

# SH7216グループ

## イーサネット PHY-LSI 自動交渉例

R01AN0053JJ0200  
Rev.2.00  
2010.07.30

### 要旨

本アプリケーションノートは、SH7216 にイーサネット PHY-LSI を接続し、相手先と自動交渉するための設定例について説明します。

### 動作確認デバイス

SH7216

### 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.....	16
4. 参考ドキュメント.....	34

## 1. はじめに

### 1.1 仕様

- 本応用例では、SH7216 に接続したイーサネット PHY-LSI から自動交渉結果を取得します。取得した接続モード(全二重モードまたは半二重モード)は EtherC に設定されます。
- イーサネット PHY-LSI は Realtek Semiconductor Corp.製の RTL8201 を使用しています。
- イーサネット PHY-LSI のリンクには自動交渉機能を用います。

### 1.2 使用機能

- ピンファンクションコントローラ (PFC)
- イーサネットコントローラ (EtherC)

### 1.3 適用条件

マイコン	SH7216
動作周波数	内部クロック : 200 MHz バスクロック : 50 MHz 周辺クロック : 50 MHz
統合開発環境	ルネサス エレクトロニクス製 High-performance Embedded Workshop Ver.4.07.00
Cコンパイラ	ルネサス エレクトロニクス製 SuperH RISC engine ファミリ C/C++コンパイラパッケージ Ver.9.03 Release 00
コンパイルオプション	High-performance Embedded Workshopでのデフォルト設定 (-cpu=sh2afpu -fpu=single -debug -gbr=auto -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1)

### 1.4 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。合わせて参照してください。

- SH7216 グループ 初期設定例
- SH7216 グループ イーサネット受信設定例
- SH7216 グループ イーサネット送信設定例

## 2. 応用例の説明

本応用例では、イーサネット PHY-LSI により自動交渉を行います。自動交渉結果は EtherC の PHY 部インタフェースレジスタ (PIR) を経由して読み出します。

### 2.1 使用機能の動作概要

物理層のリンク処理自体はイーサネット PHY-LSI の機能で実行されます。そのため SH7216 内蔵の EtherC は、イーサネット PHY-LSI から結果を読み出すだけでリンク結果を取得できます。本応用例では自動交渉機能が許可されるように PHY-LSI の端子を設定しています。イーサネット PHY-LSI の詳細な機能については、イーサネット PHY-LSI のデータシートを参照してください。

EtherC とイーサネット PHY-LSI 間のインタフェースは、MII インタフェースとして IEEE802.3 で規格化されています。図 1 に SH7216 と RTL8201CP との接続例を示します。

自動交渉結果はイーサネット PHY-LSI 内部のレジスタに格納され、MDC 端子と MDIO 端子によるシリアルインタフェースを使用して読み出します。SH7216 では、PIR レジスタを使用することで両端子のリード/ライトが可能です。PHY-LSI 内部のレジスタアクセス手順は「2.2 MII レジスタ群のアクセス手順」を参照してください。

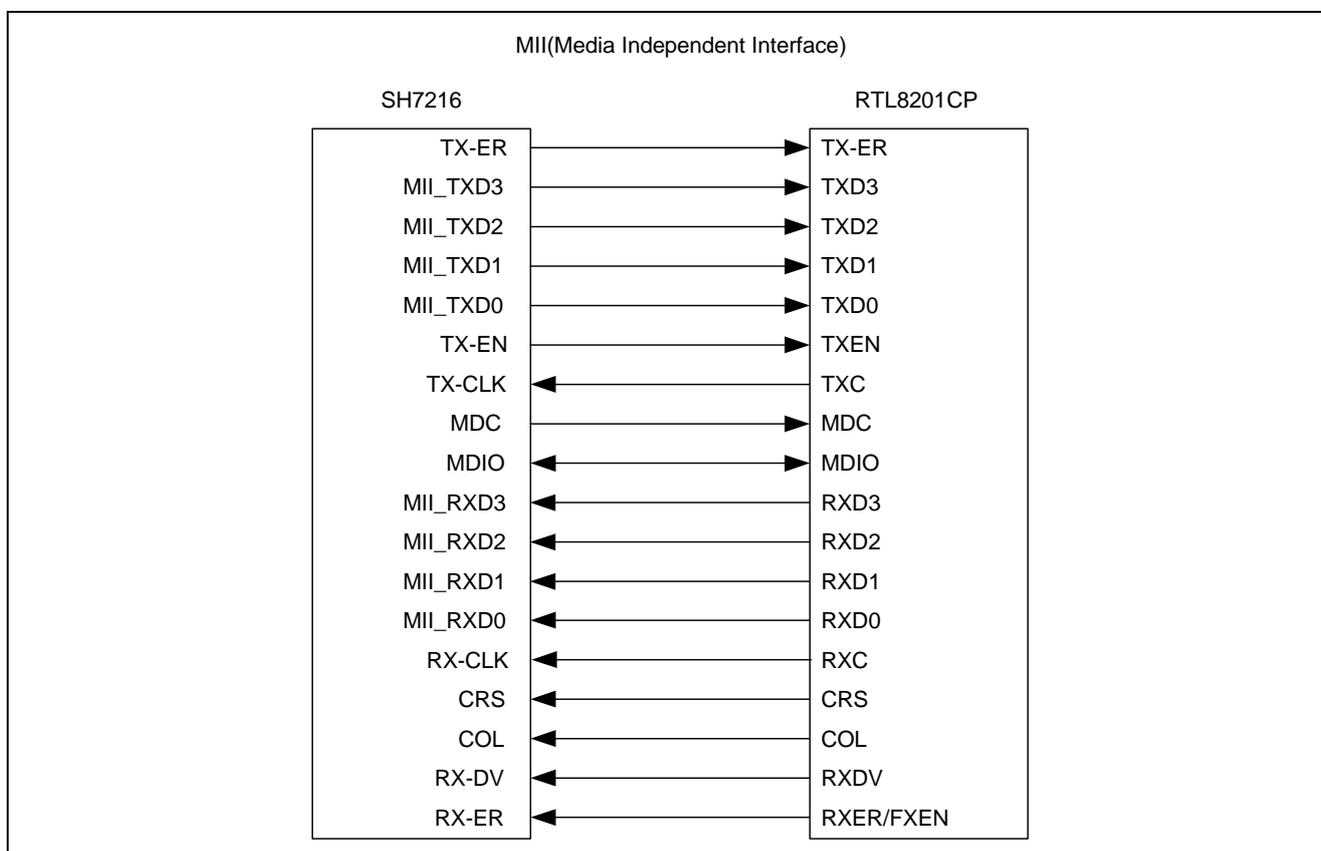


図1 RTL8201CP との接続例

## 2.2 MII レジスタ群のアクセス手順

イーサネット PHY-LSI の内部レジスタである MII レジスタ群にアクセスする手順を説明します。

MII レジスタ群へのアクセスは、MDC と MDIO(いずれも EtherC 側の端子名)の 2 本を使用して行います。MDC は同期用クロック端子、MDIO はデータ入出力端子です。各端子の状態は EtherC の PIR レジスタで参照または変更できます。制御用端子がないため、MII では必ず規定されたフォーマット (MII 管理フレーム) ごとにデータを出力しなければなりません。MII 管理フレームを図 2 に示します。本サンプルプログラムでは IDLE 状態で Z 出力を 1 ビット実行します。IEEE802.3 規格ではクロック入力について言及されていませんが、PHY-LSI によっては正しく接続できない場合があり、安全のため実行しています。

MII 管理フレームの入出力は、PRE から順に 1 ビット単位で行います。1 ビット単位の入出力フローを図 3 ~ 図 5 に示します。MDC および MDIO の入出力タイミングは IEEE802.3 規格を満たしてください。表 1 および図 6 に IEEE802.3 規格の入出力タイミングを示します。

アクセス種別	MII管理フレーム							
	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
ビット数	32	2	2	5	5	2	16	1
リード	1..1	01	10	AAAAA	RRRRR	Z0	D..D	Z
ライト	1..1	01	01	AAAAA	RRRRR	10	D..D	Z

### [記号説明]

- PRE (preamble) : 同期を取る。32個の連続した1を出力。
- ST(start of frame) : フレームの先頭を示す。01の出力。
- OP(operation code) : リード/ライトを選択。リードの場合は10、ライトの場合は01を出力。
- PHYAD(PHY Address) : 複数のPHY-LSIを接続する場合の識別アドレス。PHY-LSIの端子設定によりアドレスを指定することが多い。MSBから出力。
- REGAD(Register Address) : MIIレジスタの番号を指定。MSBから出力。
- TA(turn around) : 信号の衝突を防ぐため、MDIOの送信元を切り替える。  
(a) リード時は、1ビット分のバス解放 (Z出力) を行う。  
(PHYからは0が出力されるので、Z0と表記)  
(b) ライト時は10を出力。
- DATA(data) : レジスタのリード値またはライト値。16ビットデータ。MSBから順次ライトまたはリード。
- IDLE(IDLE condition) : 次のMII管理フォーマット入力までの待機処理。バス解放(Z出力)を行う。

図2 MII 管理フレームフォーマット

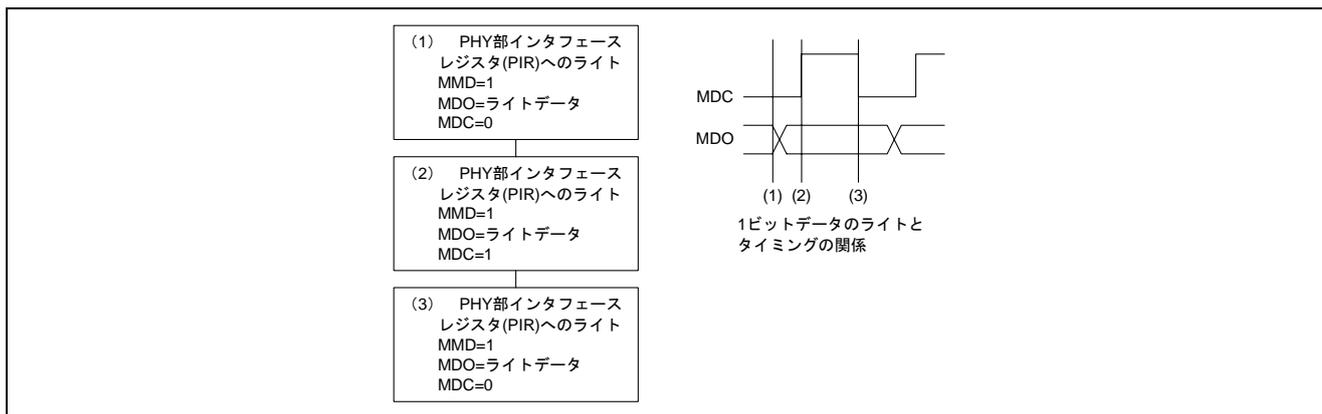


図3 1ビットデータのライトフロー

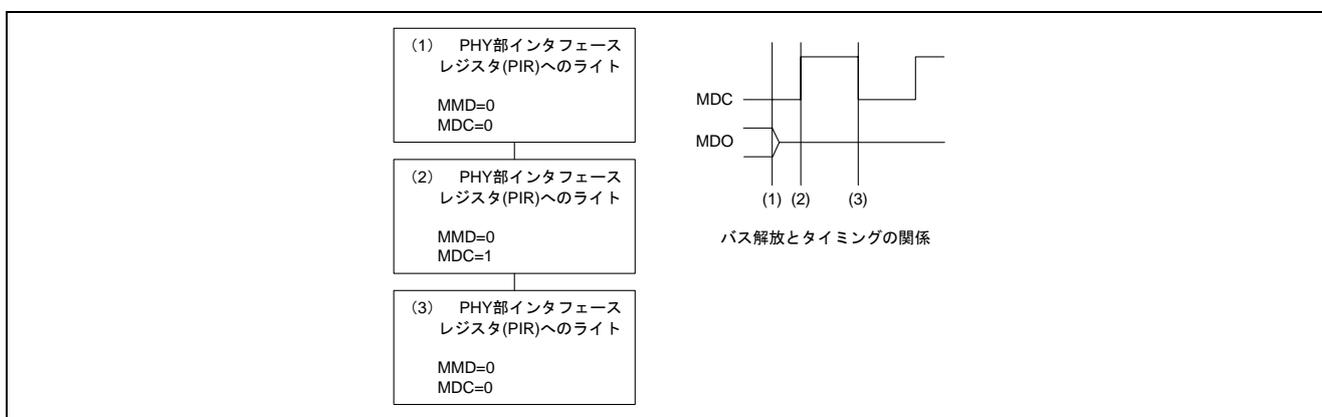


図4 バス解放フロー (Z出力)

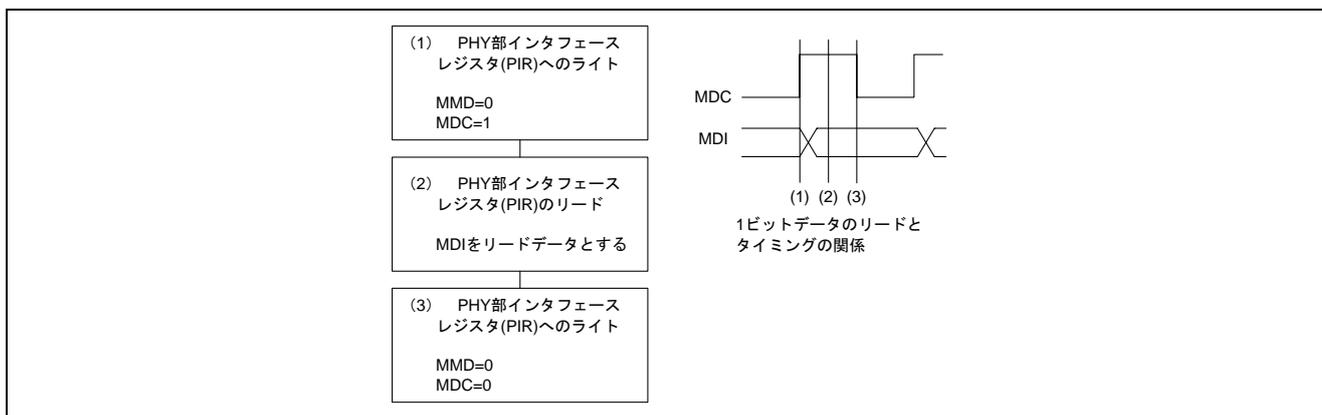


図5 1ビットデータのリードフロー

表1 MDC/MDIO 入出カタイミング

項目	記号	Min.	Max.	単位
MDC High レベルパルス幅	$t_1$	160	—	ns
MDC Low レベルパルス幅	$t_2$	160	—	ns
MDC サイクル時間	$t_3$	400	—	ns
MDIO セットアップ時間	$t_4$	10	—	ns
MDIO ホールド時間	$t_5$	10	—	ns
MDIO 出力遅延時間	$t_6$	0	300	ns

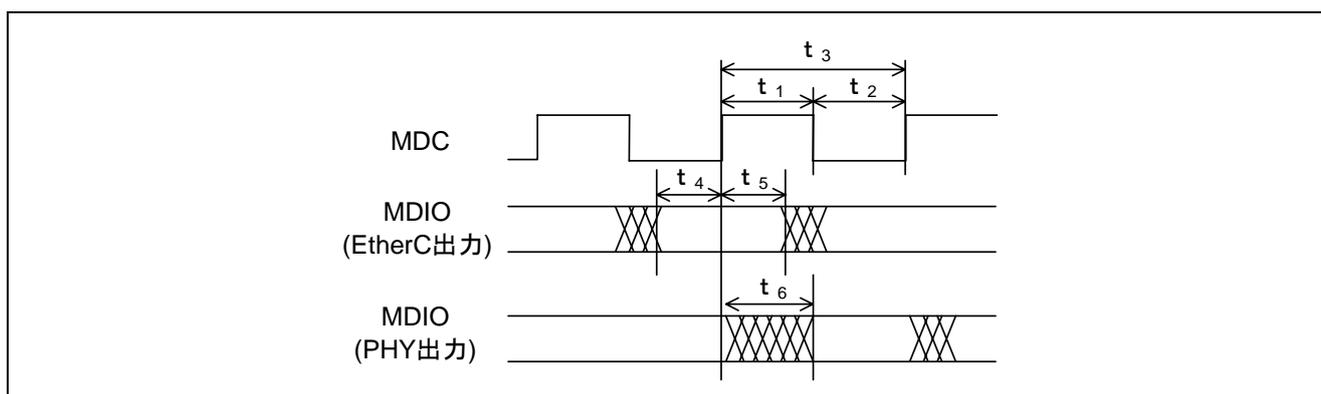


図6 MDC/MDIO 入出カタイミング

## 2.3 参考プログラムの説明

参考プログラムは、

- main.c
- phy.c

の2つのソースプログラムと、関連アプリケーションノートの「SH7216 グループ 初期設定例」で作成した各初期化用ファイルから構成されています。

- main.c

メイン関数（main 関数）が記述されており、PHY-LSI のソフトウェアリセット、自動交渉結果取得を実行します。図 7 に main 関数のフローを示します。

- phy.c

PHY-LSI の初期化関数（phy\_init 関数）、自動交渉結果の取得関数（phy\_set\_autonegotiate 関数）が記述されています。図 8 に phy\_init 関数、図 9 に phy\_set\_autonegotiate 関数のフローを示します。

図 10～図 15 に phy\_init 関数、phy\_set\_autonegotiate 関数内で実行される下位関数のフローを示します。

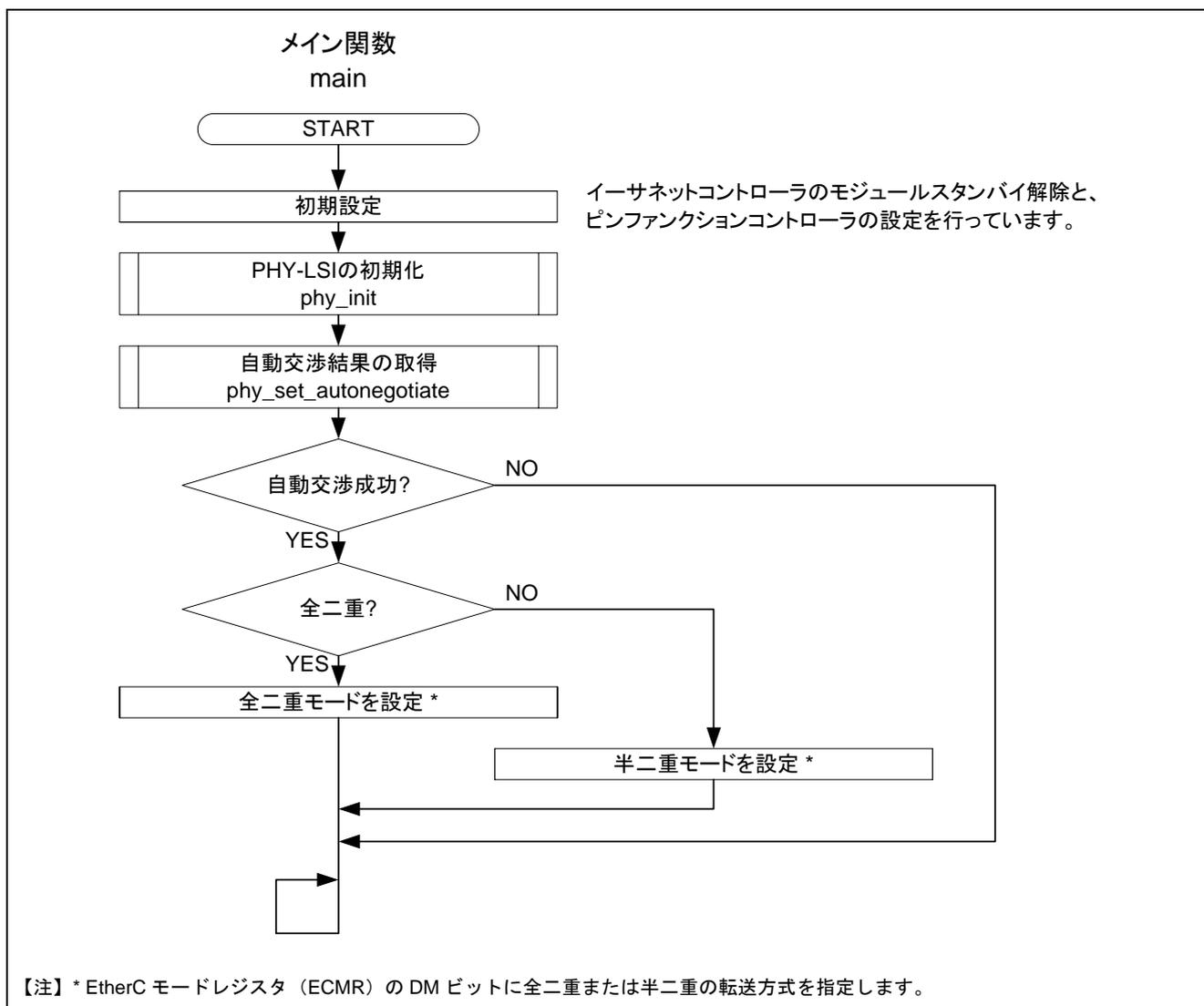


図7 メイン関数の処理フロー

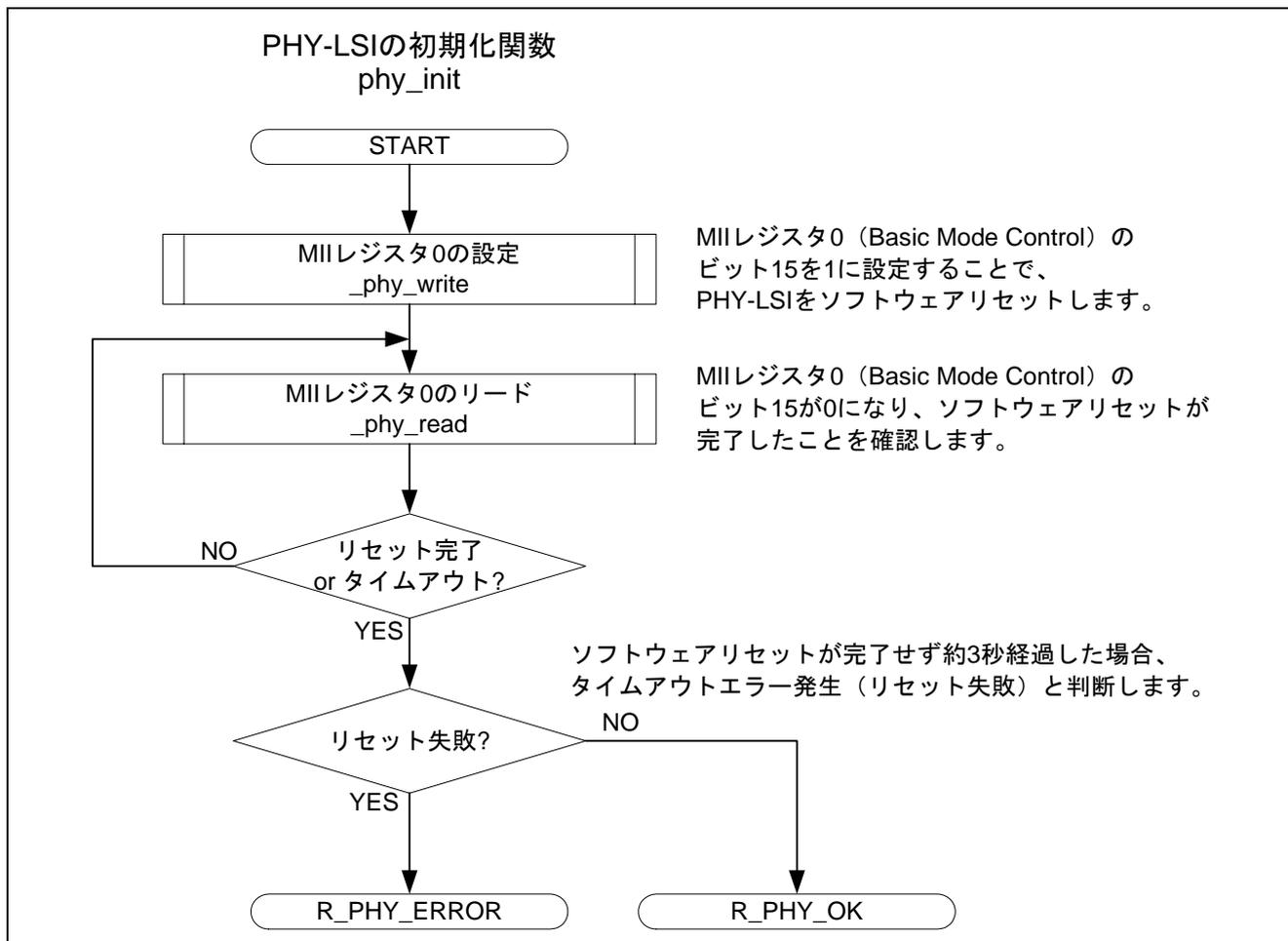


図8 PHY-LSI 初期化関数の処理フロー

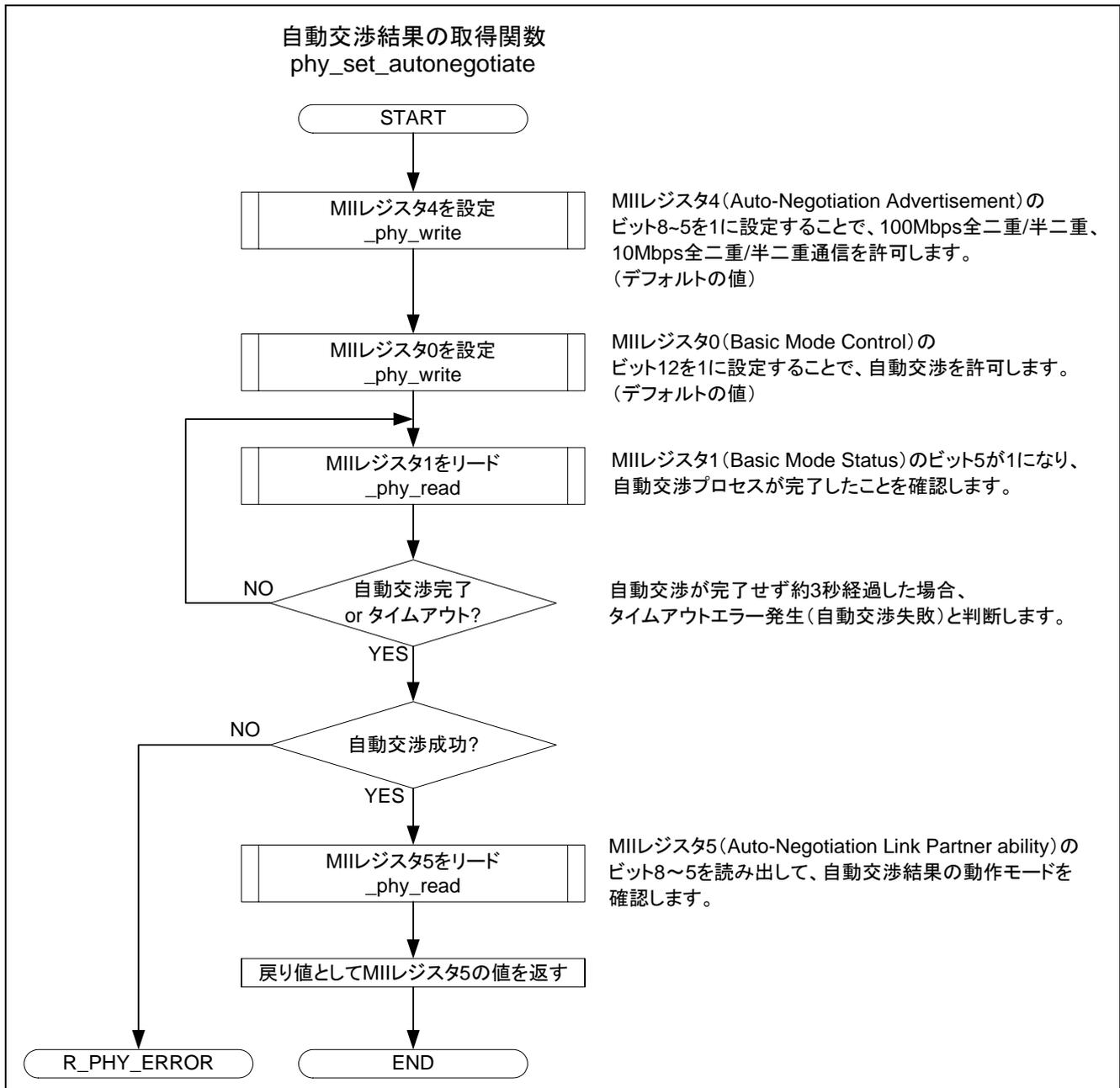


図9 自動交渉結果取得関数の処理フロー

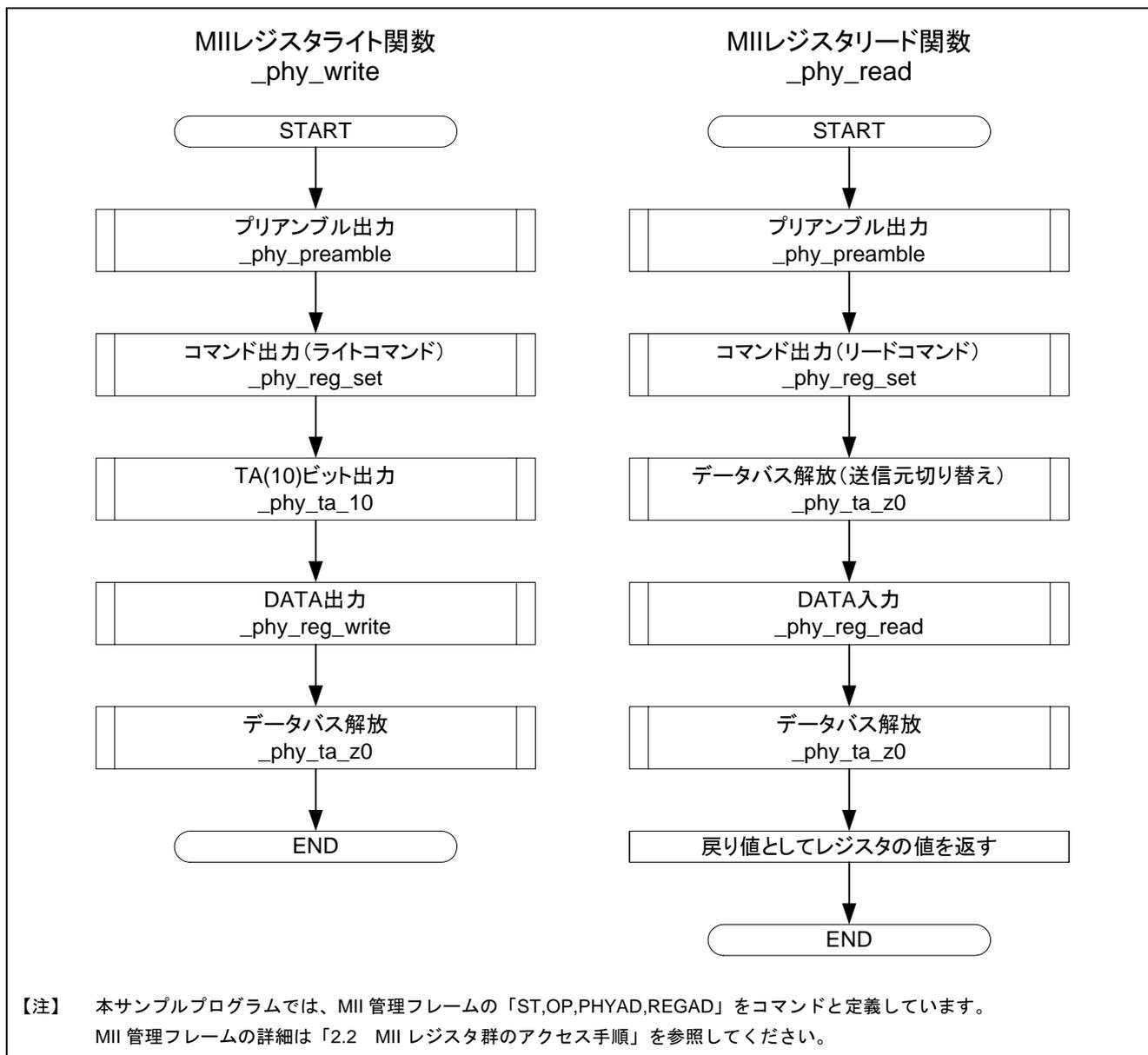


図10 MII レジスタアクセスの処理フロー (1)

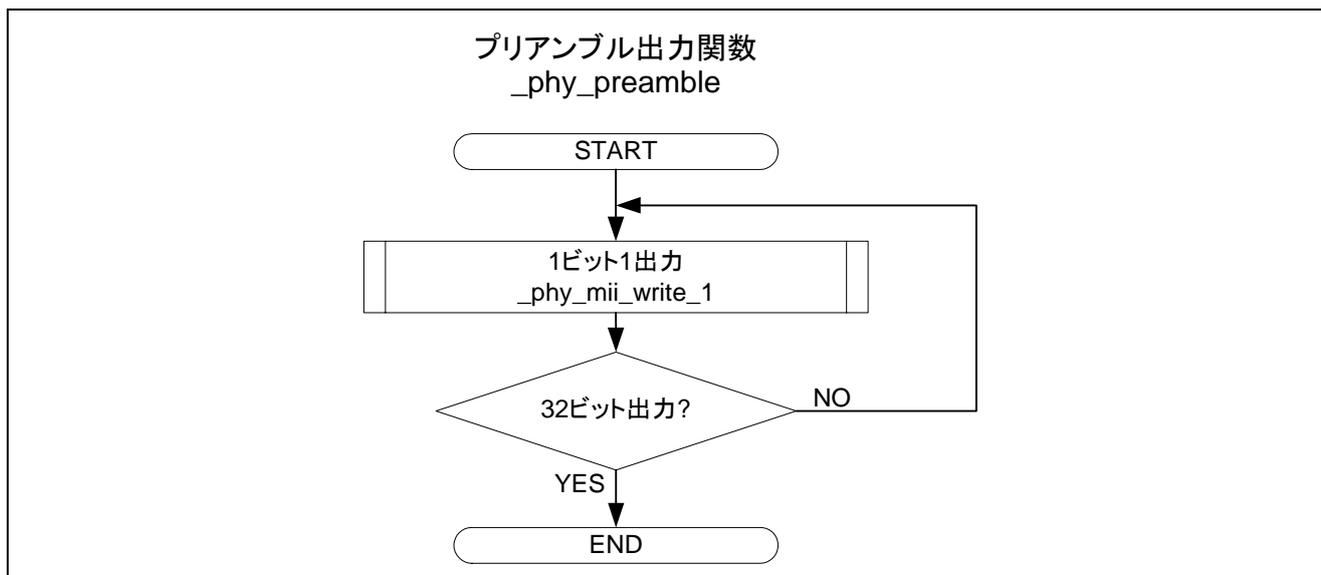


図11 MII レジスタアクセスの処理フロー(2)

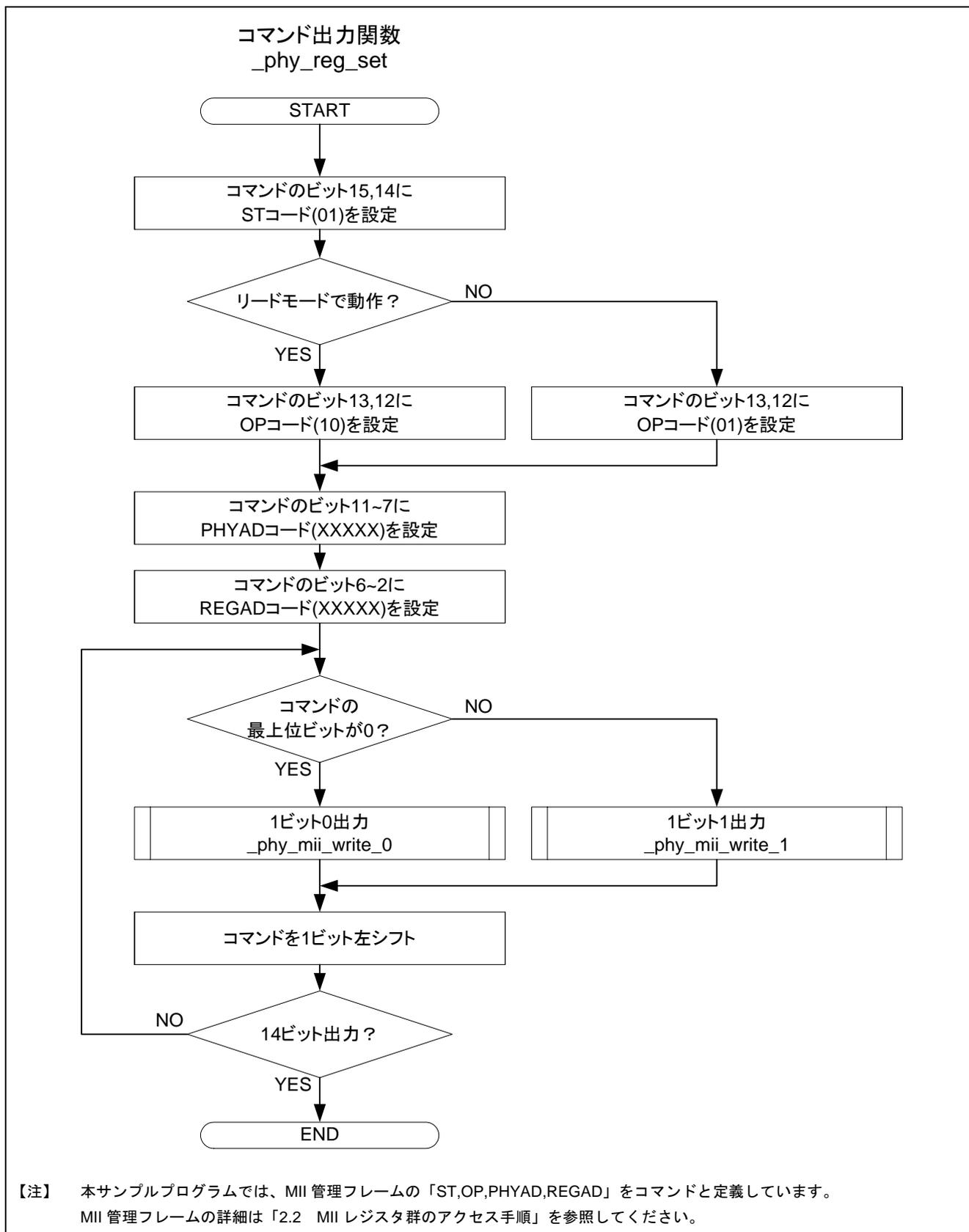


図 12 MII レジスタアクセスの処理フロー(3)

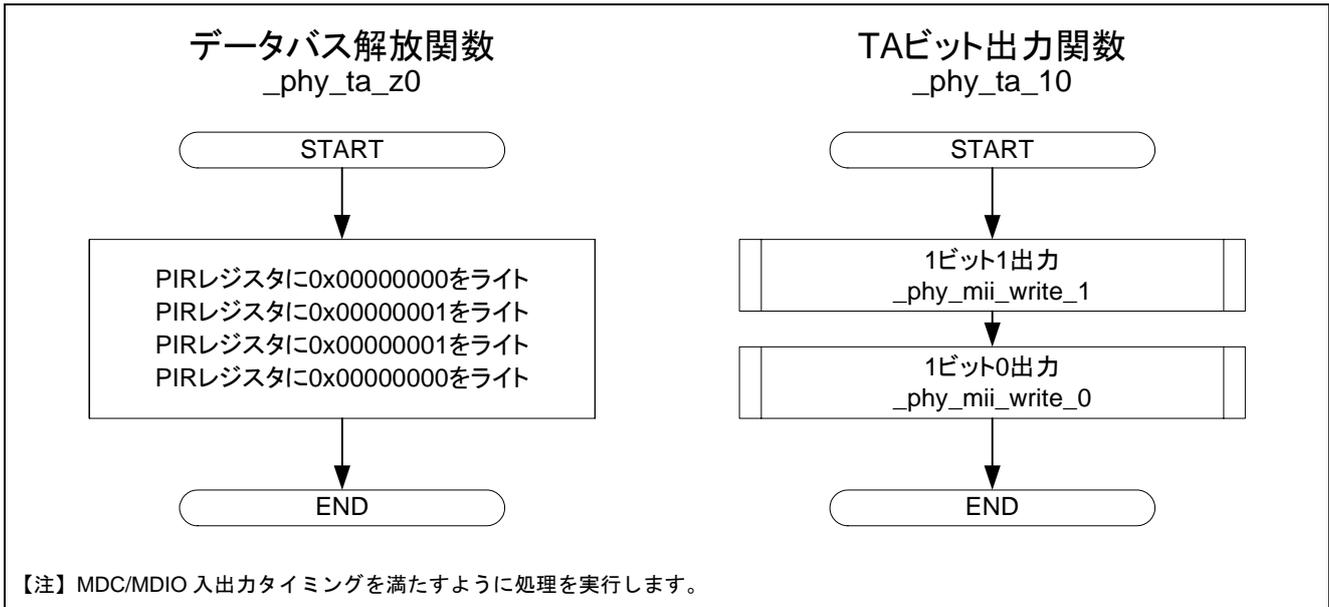


図 13 PHY-LSI レジスタアクセスの処理フロー(4)

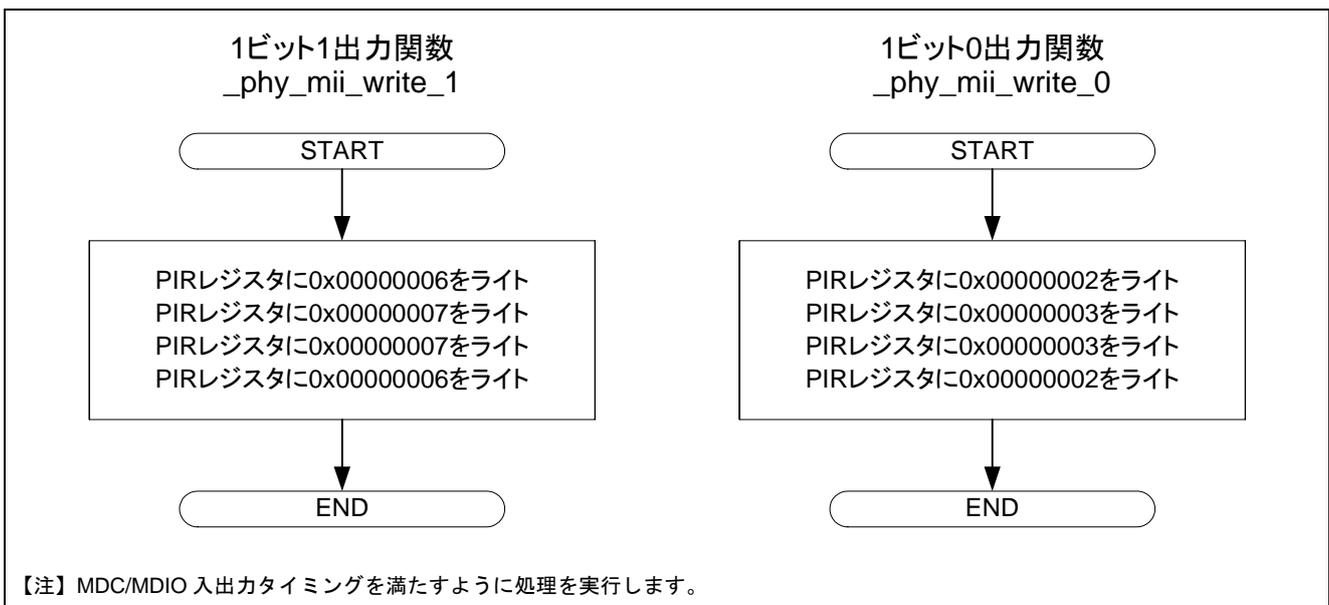


図 14 MII レジスタアクセスの処理フロー(5)

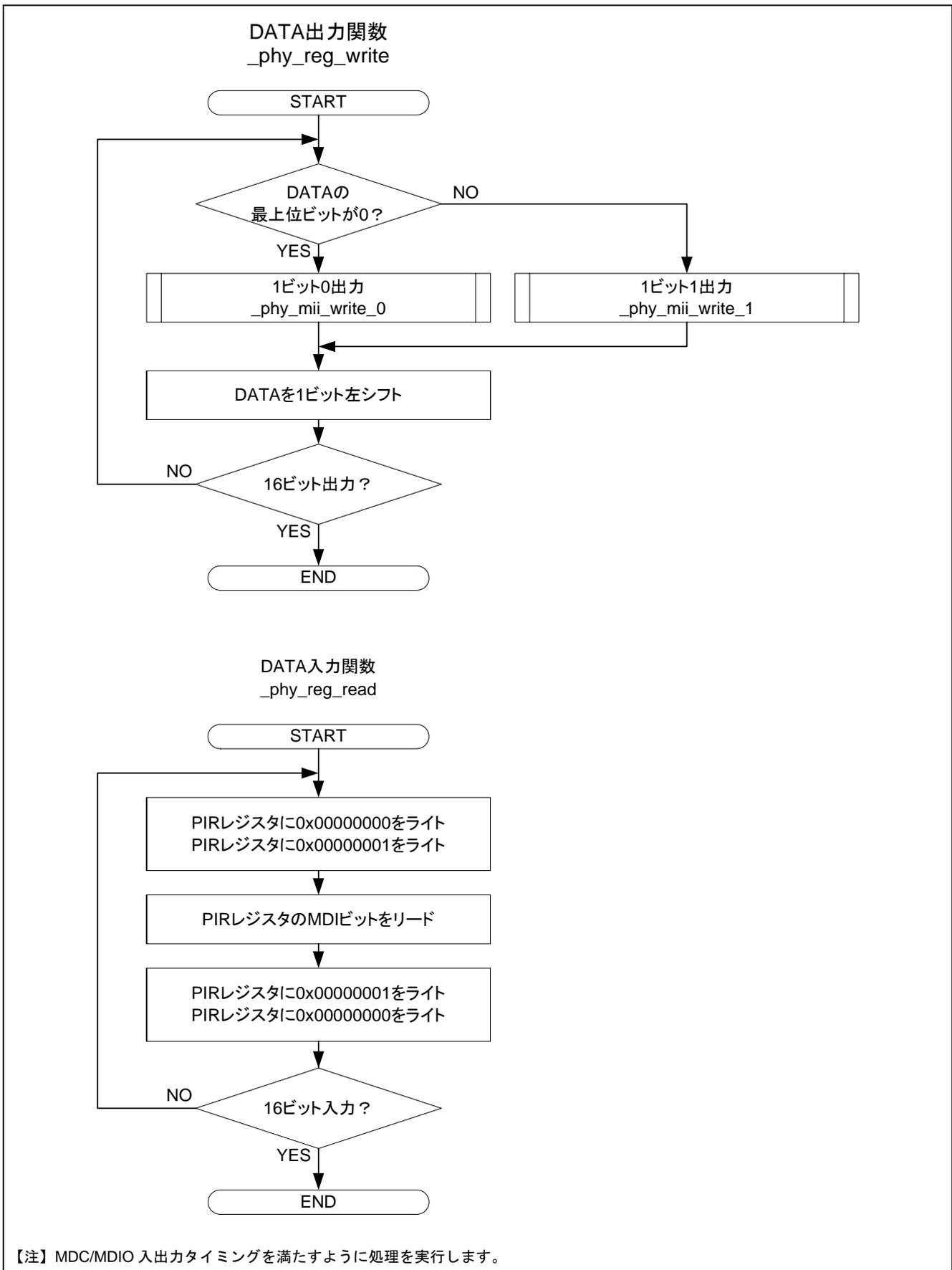


図 15 MII レジスタアクセスの処理フロー(6)

## 2.4 参考プログラムにおける設定内容

表 2に参考プログラムでの設定を示します。

表2 参考プログラムでの設定

項目	設定内容
使用する PHY-LSI	Realtek Semiconductor Corp.製 RTL8201CP
リンクモード	100 Mbps (全二重、半二重) および 10 Mbps (全二重、半二重)
リンク決定方式	自動交渉 (Auto-Negotiation)
使用する MII レジスタ	<ul style="list-style-type: none"> <li>Basic Mode Control (アドレス : H'00)</li> <li>Basic Mode Status (アドレス : H'01)</li> <li>Auto-Negotiation Advertisement (アドレス : H'04)</li> <li>Auto-Negotiation Link Partner ability (アドレス : H'05)</li> </ul>

## 2.5 参考プログラム使用時の注意点

- 参考プログラムは、PHY のリンク決定方式として自動交渉モードを前提としています。
- 相手先が自動交渉モードの場合は表 3の優先順位に従い、リンクモードが決定します。
- phy\_init 関数内のリセット完了待ち、phy\_set\_autonegotiate 関数内の自動交渉完了待ちでは、約 3 秒間 (内部クロック : 200 MHz 動作時) レジスタの値を参照するため、参照中はそれぞれの関数から戻りません。

表3 リンクモードの優先順位

優先順位		リンクモード
高	1	100 Mbps 全二重
	2	100 Mbps 半二重
	3	10 Mbps 全二重
低	4	10 Mbps 半二重

### 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 Electronics Corp. and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics 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 *  ELECTRONICS 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 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) 2010 Renesas Electronics Corporation. All Rights Reserved.
29 *  "FILE COMMENT" ***** Technical reference data *****
30 *  System Name : SH7216 Sample Program
31 *  File Name   : main.c
32 *  Abstract    : イーサネット PHY-LSI 自動交渉例
33 *  Version     : 2.00.00
34 *  Device      : SH7216
35 *  Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36 *              : C/C++ compiler package for the SuperH RISC engine family
37 *              : (Ver.9.03 Release00).
38 *  OS          : None
39 *  H/W Platform: R0K572167 (CPU board)
40 *  Description : イーサネット送信に必要な初期設定および送信処理を行います。
41 *****/
42 *  History     : Nov.18,2009 Ver.1.00.00
43 *              : Jul.23,2010 Ver.2.00.00 ルネサス標準APIに対応
44 *  "FILE COMMENT END" *****/
45 #include "iodefine.h"
46 #include "stdint.h"
47 #include "phy.h"
48
```

## 3.2 サンプルプログラムリスト "main.c" (2)

```
49  /* ==== Prototype Declaration ==== */
50  void main(void);
51
52  /* ==== Variable Declaration ==== */
53
54  /*"FUNC COMMENT"*****
55  * ID      :
56  * Outline  : サンプルプログラムメイン
57  *-----
58  * Include  : "iodefine.h", "stdint.h", and "phy.h"
59  *-----
60  * Declaration : void main(void);
61  *-----
62  * Description : R_Ether_Open 関数内の「PHYの初期化」「PHYの自動交渉開始」にて、
63  *              : PHY-LSIが自動交渉機能により決定したPHYリンクモードを、
64  *              : 内蔵イーサネットコントローラ(EtherC)を使用して読み出しています。
65  *              : 全二重/半二重の結果は、EtherCにて設定します。
66  *              : またPHYモジュールには、REALTEK社製RTL8201CPを使用します。
67  *-----
68  * Argument   : void
69  *-----
70  * Return Value : void
71  *-----
72  * Note       : None
73  *"FUNC COMMENT END"*****/
74  void main(void)
75  {
76      uint32_t ch = 0;
77      uint16_t phydata;
78
79      /* ==== EtherC/E-DMACのモジュールスタンバイ解除 ==== */
80      STB.CR4.BIT._ETHER = 0;
81      /* ==== PFC設定(EtherC用) ==== */
82      PFC.PACRL4.BIT.PA12MD = 7; /* TX_CLK (input) */
83      PFC.PACRL3.WORD = 0x7777; /* TX_EN,MII_TXD0,MII_TXD1,MII_TXD2 (output) */
84      PFC.PACRL2.BIT.PA7MD = 7; /* MII_TXD3 (output) */
85      PFC.PACRL2.BIT.PA6MD = 7; /* TX_ER (output) */
86      PFC.PDCRH4.WORD = 0x7777; /* RX_DV,RX_ER,MII_RXD3,MII_RXD2 (input) */
87      PFC.PDCRH3.WORD = 0x7777; /* MII_RXD1,MII_RXD0,RX_CLK,CRS (input) */
88      PFC.PDCRH2.WORD = 0x7777; /* COL (input),WOL,EXOUT,MDC (input) */
89      PFC.PDCRH1.BIT.PD19MD = 7; /* LINKSTA (input) */
90      PFC.PDCRH1.BIT.PD18MD = 7; /* MDIO (input/output) */
91
```

## 3.3 サンプルプログラムリスト "main.c" (3)

```
92     /* ==== PHY-LSI の初期化 ==== */
93     phy_init();
94
95     /* ==== PHY-LSI 自動交渉結果の取得 ==== */
96     phydata = phy_set_autonegotiate();
97
98     /* ---- 自動交渉の可否判定 ---- */
99     if(phydata == R_PHY_ERROR){
100         /* 自動交渉失敗 */
101     }
102     /* ---- 接続先性能の判定 ---- */
103     else if(phydata & 0x0100){           /* PHY-LSI のレジスタ 0 の判定 */
104                                         /* bit8 : DuplexMode : 1 ---- 全二重モードをサポート */
105
106         EtherC.ECMR.BIT.DM = 1;       /* 全二重通信 */
107     }
108     else{
109         EtherC.ECMR.BIT.DM = 0;       /* 半二重通信 */
110     }
111
112     while(1){
113         /* sleep */
114     }
115 }
116
117 /* End of file */
118
```

## 3.4 サンプルプログラムリスト "phy.c" (1)

```
1  /*****
2  *  DISCLAIMER
3  *
4  *  This software is supplied by Renesas Electronics Corp. and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics 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 *  ELECTRONICS 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 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) 2010 Renesas Electronics Corporation. All Rights Reserved.
29 *  "FILE COMMENT" ***** Technical reference data *****
30 *  System Name : SH7216 Sample Program
31 *  File Name   : phy.c
32 *  Version    : 2.00.00
33 *  Device     : SH7216
34 *  Tool-Chain : High-performance Embedded Workshop (Ver.4.07.00).
35 *             : C/C++ compiler package for the SuperH RISC engine family
36 *             :                               (Ver.9.03 Release00).
37 *  OS        : None
38 *  H/W Platform: R0K572167 (CPU board)
39 *  Description : Ethernet PHY device driver
40 *****/
41 *  History    : Jun.10.2009 Ver.1.00.00
42 *             : Jul.23,2010 Ver.2.00.00 ルネサス標準 API に対応
43 *  "FILE COMMENT END" *****/
44 #include "iodefine.h"
45 #include "stdint.h"
46 #include "phy.h"
47
48 /* ==== Prototype declaration ==== */
49 uint16_t _phy_read( uint16_t reg_addr );
50 void _phy_write( uint16_t reg_addr, uint16_t data );
51 void _phy_preamble( void );
```

## 3.5 サンプルプログラムリスト "phy.c" (2)

```
52 void _phy_reg_set( uint16_t reg_addr, int32_t option );
53 void _phy_reg_read( uint16_t *data );
54 void _phy_reg_write( uint16_t data );
55 void _phy_ta_z0( void );
56 void _phy_ta_10( void );
57 void _phy_mii_write_1( void );
58 void _phy_mii_write_0( void );
59
60 /*"FUNC COMMENT"*****
61 * ID      :
62 * Outline : PHY モジュールのソフトウェアリセット
63 *-----
64 * Include :
65 *-----
66 * Declaration : int16_t phy_init( void );
67 *-----
68 * Description : イーサネットの PHY-LSI を初期化します。
69 *-----
70 * Argument   : void
71 *-----
72 * Return Value : R_PHY_OK;   PHY-LSI の初期化成功
73 *              : R_PHY_ERROR; PHY-LSI の初期化失敗
74 *-----
75 * Note       : None
76 *"FUNC COMMENT END"*****/
77 int16_t phy_init( void )
78 {
79     uint16_t reg;
80     uint16_t count;
81
82     /* ==== PHY-LSI の初期化 ==== */
83     _phy_write(BASIC_MODE_CONTROL_REG, 0x8000);
84
85     count = 0;
86
87     do{
88         reg = _phy_read(BASIC_MODE_CONTROL_REG);
89         count++;
90     }while((reg & 0x8000) && (count < 0x4000));
91
92     if(count >= 0x4000){
93         return R_PHY_ERROR;
94     }
95     else{
96         return R_PHY_OK;
97     }
98 }
99
```

## 3.6 サンプルプログラムリスト "phy.c" (3)

```
100  /*"FUNC COMMENT"*****
101  * ID      :
102  * Outline : 100Mbps 全二重モードの設定
103  *-----
104  * Include :
105  *-----
106  * Declaration : void phy_set_100full( void );
107  *-----
108  * Description : PHY-LSI を 100Mbps 全二重モードに設定します。
109  *-----
110  * Argument   : void
111  *-----
112  * Return Value : void
113  *-----
114  * Note       : None
115  *"FUNC COMMENT END"*****/
116 void phy_set_100full( void )
117 {
118     _phy_write(BASIC_MODE_CONTROL_REG, 0x2100);
119 }
120
121 /*"FUNC COMMENT"*****
122 * ID      :
123 * Outline : 10Mbps 半二重モードの設定
124 *-----
125 * Include :
126 *-----
127 * Declaration : void phy_set_10half( void );
128 *-----
129 * Description : PHY-LSI を 10Mbps 半二重モードに設定します。
130 *-----
131 * Argument   : void
132 *-----
133 * Return Value : void
134 *-----
135 * Note       : None
136 *"FUNC COMMENT END"*****/
137 void phy_set_10half( void )
138 {
139     _phy_write(BASIC_MODE_CONTROL_REG, 0x0000);
140 }
141
```

## 3.7 サンプルプログラムリスト "phy.c" (4)

```

142  /*"FUNC COMMENT"*****
143  * ID      :
144  * Outline : PHY リンクの自動交渉結果検出
145  *-----
146  * Include :
147  *-----
148  * Declaration : int16_t phy_set_autonegotiate( void );
149  *-----
150  * Description : 自動交渉結果を MII 管理インターフェースを使って読み出し、
151  *              : 戻り値として返却します。
152  *              : 本サンプルプログラムで使用する PHY は、100Mbps 全二重モードまで
153  *              : すべてサポートしていますので、リンク先が自動交渉に対応して
154  *              : いれば、リンク先の最高性能となるリンクモードで接続されます。
155  *              : リンク先が自動交渉に対応していない場合は、並列検出機能により
156  *              : リンク速度が検出されます。その場合は半二重モードとなります。
157  *-----
158  * Argument  : void
159  *-----
160  * Return Value : bit 8 - Full duplex 100 mbps
161  *              : bit 7 - Half duplex 100 mbps
162  *              : bit 6 - Full duplex 10 mbps
163  *              : bit 5 - Half duplex 10 mbps
164  *              : bit 4-0 - Always set to 00001 (IEEE 802.3)
165  *              : -1 - error
166  *-----
167  * Note      : None
168  *"FUNC COMMENT END"*****/
169  int16_t phy_set_autonegotiate( void )
170  {
171      uint16_t reg;
172      uint16_t count;
173
174      _phy_write(AN_ADVERTISEMENT_REG, 0x01E1);
175      _phy_write(BASIC_MODE_CONTROL_REG, 0x1000);
176
177      count = 0;
178
179      do{
180          reg = _phy_read(BASIC_MODE_STATUS_REG);
181          count++;
182      }while(!(reg & 0x0020) && count < 0x4000);
183
184      if(count >= 0x4000){
185          return R_PHY_ERROR;
186      }
187      else{
188          return ((int16_t)_phy_read(AN_LINK_PARTNER_ABILITY_REG));
189      }
190  }
191

```

## 3.8 サンプルプログラムリスト "phy.c" (5)

```
192 /*"FUNC COMMENT"*****
193 * ID      :
194 * Outline : PHY モジュールレジスタリード
195 *-----
196 * Include :
197 *-----
198 * Declaration : uint16_t _phy_read( uint16_t reg_addr );
199 *-----
200 * Description : PHY モジュールレジスタ値を取得します。
201 *-----
202 * Argument    : uint16_t reg_addr ; I : 値を読み出す PHY レジスタアドレス
203 *-----
204 * Return Value : 取得したレジスタの値
205 *-----
206 * Note       : None
207 *"FUNC COMMENT END"*****/
208 uint16_t _phy_read( uint16_t reg_addr )
209 {
210     uint16_t data;
211
212     _phy_preamble();
213     _phy_reg_set( reg_addr, PHY_READ );
214     _phy_ta_z0();
215     _phy_reg_read( &data );
216     _phy_ta_z0();
217
218     return ( data );
219 }
220
221 /*"FUNC COMMENT"*****
222 * ID      :
223 * Outline : PHY モジュールレジスタライト
224 *-----
225 * Include :
226 *-----
227 * Declaration : void _phy_write( uint16_t reg_addr, uint16_t data );
228 *-----
229 * Description : PHY モジュールレジスタに値を設定します。
230 *-----
231 * Argument    : unsigned short reg_addr ; I : 値を書き込む PHY レジスタアドレス
232 *              : unsigned short data    ; I : PHY レジスタに設定する値
233 *-----
234 * Return Value : void
235 *-----
236 * Note       : None
237 *"FUNC COMMENT END"*****/
238 void _phy_write( uint16_t reg_addr, uint16_t data )
239 {
240     _phy_preamble();
241     _phy_reg_set( reg_addr, PHY_WRITE );
242     _phy_ta_l0();
243     _phy_reg_write( data );
244     _phy_ta_z0();
245 }
```

## 3.9 サンプルプログラムリスト "phy.c" (6)

```
246
247 /*"FUNC COMMENT"*****
248 * ID      :
249 * Outline  : PHY モジュールレジスタへのアクセス準備
250 *-----
251 * Include  :
252 *-----
253 * Declaration : void _phy_preamble( void );
254 *-----
255 * Description : PHY モジュールレジスタにアクセスするための前準備として、
256 *              : MII 管理インタフェースに 1 を出力します。
257 *-----
258 * Argument   : void
259 *-----
260 * Return Value : void
261 *-----
262 * Note       : None
263 *"FUNC COMMENT END"*****/
264 void _phy_preamble( void )
265 {
266     int16_t i;
267
268     i = 32;
269     while( i > 0 ){
270         /* MII (Media Independent Interface)ブロックに 1 を出力 */
271         _phy_mii_write_1();
272         i--;
273     }
274 }
275
```

## 3.10 サンプルプログラムリスト "phy.c" (7)

```
276 /*"FUNC COMMENT"*****
277 * ID      :
278 * Outline : PHY モジュールレジスタモード設定
279 *-----
280 * Include :
281 *-----
282 * Declaration : void _phy_reg_set( uint16_t reg_addr, int32_t option );
283 *-----
284 * Description : PHY モジュールレジスタの R/W モードを設定します。
285 *-----
286 * Argument   : unsigned short reg_addr ; I : PHY レジスタアドレス
287 *             : int option           ; I : R/W モード指定
288 *-----
289 * Return Value : void
290 *-----
291 * Note        : None
292 *"FUNC COMMENT END"*****/
293 void _phy_reg_set( uint16_t reg_addr, int32_t option )
294 {
295     int32_t i;
296     uint16_t data;
297
298     data = 0;
299     data = (PHY_ST << 14); /* ST code */
300
301     if( option == PHY_READ ){
302         data |= (PHY_READ << 12); /* OP code(RD) */
303     }
304     else{
305         data |= (PHY_WRITE << 12); /* OP code(WT) */
306     }
307
308     data |= (PHY_ADDR << 7); /* PHY Address */
309     data |= (reg_addr << 2); /* Reg Address */
310
311     i = 14;
312     while( i > 0 ){
313         if( (data & 0x8000) == 0 ){
314             _phy_mii_write_0();
315         }
316         else{
317             _phy_mii_write_1();
318         }
319         data <<= 1;
320         i--;
321     }
322 }
323
```

## 3.11 サンプルプログラムリスト "phy.c" (8)

```
324  /*"FUNC COMMENT"*****
325  * ID      :
326  * Outline : PHY モジュールレジスタ値取得
327  *-----
328  * Include :
329  *-----
330  * Declaration : void _phy_reg_read( uint16_t *data );
331  *-----
332  * Description : PHY モジュールレジスタ値を 1 ビットずつ取得します。
333  *             : 下記条件を満たすように信号を入出力します。
334  *             : ・MDC High パルス幅 : 160ns(min)
335  *             : ・MDC Low パルス幅  : 160ns(min)
336  *             : ・MDC サイクル時間  : 400ns(min)
337  *             : ・MDIO 出力遅延時間(from PHY) : 300ns(max)
338  *-----
339  * Argument  : uint16_t *data ; 0 : 取得した値の格納先アドレス
340  *-----
341  * Return Value : void
342  *-----
343  * Note      : システムに応じてウェイト時間を変更してください。
344  *"FUNC COMMENT END"*****/
345 void _phy_reg_read( uint16_t *data )
346 {
347     int32_t i,j;
348     uint16_t reg_data;
349
350     reg_data = 0;
351     i = 16;
352     /* ---- 1ビットずつデータ読み出し ---- */
353     while( i > 0 ){
354         for(j = MDC_WAIT; j > 0; j--){
355             EtherC.PIR.LONG = 0x00000000;
356         }
357         for(j = MDC_WAIT; j > 0; j--){
358             EtherC.PIR.LONG = 0x00000001;
359         }
360
361         reg_data <<= 1;
362         reg_data |= (uint16_t)((EtherC.PIR.LONG & 0x00000008) >> 3); /* MDI read */
363
364         for(j = MDC_WAIT; j > 0; j--){
365             EtherC.PIR.LONG = 0x00000001;
366         }
367         for(j = MDC_WAIT; j > 0; j--){
368             EtherC.PIR.LONG = 0x00000000;
369         }
370         i--;
371     }
372     *data = reg_data;
373 }
374
```

## 3.12 サンプルプログラムリスト "phy.c" (9)

```
375  /*"FUNC COMMENT"*****
376  * ID      :
377  * Outline : PHY モジュールレジスタ値設定
378  *-----
379  * Include :
380  *-----
381  * Declaration : void _phy_reg_write( uint16_t data );
382  *-----
383  * Description : PHY モジュールレジスタ値を 1 ビットずつ設定します。
384  *-----
385  * Argument   : uint16_t data ; I : レジスタに設定する値
386  *-----
387  * Return Value : void
388  *-----
389  * Note       : None
390  *"FUNC COMMENT END"*****/
391 void _phy_reg_write( uint16_t data )
392 {
393     int32_t i;
394
395     i = 16;
396     while( i > 0 ){
397         if( (data & 0x8000) == 0 ){
398             _phy_mii_write_0();
399         }
400         else{
401             _phy_mii_write_1();
402         }
403         i--;
404         data <<= 1;
405     }
406 }
407
```

## 3.13 サンプルプログラムリスト "phy.c" (10)

```
408 /*"FUNC COMMENT"*****
409 * ID      :
410 * Outline : PHY モジュールレジスタアクセス バス解放
411 *-----
412 * Include :
413 *-----
414 * Declaration : void _phy_ta_z0( void );
415 *-----
416 * Description : PHY モジュールへのアクセスをデータ読み出しに設定します。
417 *              : 下記条件を満たすように信号を出力します。
418 *              : ・MDC High パルス幅      : 160ns(min)
419 *              : ・MDC Low パルス幅       : 160ns(min)
420 *              : ・MDC サイクル時間      : 400ns(min)
421 *              : ・MDIO セットアップ時間 : 10ns(min)
422 *              : ・MDIO ホールド時間    : 10ns(min)
423 *-----
424 * Argument  : void
425 *-----
426 * Return Value : void
427 *-----
428 * Note      : システムに応じてウェイト時間を変更してください。
429 *"FUNC COMMENT END"*****/
430 void _phy_ta_z0( void )
431 {
432     int32_t j;
433
434     for(j = MDC_WAIT; j > 0; j--){
435         EtherC.PIR.LONG = 0x00000000;
436     }
437     for(j = MDC_WAIT; j > 0; j--){
438         EtherC.PIR.LONG = 0x00000001;
439     }
440     for(j = MDC_WAIT; j > 0; j--){
441         EtherC.PIR.LONG = 0x00000001;
442     }
443     for(j = MDC_WAIT; j > 0; j--){
444         EtherC.PIR.LONG = 0x00000000;
445     }
446 }
447
```

## 3.14 サンプルプログラムリスト "phy.c" (11)

```
448 /*"FUNC COMMENT"*****
449 * ID      :
450 * Outline : PHY モジュールレジスタアクセス TA(10)ビット出力
451 *-----
452 * Include :
453 *-----
454 * Declaration : void _phy_ta_10(void);
455 *-----
456 * Description : PHY モジュール MII 管理インタフェースへ 1,0 を出力します。
457 *-----
458 * Argument   : void
459 *-----
460 * Return Value : void
461 *-----
462 * Note       : None
463 *"FUNC COMMENT END"*****/
464 void _phy_ta_10(void)
465 {
466     _phy_mii_write_1();
467     _phy_mii_write_0();
468 }
469
```

## 3.15 サンプルプログラムリスト "phy.c" (12)

```
470 /*"FUNC COMMENT"*****
471 * ID      :
472 * Outline : PHY モジュールレジスタアクセス 1 ビット(1)出力
473 *-----
474 * Include :
475 *-----
476 * Declaration : void _phy_mii_write_1( void );
477 *-----
478 * Description : PHY モジュール MII 管理インタフェースに 1 を出力します。
479 *             : 下記条件を満たすように信号を出力します。
480 *             : ・MDC High パルス幅      : 160ns(min)
481 *             : ・MDC Low パルス幅      : 160ns(min)
482 *             : ・MDC サイクル時間     : 400ns(min)
483 *             : ・MDIO セットアップ時間 : 10ns(min)
484 *             : ・MDIO ホールド時間    : 10ns(min)
485 *-----
486 * Argument  : void
487 *-----
488 * Return Value : void
489 *-----
490 * Note      : システムに応じてウェイト時間を変更してください。
491 *"FUNC COMMENT END"*****/
492 void _phy_mii_write_1( void )
493 {
494     int32_t j;
495
496     for(j = MDC_WAIT; j > 0; j--){
497         EtherC.PIR.LONG = 0x00000006;
498     }
499     for(j = MDC_WAIT; j > 0; j--){
500         EtherC.PIR.LONG = 0x00000007;
501     }
502     for(j = MDC_WAIT; j > 0; j--){
503         EtherC.PIR.LONG = 0x00000007;
504     }
505     for(j = MDC_WAIT; j > 0; j--){
506         EtherC.PIR.LONG = 0x00000006;
507     }
508 }
509
```

## 3.16 サンプルプログラムリスト "phy.c" (13)

```
510  /*"FUNC COMMENT"*****
511  * ID      :
512  * Outline  : PHY モジュールレジスタアクセス 1 ビット(0)出力
513  *-----
514  * Include  :
515  *-----
516  * Declaration : void _phy_mii_write_0( void );
517  *-----
518  * Description : PHY モジュール MII 管理インタフェースに 0 を出力します。
519  *             : 下記条件を満たすように信号を出力します。
520  *             : ・MDC High パルス幅      : 160ns(min)
521  *             : ・MDC Low パルス幅      : 160ns(min)
522  *             : ・MDC サイクル時間     : 400ns(min)
523  *             : ・MDIO セットアップ時間 : 10ns(min)
524  *             : ・MDIO ホールド時間    : 10ns(min)
525  *-----
526  * Argument  : void
527  *-----
528  * Return Value : void
529  *-----
530  * Note      : システムに応じてウェイト時間を変更してください。
531  *"FUNC COMMENT END"*****/
532 void _phy_mii_write_0( void )
533 {
534     int32_t j;
535
536     for(j = MDC_WAIT; j > 0; j--){
537         EtherC.PIR.LONG = 0x00000002;
538     }
539     for(j = MDC_WAIT; j > 0; j--){
540         EtherC.PIR.LONG = 0x00000003;
541     }
542     for(j = MDC_WAIT; j > 0; j--){
543         EtherC.PIR.LONG = 0x00000003;
544     }
545     for(j = MDC_WAIT; j > 0; j--){
546         EtherC.PIR.LONG = 0x00000002;
547     }
548 }
549
550 /* End of File */
551
```

## 3.17 サンプルプログラムリスト "phy.h" (1)

```
1  /*****
2  *  DISCLAIMER
3  *
4  *  This software is supplied by Renesas Electronics Corp. and is only
5  *  intended for use with Renesas products. No other uses are authorized.
6  *
7  *  This software is owned by Renesas Electronics 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 *  ELECTRONICS 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 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) 2010 Renesas Electronics Corporation. All Rights Reserved.
29 *  "FILE COMMENT" ***** Technical reference data *****
30 *  System Name : SH7216 Sample Program
31 *  File Name   : phy.h
32 *  Version    : 2.00.00
33 *  Device     : SH7216
34 *  Tool-Chain : High-performance Embedded Workshop (Ver.4.07.00).
35 *             : C/C++ compiler package for the SuperH RISC engine family
36 *             : (Ver.9.03 Release00).
37 *  OS        : None
38 *  H/W Platform: R0K572167 (CPU board)
39 *  Description : Ethernet PHY device driver
40 *****/
41 *  History    : Jun.10.2009 Ver.1.00.00
42 *             : Jul.23,2010 Ver.2.00.00 ルネサス標準APIに対応
43 *  "FILE COMMENT END" *****/
44 #ifndef PHY_H
45 #define PHY_H
46
```

## 3.18 サンプルプログラムリスト "phy.h" (2)

```
47  /* ==== Macro definition ==== */
48  /* ---- Standard PHY Registers ---- */
49  #define BASIC_MODE_CONTROL_REG      0
50  #define BASIC_MODE_STATUS_REG       1
51  #define PHY_IDENTIFIER1_REG         2
52  #define PHY_IDENTIFIER2_REG         3
53  #define AN_ADVERTISEMENT_REG        4
54  #define AN_LINK_PARTNER_ABILITY_REG  5
55  #define AN_EXPANSION_REG             6
56
57  /* ---- Media Independent Interface ---- */
58  #define PHY_ST      1
59  #define PHY_READ    2
60  #define PHY_WRITE   1
61  #define PHY_ADDR    1
62
63  #define MDC_WAIT    3
64
65  /* ---- PHY return definitions ---- */
66  #define R_PHY_OK    0
67  #define R_PHY_ERROR -1
68
69  /* ==== Prototype declaration ==== */
70  /* ----External prototypes ---- */
71  int16_t phy_init( void );
72  void     phy_set_100full( void );
73  void     phy_set_10half( void );
74  int16_t phy_set_autonegotiate( void );
75
76  #endif /* PHY_H */
```

#### 4. 参考ドキュメント

- ソフトウェアマニュアル  
SH-2A, SH2A-FPU ソフトウェアマニュアル Rev.3.00  
(最新版をルネサス エレクトロニクス ホームページから入手してください。)
- ハードウェアマニュアル  
SH7214 グループ、SH7216 グループ ユーザーズマニュアル ハードウェア編 Rev.2.00  
(最新版をルネサス エレクトロニクス ホームページから入手してください。)

## ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

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

お問い合わせ先

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

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。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.04.21	—	初版発行
2.00	2010.07.30	全頁	サンプルプログラムの API 変更に伴い、内容を修正。

## 製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

### 1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

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

### 2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

### 3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

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

### 4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

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

### 5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違くと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

## 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て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きを行ってください。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。



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

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

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

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。  
総合お問合せ窓口：<http://japan.renesas.com/inquiry>