 容量(2048 バイト)          */
315     EDMAC0.FDR.BIT.RFD = 0x1F;      /* 受信 FIFO 容量(8192 バイト)          */
316     EDMAC0.RMCR.BIT.RNC = 1;        /* 連続受信可能                          */
317     EDMAC0.FCFTR.LONG = 0x00170007; /* フロー制御しきい値設定 E-MAC で禁止 */
318     EDMAC0.RPADIR.LONG = 0x00000000; /* パディング挿入なし                    */
319     /* ==== E-MAC ==== */
320     MAC0.ECMR.LONG   = 0x00000000; /* カウンタクリアモード                  */
321                                     /* サムチェック計算をしない              */
322                                     /* ショートフレームにパディング付加      */
323                                     /* OTIMEPAUSE フレーム受信無効          */
324                                     /* PAUSE フレームを転送しない            */
325                                     /* キャリア消失エラーをチェック        */
326                                     /* PAUSE フレームを転送しない            */
327                                     /* フロー制御無効                        */
328                                     /* CAM エントリ以外のマルチキャストフレームを受信 */
329                                     /* Magic Packet 検出を許可しない        */
330                                     /* 受信禁止                              */
331                                     /* 送信禁止                              */
332                                     /* 内部ループバックを行わない            */
333                                     /* 外部ループバックを行わない            */
334                                     /* デュプレックスモード(半二重)          */
335                                     /* プロミスキャスモードを行わない        */
336     MAC0.MAHR        = DEFAULT_MAC_H; /* MAC アドレスの設定                    */
337     MAC0.MALR        = DEFAULT_MAC_L;
338     MAC0.RFLR.LONG   = 0x00000;      /* 最大受信フレーム長(1518 バイト)      */
339     MAC0.PIPR.BIT.PHYIP = 0;        /* ET_PHY-INT はローアクティブ          */
340     MAC0.APR.BIT.AP   = 0x0000;     /* 自動 PAUSE フレーム TIME パラメータ: フロー制御無効 */
341     MAC0.MPR.BIT.MP   = 0x0000;     /* 手動 PAUSE フレーム TIME パラメータ: フロー制御無効 */
342     MAC0.TPAUSER      = 0x00000000; /* 自動 PAUSE フレーム再送回数無制限    */
343     if( link == FULL_TX || link == FULL_10M ){
344         MAC0.ECMR.BIT.DM = 1;        /* 全二重モードに設定                    */
345     }
346     if( link == FULL_TX || link == HALF_TX ){
347         MAC0.GECMR.LONG = 0x00000004; /* 100Mbps に設定                        */
348     }
349     else{
350         MAC0.GECMR.LONG = 0x00000000; /* 10Mbps に設定                          */
    
```

## 3.10 サンプルプログラムリスト "ether.c"(8)

```

351     }
352     MAC0.BCULR.BIT.BSTLMT=  0x000;      /* バースト回数上限 256 サイクル          */
353     /* ==== 割り込み関連 ==== */
354     EDMAC0.EESR          = 0xFF3F07FF; /* 全ステータスクリア(1 ライトでクリア) */
355     EDMAC0.EESIPR.LONG   = EDMAC_EESIPR_INI_SEND | EDMAC_EESIPR_INI_RECV | EDMAC_EESIPR_INI_EtherC;
356                               /* 送受信 + E-MAC の割り込みを許可          */
357     MAC0.ECSR.LONG      = 0x00000017; /* 全ステータスクリア(1 ライトでクリア) */
358     MAC0.ECSIPR.LONG    = EtherC_ECSIPR_INI; /* 割り込み許可                          */
359     INTC.INT2PRI12.BIT._GETHER = 5;      /* E-DMAC(EINT0)割り込み優先順位 5      */
360     INTC.INT2MSKCR1.BIT._GETHER = 1;     /* GETHER 割り込みマスククリア          */
361
362     /* ==== 送受信の許可 ==== */
363     /* ==== E-MAC の許可 ==== */
364     MAC0.ECMR.BIT.RE = 1;                /* 受信機能有効                          */
365     MAC0.ECMR.BIT.TE = 1;                /* 送信機能有効                          */
366     /* ==== E-DMAC の許可 ==== */
367     if(EDMAC0.EDRRR.BIT.RR == 0){
368         EDMAC0.EDRRR.BIT.RR = 1;        /* 受信起動                              */
369     }
370 }
371
372 /*"FUNC COMMENT"*****
373 * ID          :
374 * Outline     : 送信割り込み関数
375 * Include     : #include "iodefine.h"
376 *             : #include "ether.h"
377 * Declaration : void lan_send_handler( unsigned long status )
378 * Description  : EDMAC (EESR)に関する送信関連の割り込みハンドラ。
379 *             :
380 * Argument    : unsigned long status
381 * Return Value : none
382 * Calling Functions :
383 *"FUNC COMMENT END"*****/
384 void lan_send_handler( unsigned long status )
385 {
386 }
387
388 /*"FUNC COMMENT"*****
389 * ID          :
390 * Outline     : 受信割り込み関数
391 * Include     : #include "iodefine.h"
392 *             : #include "ether.h"
393 * Declaration : void lan_recv_handler( unsigned long status )
394 * Description  : EDMAC (EESR)に関する受信関連の割り込みハンドラ。
395 *             :
396 * Argument    : unsigned long status
397 * Return Value : none
398 * Calling Functions :
399 *"FUNC COMMENT END"*****/
400 void lan_recv_handler( unsigned long status )
    
```

### 3.11 サンプルプログラムリスト "ether.c"(9)

```

401  {
402      c_recv++;                /* 受信割込み回数をカウント */
403  }
404
405  /*"FUNC COMMENT"*****
406  * ID                          :
407  * Outline                     : E-MAC 割込み関数
408  * Include                     : #include "iodefine.h"
409  *                             : #include "ether.h"
410  * Declaration                 : void lan_etherc_handler( unsigned long status )
411  * Description                 : E-MAC(ECSR)に関する割込みハンドラ。
412  *                             :
413  * Argument                    : unsigned long status
414  * Return Value                : none
415  * Calling Functions          :
416  *"FUNC COMMENT END"*****/
417  void lan_etherc_handler( unsigned long status )
418  {
419  }
420
421
422  /* End of file */
    
```

#### 4. 参考ドキュメント

- ソフトウェアマニュアル  
SH-4A ソフトウェアマニュアル (RJJ09B0090)  
(最新版をルネサス テクノロジホームページから入手してください)
- ハードウェアマニュアル  
SH7763 グループ ハードウェアマニュアル (RJJ09B0260)  
(最新版をルネサス テクノロジホームページから入手してください)

ホームページとサポート窓口

ルネサス 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	2009.10.30	—	初版発行

すべての商標および登録商標は、それぞれの所有者に帰属します。



## 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事業の目的で使用しないでください。また、輸出に際しては、「外国為替および外国貿易法」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いただきますとともに、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ください。
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のですが、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ください。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ください。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ください。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。
  - 1 1. 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。
  - 1 2. 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いたします。
  - 1 3. 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ください。

D039444