サイプレスはインフィニオン テクノロジーズになりました

この表紙に続く文書には「サイプレス」と表記されていますが、これは同社が最初にこの製品を開発したからです。新規および既存のお客様いずれに対しても、引き続きインフィニオンがラインアップの一部として当該製品をご提供いたします。

文書の内容の継続性

下記製品がインフィニオンの製品ラインアップの一部として提供されたとしても、それを理由としてこの文書に変更が加わることはありません。今後も適宜改訂は行いますが、変更があった場合は文書の履歴ページでお知らせします。

注文時の部品番号の継続性

インフィニオンは既存の部品番号を引き続きサポートします。ご注文の際は、データシート記載の注文部品番号をこれまで通りご利用下さい。

32 ビット・マイクロコントローラ FM4 ファミリ
Peripheral Manual Ethernet 編

Doc. No. 002-04964 Rev.*C

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

Copyrights

© Cypress Semiconductor Corporation, 2014-2018. 本書面は、Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社（以下、「Cypress」という。）に帰属する財産である。本書面（本書面に含まれ又は言及されているあらゆるソフトウェア又はファームウェア（以下、「本ソフトウェア」という。）を含む）は、アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき、Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し、また、本段落で特に記載されているものを除き、Cypress の特許権、著作権、商標権又はその他の知的財産権のライセンスを一切許諾していない。本ソフトウェアにライセンス契約書が伴っておらず、かつ、あなたが Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意をしていない場合、Cypress は、あなたに対して、(1) 本ソフトウェアの著作権に基づき、(a) ソースコード形式で提供されている本ソフトウェアについて、Cypress ハードウェア製品と共に用いるためにのみ、組織内部でのみ、本ソフトウェアの修正及び複製を行うこと、並びに (b) Cypress のハードウェア製品ユニットに用いるためにのみ、(直接又は再販売者及び販売代理店を介して間接のいずれかで) エンドユーザーに対して、バイナリーコード形式で本ソフトウェアを外部に配布すること、並びに (2) 本ソフトウェア (Cypress により提供され、修正がなされていないもの) に抵触する Cypress の特許権のクレームに基づき、Cypress ハードウェア製品と共に用いるためにのみ、本ソフトウェアの作成、利用、配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス (サブライセンスの権利を除く) を付与する。本ソフトウェアのその他の使用、複製、修正、変換又はコンパイルを禁止する。

適用される法律により許される範囲内で，Cypress は，本書面又はいかなる本ソフトウェア若しくはこれに伴うハードウェアに関しても，明示又は黙示をとわず，いかなる保証（商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない）も行わない。いかなるコンピューティングデバイスも絶対に安全ということはない。従って，Cypress のハードウェアまたはソフトウェア製品に講じられたセキュリティ対策にもかかわらず，Cypress は，Cypress 製品への権限のないアクセスまたは使用といったセキュリティ違反から生じる一切の責任を負わない。加えて，本書面に記載された製品には，エラッタと呼ばれる設計上の欠陥またはエラーが含まれている可能性があり，公表された仕様とは異なる動作をする場合がある。適用される法律により許される範囲内で，Cypress は，別途通知することなく，本書面を変更する権利を留保する。Cypress は，本書面に記載のある，いかなる製品若しくは回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報（あらゆるサンプルデザイン情報又はプログラムコードを含む）は，参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計，プログラム，かつテストすることは，本書面のユーザーの責任において行われるものとする。Cypress 製品は，兵器，兵器システム，原子力施設，生命維持装置若しくは生命維持システム，蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム，汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分としての使用，又は装置若しくはシステムの不具合が人身傷害，死亡若しくは物的損害を生じさせるようなその他の使用（以下「本目的外使用」という。）のためには設計，意図又は承認されていない。重要な構成部分とは，それの不具合が装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できるような装置若しくはシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ，若しくは本目的外使用に関連するいかなる請求，損害又はその他の責任についても，Cypress はその全部又は一部をとわず一切の責任を負わず，かつ Cypress はそれら一切から本書により免除される。Cypress は Cypress 製品の本目的外使用から生じ又は本目的外使用に関連するあらゆる請求，費用，損害及びその他の責任（人身傷害又は死亡に基づく請求を含む）から免責補償される。

Cypress、Cypress のロゴ、Spansion、Spansion のロゴ及びこれらの組み合わせ、WICED、PSoC、CapsSense、EZ-USB、F-RAM、及び Traveo は、米国及びその他の国における Cypress の商標又は登録商標である。Cypress の商標のより完全なリストは、cypress.com を参照のこと。その他の名称及びブランドは、それぞれの権利者の財産として権利主張がなされている可能性がある。

Arm and Cortex are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

はじめに

Cypress (サイプレス) 製品につきまして、平素より格別のご愛顧を賜り厚くお礼申し上げます。
本ファミリをご利用になる前に、『ペリフェラルマニュアル』およびご使用する製品の『データシート』をご一読ください。
なお本書は、ペリフェラルマニュアルから Ethernet に関する内容を抜きだした別冊として定義しております。

本書の目的と対象読者

本書は、実際に本ファミリを使用して製品を開発される技術者を対象に、本ファミリの機能や動作, 使い方について解説しています。

＜注意事項＞
本マニュアルは周辺機能の構成および動作を説明するものであり、各デバイスの仕様を説明するものではありません。
デバイス仕様の詳細については、それぞれのデータシートを参照してください。
周辺機能の搭載有無については各々のデバイスにより異なります。搭載有無については各デバイスのデータシートを参照ください。

サンプルプログラムおよび開発環境

FM4 ファミリの周辺機能を動作させるためのサンプルプログラムを無償で提供しております。また、本ファミリで使用する開発環境も掲載しています。当社マイコンの動作仕様や使用方法の確認などにお役立てください。

マイコンサポート情報

https://community.cypress.com/community/MCU

＜注意事項＞
サンプルプログラムは、予告なしに変更することがあります。また、サンプルプログラムは標準的な動作や使い方を示したものですので、お客様のシステム上でご使用の際は十分評価された上でご使用ください。また、サンプルプログラムの使用に起因し生じた損害については、当社は一切その責任を負いません。

関連マニュアル

本ファミリに関連するマニュアルを示します。状況に応じて必要なマニュアルを参照してください。
本書に記載したマニュアルの内容は予告なく変更することがあります。最新版をお問い合わせください。

ペリフェラルマニュアル

FM4 ファミリ ペリフェラルマニュアル (002-04857)
　以降、『ペリフェラルマニュアル』とよびます。
FM4 ファミリ ペリフェラルマニュアル タイマ編 (002-04859)
　以降、『タイマ編』とよびます。
FM4 ファミリ ペリフェラルマニュアル アナログマクロ編 (002-4861)
　以降、『アナログマクロ編』とよびます。
FM4 ファミリ ペリフェラルマニュアル 通信マクロ編 (002-04904)
　以降、『通信マクロ編』とよびます。
FM4 ファミリ ペリフェラルマニュアル Ethernet 編 (本書)
　以降、『Ethernet 編』とよびます。

データシート

デバイス仕様, 電気的特性, 外形寸法, オーダ型格などの詳細は以下を参照してください。

32 ビット FM4 ファミリ データシート

<注意事項>
データシートはシリーズごとに用意されています。
ご使用する製品のデータシートを参照してください。

CPU プログラミングマニュアル

Arm Cortex-M4F コアの詳細は http://www.arm.com/ から入手できる以下を参照してください。

Cortex-M4 テクニカルリファレンスマニュアル
Arm v7-M アーキテクチャ アプリケーション レベル リファレンス マニュアル

フラッシュプログラミングマニュアル

内蔵されているフラッシュメモリの機能や動作の詳細は以下を参照してください。

FM4 ファミリ フラッシュプログラミングマニュアル

<注意事項>
フラッシュプログラミングマニュアルはシリーズごとに用意されています。
ご使用する製品のフラッシュプログラミングマニュアルを参照してください。

本書の使い方

機能の探し方

本書では次の方法で、使いたい機能の説明を探すことができます。

目次から探す
本書の内容を記載順に示します。

レジスタから探す
本文中では各レジスタの配置アドレスを記載しておりません。各レジスタのアドレスを確認するときは『FM4 ファミリ ペリフェラルマニュアル』の『Appendixes　A.レジスタマップ』を参照してください。

章について

本書では、Ethernet について説明しています。

用語について

本書で使用している用語について示します。

用語	説明
ワード	32 ビット単位でのアクセスを指します。
ハーフワード	16 ビット単位でのアクセスを指します。
バイト	8 ビット単位でのアクセスを指します。

表記について

■ 本書のレジスタ説明中のビット構成図では以下のように表記しています。
- bit:　ビット番号
- Field:　ビットフィールド名
- 属性:　各ビットのリード、ライト属性
  - ・R:　リードオンリ
  - ・W:　ライトオンリ
  - ・R/W:　リード・ライト可能
  - ・-:　未定義
- 初期値:　リセット直後のレジスタ初期値
  - ・0:　初期値"0"
  - ・1:　初期値"1"
  - ・X:　初期値不定

■ 本書では、複数のビットを以下のように表記しています。
- 例: bit7 から bit0 の場合は bit7:0

■ 本書では、アドレスなどの数値を以下のように表記しています。
- 16 進数:　プレフィックス(接頭辞)として"0x"を付けて表記しています(例 ：0xFFFF)。
- 2 進数:　プレフィックス(接頭辞)として"0b"を付けて表記しています(例 ：0b1111)。
- 10 進数:　数値だけで表記しています(例 ：1000)。

本マニュアルにおける対象製品

Table 1 対象製品一覧(TYPE3-M4 製品)

タイプ名*	フラッシュメモリサイズ		
	2 Mbytes	1.5 Mbytes	1 Mbytes
TYPE3-M4	S6E2CCAL S6E2CCAJ S6E2CCAH	S6E2CC9L S6E2CC9J S6E2CC9H	S6E2CC8L S6E2CC8J S6E2CC8H
	S6E2C2AL S6E2C2AJ S6E2C2AH	S6E2C29L S6E2C29J S6E2C29H	S6E2C28L S6E2C28J S6E2C28H

*: FM4 ファミリペリフェラルマニュアルにおいて製品を分類するために使用している表記です。

Table 2 対象製品一覧(TYPE5-M4 製品)

タイプ名*	フラッシュメモリサイズ	
	1 Mbytes	512Kbytes
TYPE5-M4	S6E2GM8J S6E2GM8H	S6E2GM6J S6E2GM6H
	S6E2GK8J S6E2GK8H	S6E2GK6J S6E2GK6H
	S6E2G28J S6E2G28H	S6E2G26J S6E2G26H

*: FM4 ファミリペリフェラルマニュアルにおいて製品を分類するために使用している表記です。

Contents

CHAPTER 1: Ethernet 11
1. 概要 12
1.1. 概要 12
1.2. ブロックダイヤグラム 12
1.3. 各ブロックの説明 13
2. Ethernet の入出力信号 15
2.1. マイコン外部端子 15
2.2. 内部接続端子 18
3. Ethernet-MAC セットアップ制御手順 19
4. Ethernet システム制御部レジスタ 21
4.1. モード選択レジスタ (ETH_MODE) 22
4.2. クロックゲーティングレジスタ (ETH_CLKG) 24
CHAPTER 2: Ethernet-MAC 25
1. 概要 26
2. ブロック構成 27
3. アーキテクチャ 28
3.1. 端子機能 29
3.2. AHB アプリケーション・ホスト・インタフェース 31
3.3. DMA コントローラ 32
3.3.1. ディスクリプタ構造 32
3.3.2. DMA コントローラの初期化 33
3.3.3. 送信動作 35
3.3.4. 受診動作 39
3.3.5. 割込み 43
3.3.6. DMA へのエラー応答 43
3.3.7. CRC 43
3.4. チェックサムエンジン 44
3.4.1. 送信チェックサムオフロードエンジン 44
3.4.2. 受診チェックサムオフロードエンジン 45
3.5. Energy Efficient Ethernet 47
3.6. MAC マネジメントカウンタ 50
3.7. Station Management Agent 51
3.8. IEEE1588 53
4. レジスタ 61
4.1. GMAC Register 0 (MCR) 67
4.2. GMAC Register 1 (MFFR) 70
4.3. GMAC Register 2, 3 (MHTRH, MHTRL) 73
4.4. GMAC Register 4 (GAR) 74
4.5. GMAC Register 5 (GDR) 76
4.6. GMAC Register 6 (FCR) 77
4.7. GMAC Register 7 (VTR) 79
4.8. GMAC Register 10 (RWFFR) 80
4.9. GMAC Register 11 (PMTR) 82
4.10. GMAC Register 12 (LPICSR) 84
4.11. GMAC Register 13 (LPITCR) 86

4.12. GMAC Register 14 (ISR) 87
4.13. GMAC Register 15 (IMR) 89
4.14. GMAC Register 16 (MAR0H) 90
4.15. GMAC Register 17 (MAR0L) 91
4.16. GMAC Register 18,20,22,..., 542)(MAR1H, 2H, 3H,..., 31H) 92
4.17. GMAC Register 19,21,23,..., 543 (MAR1L, 2L, 3L,..., 31L) 94
4.18. GMAC Register 54 (RGSR) 95
4.19. GMAC Register 448 (TSCR) 96
4.20. GMAC Register 449 (SSIR) 99
4.21. GMAC Register 450 (STSR) 100
4.22. GMAC Register 451 (STNR) 101
4.23. GMAC Register 452 (STSUR) 102
4.24. GMAC Register 453 (STNUR) 103
4.25. GMAC Register 454 (TSAR) 104
4.26. GMAC Register 455 (TTSR) 105
4.27. GMAC Register 456 (TTNR) 106
4.28. GMAC Register 457 (STHWSR) 107
4.29. GMAC Register 458 (TSR) 108
4.30. GMAC Register 459 (PPSCR) 109
4.31. GMAC Register 460 (ATNR) 110
4.32. GMAC Register 461 (ATSR) 111
4.33. DMA Register 0 (BMR) 112
4.34. DMA Register 1 (TPDR) 114
4.35. DMA Register 2 (RPDR) 115
4.36. DMA Register 3 (RDLAR) 116
4.37. DMA Register 4 (TDLAR) 117
4.38. DMA Register 5 (SR) 118
4.39. DMA Register 6 (OMR) 122
4.40. DMA Register 7 (IER) 125
4.41. DMA Register 8 (MFBOCR) 128
4.42. DMA Register 9 (RIWTR) 129
4.43. DMA Register 11 (AHBSR) 130
4.44. DMA Register 18 (CHTDR) 131
4.45. DMA Register 19 (CHRDR) 132
4.46. DMA Register 20 (CHTBAR) 133
4.47. DMA Register 21 (CHRBAR) 134
4.48. MMC Register list 135
4.49. GMAC Register.64 (mmc_cntl) 139
4.50. GMAC Register.65 (mmc_intr_rx) 140
4.51. GMAC Register.66 (mmc_intr_tx) 141
4.52. GMAC Register.67 (mmc_intr_mask_rx) 142
4.53. GMAC Register.68 (mmc_intr_mask_tx) 143
4.54. GMAC Register.128 (mmc_ipc_intr_mask_rx) 144
4.55. GMAC Register.130 (mmc_ipc_intr_rx) 145
5. ディスクリプタ 146
5.1. 送信拡張ディスクリプタ 147
5.1.1. Transmit Enhanced Descriptor 0 (TDES0) 148
5.1.2. Transmit Enhanced Descriptor 1 (TDES1) 152
5.1.3. Transmit Enhanced Descriptor 2 (TDES2) 153
5.1.4. Transmit Enhanced Descriptor 3 (TDES3) 154
5.1.5. Transmit Enhanced Descriptor 6 (TDES6) 155
5.1.6. Transmit Enhanced Descriptor 7 (TDES7) 156

5.2. 受信拡張ディスクリプタ 157
5.2.1. Receive Enhanced Descriptor 0 (RDES0) 158
5.2.2. Receive Enhanced Descriptor 1 (RDES1) 160
5.2.3. Receive Enhanced Descriptor 2 (RDES2) 161
5.2.4. Receive Enhanced Descriptor 3 (RDES3) 162
5.2.5. Receive Enhanced Descriptor 4 (RDES4) 163
5.2.6. Receive Enhanced Descriptor 6 (RDES6) 165
5.2.7. Receive Enhanced Descriptor 7 (RDES7) 166
6. プログラミングガイド 167
6.1. DMA の初期化・ディスプリクタ 167
6.2. GMAC の初期化 168
6.3. 通常の受信および送信動作 169
6.4. 動作の停止および開始 169
6.5. リンクの停止/確立シーケンス 169
6.6. IEEE タイムスタンプ生成のプログラミング・ガイドライン 170
6.7. Energy Efficient Ethernet のプログラミング・ガイドライン 171
6.8. スタンバイモードへの移行と SYS_CLK の停止について 171
Appendixes 172
1. 主な変更内容 172
改訂履歴 173

CHAPTER 1: Ethernet

Ethernet 関連ブロックの構成について説明します。

1. 概要
2. Ethernet の入出力信号
3. Ethernet-MAC セットアップ制御手順
4. Ethernet システム制御部レジスタ

1. 概要

Ethernet 機能の概要と Ethernet-MAC セットアップ制御手順について説明します。

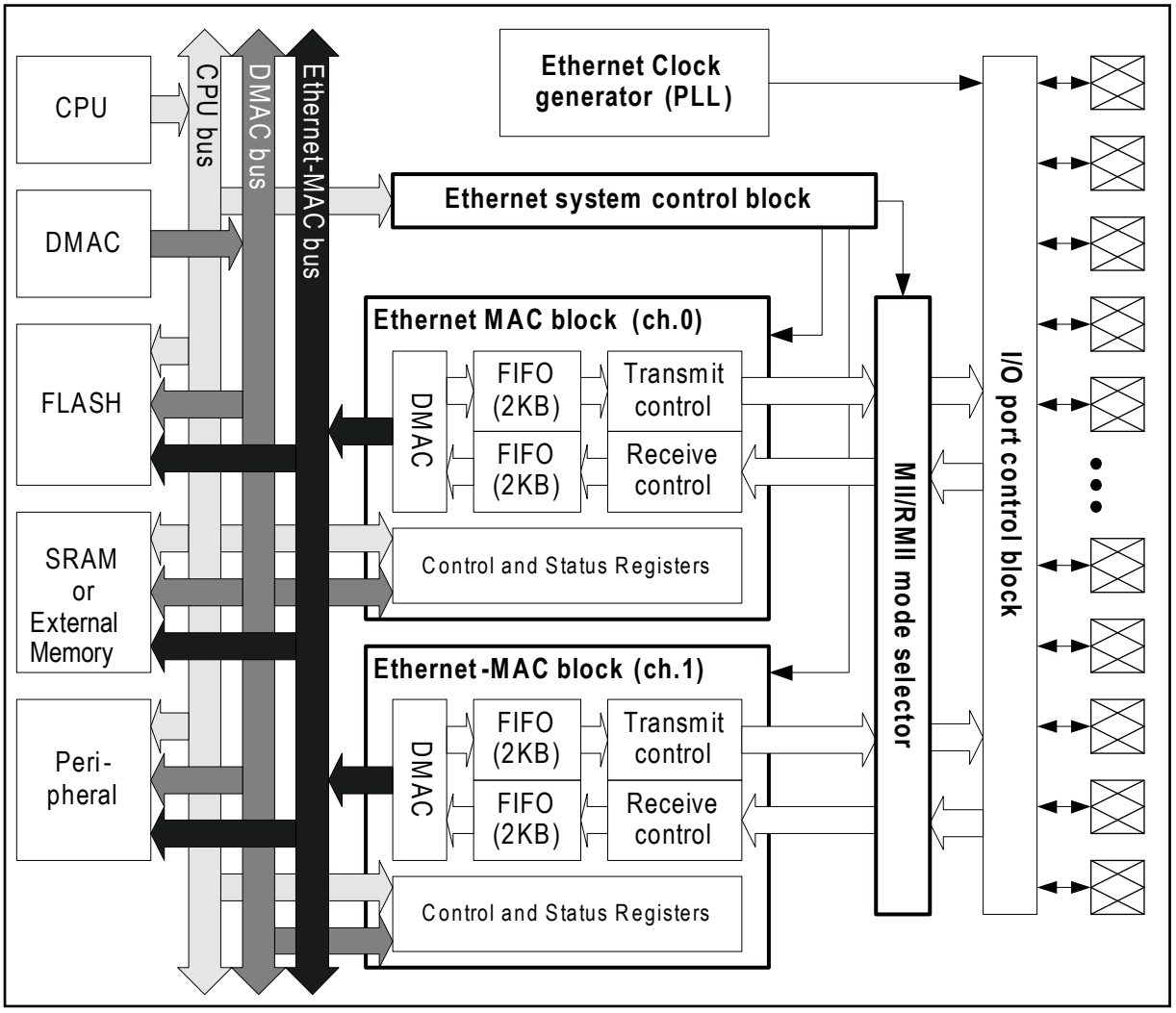
1.1 概要

本ファミリの Ethernet 機能は、Ethernet-MAC 部と、その周辺回路から構成されます。本章では、Ethernet-MAC 周辺回路の説明と、Ethernet-MAC セットアップ制御手順について説明します。

1.2 ブロックダイヤグラム

Ethernet-MAC と周辺回路の構成図を Figure 1-1 に示します。

Figure 1-1 Ethernet-MAC と周辺回路の構成図



1.3 各ブロックの説明

Ethernet-MAC 部

Ethernet-MAC は、Ethernet 通信プロトコル制御を行うブロックです。ブロックダイヤグラムには ch.0 と ch.1 の 2 個の Ethernet-MAC が記載されていますが、ch.0 の 1 個のみ搭載している品種と、ch.0 と ch.1 の 2 個搭載している品種があります。1 個搭載品種の場合、ch.1 側は使用できません。2 個搭載品種は、両方の Ethernet-MAC を独立して使用できます。
各 Ethernet-MAC は、送信制御部, 受信制御部, FIFO メモリ, 専用 DMAC から構成されています。CPU からの制御レジスタ部 (Control and Status Registers) への指示により動作の制御を行います。専用 DMAC は、CPU により形成されたメモリ上のディスクリプタを使用し、送受信データの転送処理を行います。Ethernet-MAC に関する詳細は、『Ethernet-MAC』の章を参照してください。
各 Ethernet-MAC を動作させる場合、あらかじめ、Ethernet システム制御部へのレジスタ書込みにより、以下の手順で Ethernet-MAC セットアップをしてください。

1. 各 Ethernet-MAC に対するクロック供給を開始する。
2. MII/RMII モードを選択する。
3. 各 Ethernet-MAC へハードウェアリセットを発行し、Ethernet-MAC を再起動する。

上記については、「3. Ethernet-MAC セットアップ制御手順」を参照してください。
各 Ethernet-MAC は、上記のセットアップのハードウェアリセット解除後に、初期設定を行い、動作開始を指示します。各 Ethernet-MAC の内部レジスタには、MII/RMII モードの選択を行う箇所がありますが、これらは上記セットアップ手順とは別に設定を行います。
Figure 1-1 に示すように、Ethernet-MAC のうち DMAC は、CPU およびマイコンの DMAC とは独立した専用システムバス (AHB) を持っており、CPU, マイコン DMAC と同時動作が可能です。

Ethernet-システム制御部 (Ethernet system control block)

Ethernet システム制御部は、Ethernet-MAC の周辺回路の制御を行い、各 Ethernet-MAC のセットアップ制御を行います。「3. Ethernet-MAC セットアップ制御手順」に、セットアップ制御手順を記載しています。また、「4. Ethernet システム制御部レジスタ」に、Ethernet システム制御部のレジスタ機能を記載しています。
各 Ethernet-MAC を動作させない場合、クロック信号の供給を停止しておくことで、マイコンの低消費電力化を行うことができます。

MII/RMII モード選択部 (MII/RMII mode selector)

MII/RMII モード選択部は、Ethernet-MAC の外部 PHY インタフェース信号を I/O ポート制御部と接続するセレクタブロックです。Ethernet システム制御部のレジスタ設定に基づき、使用するインタフェース (MII/RMII) モードを選択します。

I/O port 制御部 (I/O port control block)

マイコンの外部端子は、GPIO など, Ethernet 以外の周辺機能ブロックの端子と端子機能が兼用されています。マイコンの外部端子を Ethernet の外部 PHY インタフェース端子として使用するために、I/O ポート部の制御レジスタの設定が必要になります。
マイコンが低消費電力モード (STOP モード, タイマモード) の状態にあるとき、外部 PHY からの Wakeup パケットを受信するためには、PHY インタフェース端子の入出力信号を有効とする状態 (直前保持状態とよびます) にしておく必要があります。
I/O ポート制御部に関する詳細は、『ペリフェラルマニュアル』の『I/O ポート』の章を参照してください。

Ethernet クロック生成部 (Ethernet clock generator block)

Ethernet クロック生成部から出力されるクロック信号を、マイコン外部端子へ出力できます。このクロック信号出力を一般的な外部 PHY デバイスの入力クロックに使う場合、PLL を使用せず、メイン発振器の出力を Ethernet クロック生成部でスルーして供給するようにしてください。PLL を使用してクロック出力信号を生成した場合、一般的な PHY デバイスに要求される入力クロック発振精度を保証できません。Ethernet クロック生成部に関する詳細は、『FM3 ファミリ ペリフェラルマニュアル 通信マクロ編』の『USB/Ethernet クロック生成部』の章を参照してください。

2. Ethernet の入出力信号

Ethernet 機能に関連する入出力信号の接続を説明します。

2.1 マイコン外部端子

外部 PHY インタフェース信号

外部端子は、Ethernet-MAC の ch.0, ch.1 および MII/RMII の PHY インタフェース信号端子の機能が兼用されています。Ethernet システム制御部のセットアップ制御時に、どのインタフェース信号を用いるかが決定されます。Ethernet-MAC 1 個搭載品と、2 個搭載品のそれぞれの外部端子名と各 Ethernet-MAC の PHY インタフェース信号の対応関係の一覧を Table 2-1 に示します。

Table 2-1 外部端子名と PHY インタフェース信号の対応関係

外部端子名		MII (ch.0 使用)	RMII (ch.0 使用)	RMII (ch.0, ch.1 使用)	備考
ch.0 のみ 搭載品種	ch.0, ch.1 搭載品種				
E_RXCK_REFCK	E_RXCK0_REFCK	ch.0 RX_CLK	ch.0 REF_CLK	ch.0 REF_CLK ch.1 REF_CLK	*1
E_RX00	E_RX00	ch.0 RXD[0]	ch.0 RXD[0]	ch.0 RXD[0]	
E_RX01	E_RX01	ch.0 RXD[1]	ch.0 RXD[1]	ch.0 RXD[1]	
E_RX02	E_RX02_RX10	ch.0 RXD[2]	使用しない	ch.1 RXD[0]	
E_RX03	E_RX03_RX11	ch.0 RXD[3]	使用しない	ch.1 RXD[1]	
E_RXDV	E_RXDV0	ch.0 RX_DV	ch.0 CRS_DV	ch.0 CRS_DV	
E_RXER	E_RXER0_RXDV1	ch.0 RX_ER	使用しない	ch.1 CRS_DV	*2
E_TCK	E_TCK0_MDC1	ch.0 TX_CLK	使用しない	ch.1 MDC	
E_TX00	E_TX00	ch.0 TXD[0]	ch.0 TXD[0]	ch.0 TXD[0]	
E_TX01	E_TX01	ch.0 TXD[1]	ch.0 TXD[1]	ch.0 TXD[1]	
E_TX02	E_TX02_TX10	ch.0 TXD[2]	使用しない	ch.1 TXD[0]	
E_TX03	E_TX03_TX11	ch.0 TXD[3]	使用しない	ch.1 TXD[1]	
E_TXEN	E_TXEN0	ch.0 TX_EN	ch.0 TX_EN	ch.0 TX_EN	
E_TXER	E_TXER0_TXEN1	ch.0 TX_ER	使用しない	ch.1 TX_EN	*3
E_CRS	E_CRS0	ch.0 CRS	使用しない	使用しない	
E_COL	E_COL0	ch.0 COL	使用しない	使用しない	
E_MDC	E_MDC0	ch.0 MDC	ch.0 MDC	ch.0 MDC	
E_MDIO	E_MDIO0	ch.0 MDIO	ch.0 MDIO	ch.0 MDIO	
-	E_MDIO1	使用しない	使用しない	ch.1 MDIO	

*1: この端子は、MII の場合、RX_CLK 入力として使用します。RMII の場合、REF_CLK 入力として使用します。Ethernet-MAC を ch.0, ch.1 の両方を使用する場合、両チャネル共通の入力です。

*2: RMII の場合、外部 PHY からの RX_ER 入力は使用しないため、接続する必要はありません。

*3: MII の場合、TX_ER 出力は、EEE (Energy Efficient Ethernet) における LPI モード時のみ"H"レベル出力を行います。EEE を使用しない場合、PHY との接続は不要です。

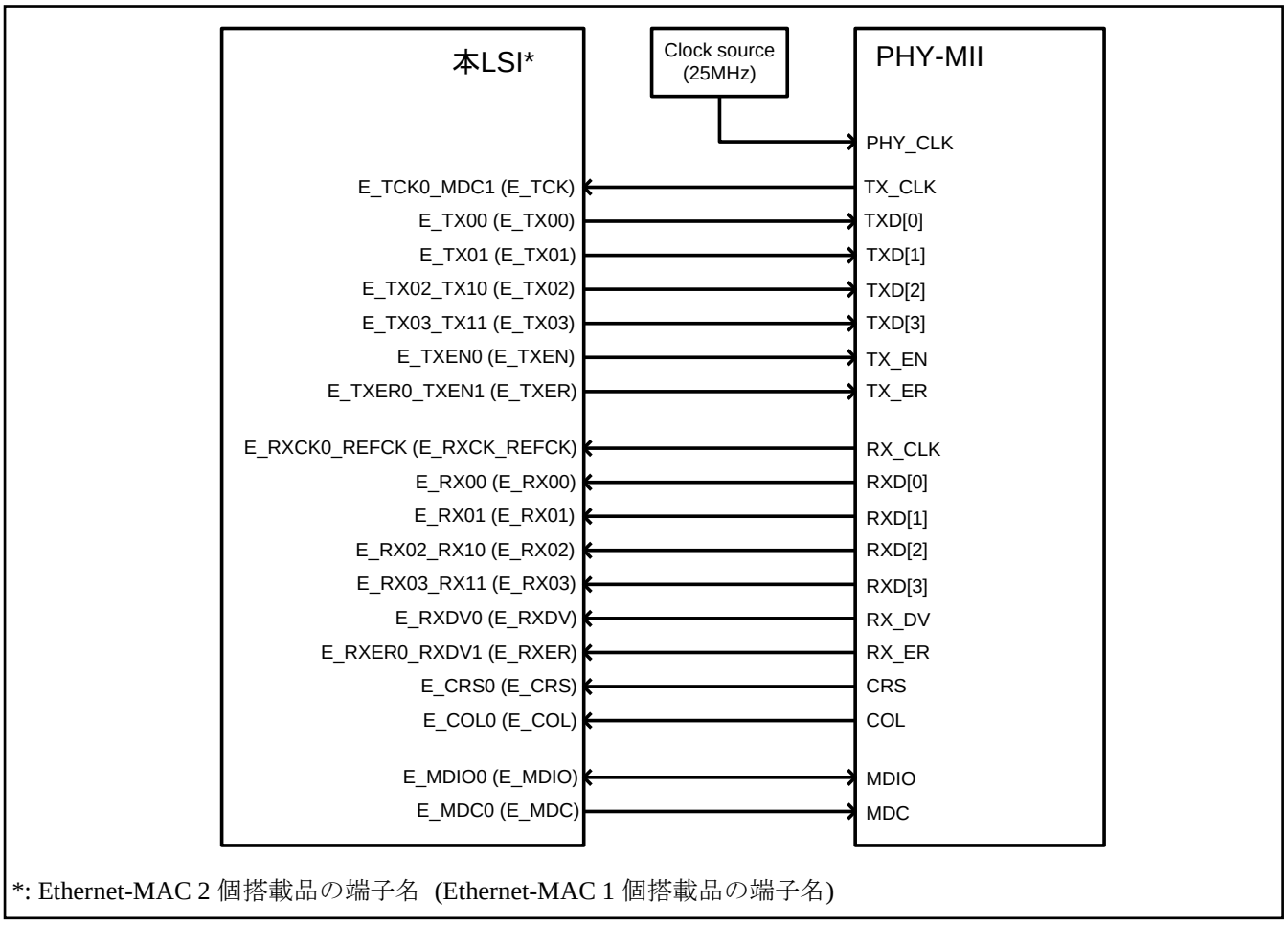
Ethernet-MAC の ch.0, ch.1 両方を使用する場合、MII は選択できません。

外部 PHY 接続例

以下に外部 PHY デバイスとの接続例を示します。
Figure 2-1～Figure 2-3 に記載していませんが、MDIO 信号はプルアップすることを推奨します。

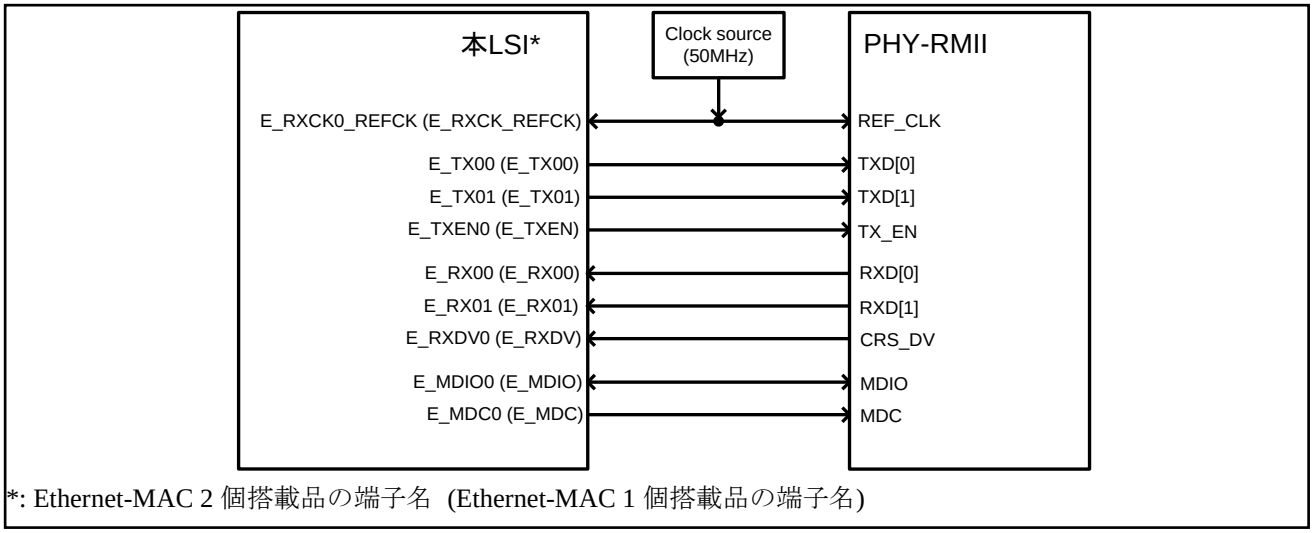
Ethernet-MAC (ch.0)を使用し、MII モードで PHY と接続 (Figure 2-1)
Ethernet-MAC (ch.0)を使用し、RMII モードで PHY と接続 (Figure 2-2)
Ethernet-MAC (ch.0, ch.1)を使用し、RMII モードで 2 個の PHY と接続 (Figure 2-3)

Figure 2-1 MII モード接続図 (Ethernet-MAC ch.0 を使用)



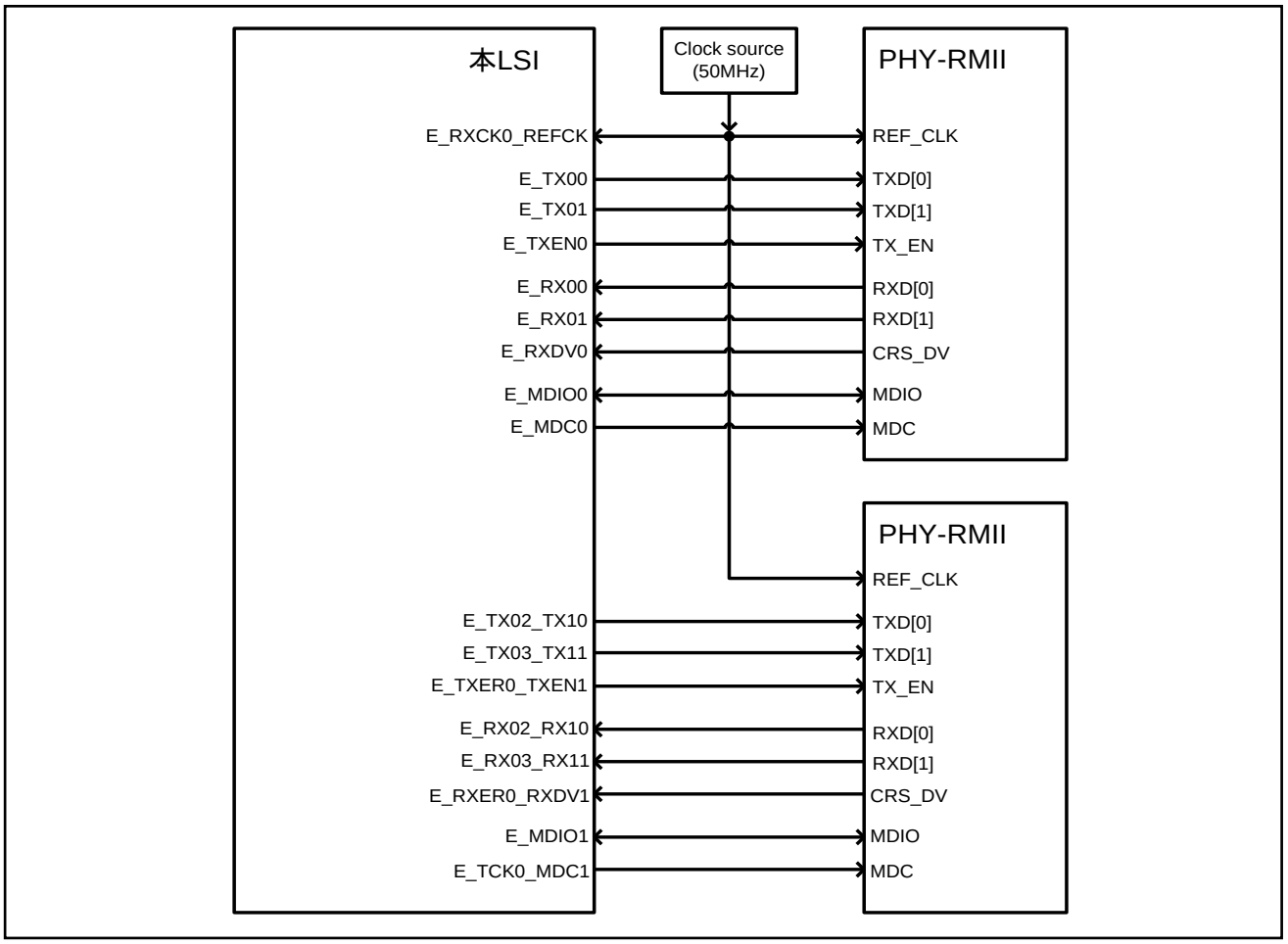
*: Ethernet-MAC 2 個搭載品の端子名 (Ethernet-MAC 1 個搭載品の端子名)

Figure 2-2 RMII モード接続図 (Ethernet-MAC ch.0 を使用)



*: Ethernet-MAC 2 個搭載品の端子名 (Ethernet-MAC 1 個搭載品の端子名)

Figure 2-3 RMII モード接続図 (Ethernet-MAC ch.0, ch.1 を使用)



その他の外部端子

Table 2-1 に示す以外で、Ethernet 機能に関連する外部端子を Table 2-2 に示します。

Table 2-2 その他の外部端子

外部端子名	機能
E_PPS0_PPS1 *	Ethernet-MAC 部から、PTP 機能のシステム・タイム・カウンタの 1 秒ごとのパルスを出力します。ETH_MODE.PPSSEL レジスタにて、ch.0, ch.1 のどちらの出力を選択するか決定できます。
E_COUT	Ethernet クロック生成部で生成したクロック信号を出力する端子です。

* : Ethernet-MAC を 1 個搭載の品種は、E_PPS 端子です。ch.1 の出力は選択できません。

2.2 内部接続端子

システムクロック信号および AHB バスインタフェース

各 Ethernet-MAC は、AHB バスを経由して CPU, メモリに接続します。各 Ethernet-MAC を使用する際、AHB クロック (HCLK) を 25 MHz 以上に設定してください。

各 Ethernet-MAC のシステム・タイム・カウンタのソースクロック (PTP_CLK) には、AHB クロック (HCLK) が接続されます。

割込み信号インタフェース

各 Ethernet-MAC 部から出力される割込み信号を Table 2-3 に示します。これらの割込みは、NVIC に接続されています。割込みについては、『ペリフェラルマニュアル』の『割込み』の章を参照してください。

Table 2-3 Ethernet-MAC からの割込み信号

割込み信号名	機能
MAC0_INT_SBD	Ethernet-MAC ch.0 の割込みです。
MAC0_INT_PMT	Ethernet-MAC ch.0 の WAKE-UP フレーム受信割込みです。
MAC0_INT_LPI	Ethernet-MAC ch.0 の LPI ステート終了通知の割込みです。
MAC1_INT_SBD	Ethernet-MAC ch.1 の割込みです。
MAC1_INT_PMT	Ethernet-MAC ch.1 の WAKE-UP フレーム受信割込みです。

Ethernet-MAC ch.1 は、MII モードを持たないため、LPI 割込みを発生しません。

3. Ethernet-MAC セットアップ制御手順

Ethernet-MAC セットアップ制御手順について説明します。

セットアップについて

マイコンに対する電源投入などにより、AHB バス上にリセットが発生した場合は、以下のセットアップ制御を実施してください。
I/O ポート部の選択を行った後、以下手順を実施し、各 Ethernet-MAC のセットアップを行います。各 Ethernet -MAC へのクロック供給, MII/RMII のモードの選択, 各 Ethernet-MAC へのハードウェアリセット発行および解除を行います。セットアップ完了後、各 Ethernet-MAC の初期設定を行い、動作を開始させます。

MII モード選択時の手順

1. ETH_CLKG.MACEN[1:0]=01 を書き込みます。Ethernet-MAC(ch.0)にクロック供給を開始します。
2. ETH_MODE.IFMODE=0, ETH_MODE.RST0=1, ETH_MODE.RST1=0 を書き込みます。 MII を選択し、Ethernet-MAC(ch.0) にハードウェアリセットを発行します。
3. この時点で、外部 PHY からクロック信号 (RX_CLK, TX_CLK) が入力されている必要があります。クロック信号が入力されていない場合、入力されるまで待ちます。
4. ETH_MODE.IFMODE=0, ETH_MODE.RST0=0, ETH_MODE.RST1=0 を書き込みます。 MII を選択し、Ethernet-MAC(ch.0) のハードウェアリセットを解除します。

RMII モード (ch.0 のみ使用) 選択時の手順

1. ETH_CLKG.MACEN[1:0]=01 を書き込みます。Ethernet-MAC(ch.0)にクロック供給を開始します。
2. ETH_MODE.IFMODE=1, ETH_MODE.RST0=1, ETH_MODE.RST1=0 を書き込みます。RMII を選択し、Ethernet-MAC(ch.0) にハードウェアリセットを発行します。
3. この時点で、外部 PHY からクロック信号 (REF_CLK) が入力されている必要があります。クロック信号が入力されていない場合、入力されるまで待ちます。
4. ETH_MODE.IFMODE=1, ETH_MODE.RST0=0, ETH_MODE.RST1=0 を書き込みます。RMII を選択し、Ethernet-MAC(ch.0) のハードウェアリセットを解除します。

RMII モード (ch.0, ch.1 両方使用) 選択時の手順

1. ETH_CLKG.MACEN[1:0]=11 を書き込みます。Ethernet-MAC(ch.0, ch.1)にクロック供給を開始します。
2. ETH_MODE.IFMODE=1, ETH_MODE.RST0=1, ETH_MODE.RST1=1 を書き込みます。RMII を選択し、Ethernet-MAC(ch.0, ch.1)にハードウェアリセットを発行します。
3. この時点で、外部 PHY からクロック信号 (REF_CLK) が入力されている必要があります。クロック信号が入力されていない場合、入力されるまで待ちます。
4. ETH_MODE.IFMODE=1, ETH_MODE.RST0=0, ETH_MODE.RST1=0 を書き込みます。RMII を選択し、Ethernet-MAC(ch.0, ch.1)のハードウェアリセットを解除します。

＜注意事項＞

ETH_MODE.PTPSEL は、出力する PTP 信号のチャネルに合わせて任意の選択が可能です。

4. Ethernet システム制御部レジスタ

Ethernet システム制御部のレジスタについて説明します。

Ethernet システム制御部のレジスタ一覧を Table 4-1 に示します。

Table 4-1 Ethernet システム制御部レジスター覧

オフセットアドレス	レジスタ略称	レジスタ名	参照先
0x00	ETH_MODE	モード選択レジスタ	4.1
0x08	ETH_CLKG	クロックゲーティングレジスタ	4.2

Ethernet システム制御部ベースアドレス：0x40066000

4.1 モード選択レジスタ (ETH_MODE)

ETH_MODE は、Ethernet の外部インタフェースモードを制御します。

レジスタ構成

bit	31	30	29	28	27	26	25	24
Field	予約			PPSSEL	予約			
属性	R	R	R	R/W	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約						RST1	RST0
属性	R	R	R	R	R	R	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約							IFMODE
属性	R	R	R	R	R	R	R	R/W
初期値	0	0	0	0	0	0	0	0

レジスタ機能

[bit31:29] 予約

書込み時は"0"を書き込んでください。読出し時は"0"が読み出されます。

[bit28] PPSSEL

各 Ethernet-MAC の PTP 機能のシステム・タイム・カウンタ・パルス出力のうち、どちらを E_PPS0_PPS1 端子に出力するかを選択します。

bit	動作
書込み時 0	Ethernet-MAC(ch.0)を選択します。
書込み時 1	Ethernet-MAC(ch.1)を選択します。
読出し時	レジスタ設定値を読み出します。

[bit27:10] 予約

書込み時は"0"を書き込んでください。読出し時は"0"が読み出されます。

[bit9] RST1

Ethernet-MAC(ch.1) に対するハードウェアリセット信号を制御します。

bit	動作
書込み時 0	Ethernet-MAC(ch.1)に対するハードウェアリセットを解除します。
書込み時 1	Ethernet-MAC(ch.1)に対するハードウェアリセットを発行します。
読出し時	レジスタ設定値を読み出します。

[bit8] RST0

Ethernet-MAC (ch.0) に対するハードウェアリセット信号を制御します。

bit	動作
書込み時 0	Ethernet-MAC(ch.0) に対するハードウェアリセットを解除します。
書込み時 1	Ethernet-MAC(ch.0) に対するハードウェアリセットを発行します。
読出し時	レジスタ設定値を読み出します。

[bit7:1] 予約

書込み時は"0"を書き込んでください。読出し時は"0"が読み出されます。

[bit0] IFMODE

MII/RMII モードセレクタの接続を選択します。

bit	動作
書込み時 0	MII を選択します。
書込み時 1	RMII を選択します。
読出し時	レジスタ設定値を読み出します。

4.2 クロックゲーティングレジスタ (ETH_CLKG)

ETH_CLKG は、各 Ethernet-MAC へのクロック供給を制御します。

レジスタ構成

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約						MACEN[1:0]	
属性	R	R	R	R	R	R	R/W	R/W
初期値	0	0	0	0	0	0	0	0

レジスタ機能

[bit31:2] 予約

書込み時は"0"を書き込んでください。読出し時は"0"が読み出されます。

[bit1:0] MACEN[1:0]

Ethernet-MAC へのシステムクロック供給を選択します。

bit1:0	Ethernet-MAC 1 個搭載品種	Ethernet-MAC 2 個搭載品種
書込み時 00	Ethernet-MAC(ch.0) へクロック供給停止	Ethernet-MAC(ch.0, ch.1)へクロック供給停止
書込み時 01	Ethernet-MAC (ch.0)へクロック供給開始	Ethernet-MAC(ch.0)へクロック供給開始 Ethernet-MAC(ch.1)へクロック供給停止
書込み時 11	設定禁止	Ethernet-MAC(ch.0, ch.1)へクロック供給開始
書込み時 10	設定禁止	設定禁止
読出し時	レジスタ設定値を読み出します。	レジスタ設定値を読み出します。

CHAPTER 2: Ethernet-MAC

Ethernet-MAC の機能と動作の説明を行います。

1. 概要
2. ブロック構成
3. アーキテクチャ
4. レジスタ
5. ディスクリプタ
6. プログラミングガイド

1. 概要

Ethernet-MAC は、以下の 4 つのブロックで構成されています。各ブロックの主な機能を説明します。

GMAC: Ethernet Media Access Controller

- IEEE802.3-2005 準拠
- 10/100Mbps のデータ転送速度に対応
- IEEE802.3 準拠の MII インタフェース
- 10/100 Mbps において全二重と半二重動作に対応
- 半二重動作で CSMA/CD プロトコルに対応
- 半二重モードでバックプレッシャに対応
- 全二重動作で IEEE 802.3x フローコントロールに対応
- CRC およびデータパディングの自動生成
- ジャンボフレーム対応
- 種々のフレキシブル・アドレスフィルタリングモードに対応
- MII 上のデバッグ用内部ループバックに対応
- IEEE802.3Q VLAN パケットに対応
- Wake-On-LAN (Remote Wake-Up, Magic Packet)に対応
- Checksum Offload に対応
- Energy Efficient Ethernet (EEE)用の IEEE 802.3-az-2010 に対応
- RMII インタフェースに対応
- IEEE1588-2008 準拠
- MAC Management Counter をサポート

MTL: MAC Transaction Layer

- 送信 FIFO 2K バイト, 受信 FIFO 2K バイト
- コリジョンフレームの自動再送信処理
- レイトコリジョン, アンダーラン・フレームの廃棄

DMA: DMA Controller

- ディスクリプタ方式 DMA
- 送信および受信データパスはリトルエンディアンに対応

AHB: AHB Interface

- AMBA Specification Rev.2.0 準拠
- 32 ビット幅のアドレスおよびデータ
- マルチレイヤ・バス対応

2. ブロック構成

Ethernet-MAC のブロック構成を説明します。

Figure 2-1 Ethernet-MAC のブロック図

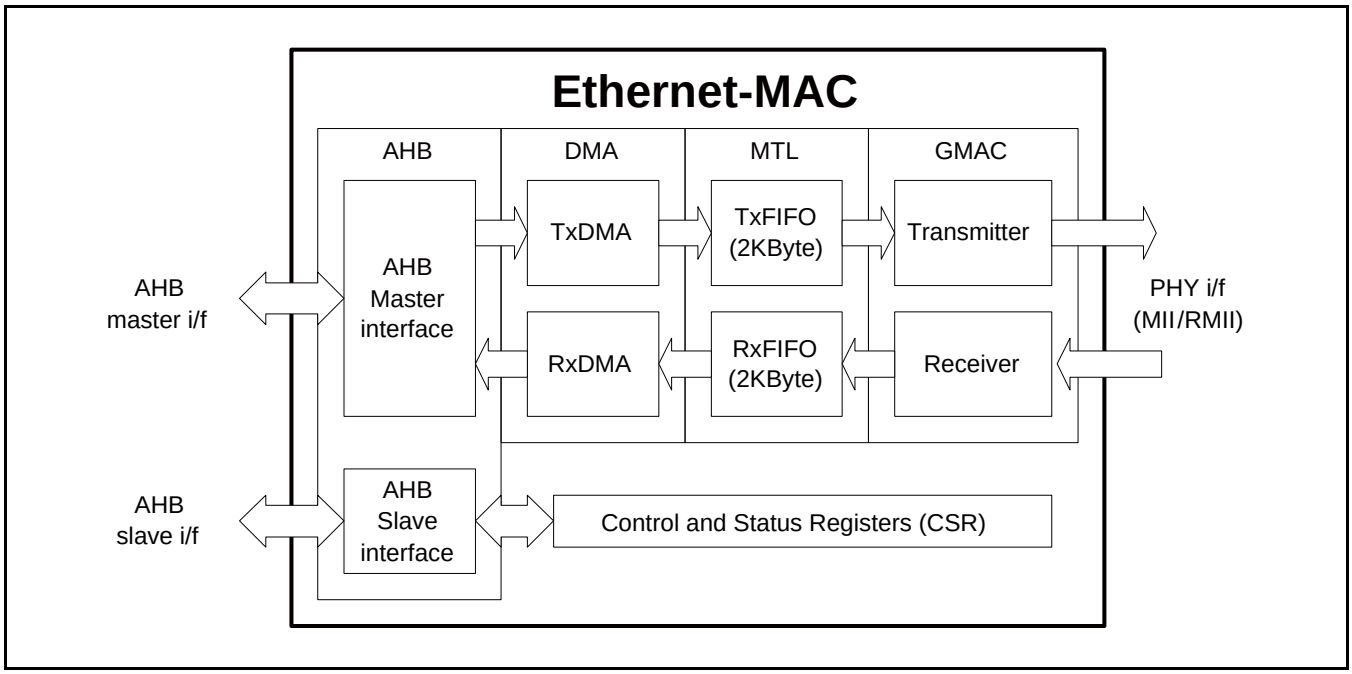


Figure 2-1 に、Ethernet-MAC のブロック図を示します。以下に各機能ブロックの説明を行います。

AHB: AHB Interface

システムバス (AHB) とのインタフェースを制御する機能ブロックです。内部 DMA は、AHB マスタ・インタフェースを持ち、システムバス経由でホストメモリに直接アクセスできます。AHB スレーブ・インタフェースは、ホスト CPU と Control Status Registers (CSR)とのインタフェースを行います。

DMA: DMA Controller

送信 DMA (TxDMA) と受信 DMA (RxDMA) が独立に動作できる構成です。DMA はホストメモリに構築されたディスクリプタテーブルの内容に従い、データ転送処理を行います。

MTL: MAC Transaction Layer

送信データ, 受信データのバッファリング用にそれぞれ 2K バイトの FIFO メモリを搭載しています。MTL は FIFO の制御を行う機能ブロックで、自動再送信, フレームの破棄処理などを行います。

GMAC: Ethernet Media Access Controller

Ethernet の通信プロトコルを実現する機能ブロックです。PHY インタフェースは、MII/RMII をサポートしています。

3. アーキテクチャ

Ethernet-MAC のアーキテクチャについて説明します。

3.1 端子機能
3.2 AHB アプリケーション・ホスト・インタフェース
3.3 DMA コントローラ
3.4 チェックサムエンジン
3.5 Energy Efficient Ethernet
3.6 MAC マネジメントカウンタ
3.7 Station Management Agent
3.8 IEEE1588

3.1 端子機能

Ethernet-MAC の端子機能について説明します。

PHY インタフェース端子

Table 3-1 に PHY インタフェース端子を示します。GMAC は、MII/RMII の両方のインタフェースをサポートします。表中の MII/RMII の列で○が付いている端子を PHY に接続する必要があります。

Table 3-1 PHY インタフェース端子

信号端子名	I/O	機能	MII	RMII	備考
RX_CLK	IN	受信クロック	○	×	100 Mbps 時: 25 MHz 10 Mbps 時: 2.5 MHz
RXD[0]	IN	受信データ 0	○	○	
RXD[1]	IN	受信データ 1	○	○	
RXD[2]	IN	受信データ 2	○	×	
RXD[3]	IN	受信データ 3	○	×	
RX_DV	IN	受信データ有効	○	×	
RX_ER	IN	受信エラー検出	○	×	RMII 時接続不要
TX_CLK	IN	送信クロック	○	×	100 Mbps 時: 25 MHz 10 Mbps 時: 2.5 MHz
TXD[0]	OUT	送信データ 0	○	○	
TXD[1]	OUT	送信データ 1	○	○	
TXD[2]	OUT	送信データ 2	○	×	
TXD[3]	OUT	送信データ 3	○	×	
TX_EN	OUT	送信データ有効	○	○	
TX_ER	OUT	送信エラー	○	×	Energy Efficient Ethernet 対応 PHY にのみ接続
CRS	IN	キャリア検出	○	×	
COL	IN	衝突検出	○	×	
MDC	OUT	マネジメントクロック	○	○	SYS_CLK 入力を分周して生成します。
MDIO	IO	マネジメントデータ	○	○	外部端子にて双方向制御されます。プルアップすることを推奨します。
REF_CLK	IN	リファレンスクロック	×	○	50 MHz クロック入力
CRS_DV	IN	キャリア検出/データ有効	×	○	

ホスト・インタフェース，その他の端子

Table 3-2 にホスト・インタフェース端子，その他の端子を示します。本回路の動作，機能を説明するに必要な端子に限定しています。

Table 3-2 ホスト・インタフェース，その他の端子

端子名	I/O	機能	説明
SYS_CLK	IN	システムクロック	– AHB バス上のクロック (HCLK) が、接続されています。 – DMA の転送動作の基準クロックとなります。 – マネジメントインタフェースの MDC は、SYS_CLK を分周して生成します。 – 必ず、25 MHz 以上の HCLK を供給してください。
PTP_CLK	IN	PTP クロック	– PTP システムカウンタモジュールの基準クロック入力です。 – マイコン上の AHB クロック (HCLK)が接続されています。
INT_SBD	OUT	Ethernet-MAC 割込み	– Ethernet-MAC からの様々なイベントを通知する割込み信号です。 – マイコンの NVIC に接続されます。
INT_PMT	OUT	WakeUp 割込み	– GMAC レシーバが、PHY から WakeUp パケットを受信したイベントを通知する割込み信号です。 – マイコンをスタンバイモードから復帰させることができます。
INT_LPI	OUT	LPI 割込み	– GMAC レシーバが、PHY から LPI (Low Power Idle)状態を抜けた通知を受けたイベントを通知する割込み信号です。 – マイコンをスタンバイモードから復帰させることができます。
PTPPPS	OUT	秒カウント出力	– PTP システムタイムカウンタモジュールの秒カウント出力です。 – マイコンの外部端子へ出力できます。詳細は、「4.30. GMAC Register 459 (PPSCR)」を参照してください。

3.2 AHB アプリケーション・ホスト・インタフェース

Ethernet-MAC の AHB インタフェースについて、説明します。AHB マスタ・インタフェースは、内部 DMA 要求サイクルを AHB サイクルに変換し、データの転送を制御します。AHB スレーブ・インタフェースは、ホスト CPU からの GMAC と DMA の CSR (Control and Status Register) スペースへのアクセスを提供します。AHB スレーブ・インタフェースへのアクセス中に、AHB マスタ・インタフェースの動作が可能です。

AHB マスタ・インタフェース

- DMA Bus Mode レジスタの FB または MB ビットをプログラミングすることによって、固定長バースト・モード(SINGLE, INCR4, INCR8, INCR16)または、不定長バースト・モード(SINGLE, INCR)転送、またはその混合を選択できます。
- 固定長バースト・モードを選択した場合、AHB マスタは常に SINGLE, INCR4, INCR8 または INCR16 タイプのバーストを開始します。固定長バースト・モードでは、DMA が INCR4/8/16 でないバースト転送を要求した場合、AHB ホスト・インタフェースは転送を複数のバースト・トランザクションに分割します。例えば、DMA が 15 ビートのバースト転送を要求した場合、AHB インタフェースはそれを INCR8 と INCR4 と 3 つの SINGLE トランザクションに分割します。
- 不定長バースト・モードでは、AHB マスタは常に INCR で転送を開始し、DMA によって要求されたバーストを 1 つの転送で完了します。
- 混合バースト・モードででは、AHB マスタは DMA が 16 ビート以下のサイズの転送を要求したとき、固定長バースト (INCRx)を開始します。DMA が 17 ビート以上の長さのバーストを要求したとき、AHB マスタは不定長バースト (INCR) で転送を開始し、バーストを 1 つの転送で完了します。
- AHB マスタからアクセスできないメモリ領域 (メモリが存在しない領域) へのアクセスが発生した場合、バスエラーが発生します。この場合、DMA はすべてのトランザクションを停止し、CSR および割込みを通じてエラーが致命的であることを知らせます。動作を再開するには、アプリケーションは、ソフトリセットを実行しなければなりません。
- AHB マスタからの転送は、常に 32 ビット (4byte) アドレス境界にアライメントされます。
- DMA コントローラは、送信 FIFO にバーストデータを完全に受け入れられるスペースがあるときにだけ、AHB バーストリード転送を要求します。
- DMA コントローラは、受信 FIFO に十分なバーストデータがあるときにだけ、AHB バーストライト転送を要求します。AHB マスタ・インタフェースは常に、AHB バスにプッシュするデータが利用可能であることを前提としています。しかし、DMA はバースト中に、完了前に end-of-valid data (有効なデータの終わり)を示すことがあります(Ethernet フレームの end-of-frame の送信のため)。固定長バースト・モードでは、AHB マスタ・インタフェースは指定された長さの転送が完了するまでダミーデータを使ってバーストを継続します。INCR モードでは、バースト転送を完了前に終了するステップを実行します。

AHB スレーブ・インタフェース

- シングルおよびバースト転送をサポート
- CSR との間の 32 ビット, 16 ビット, 8 ビットのライト/リード転送をサポートしています。ただし、ソフトウェア同期の問題を避けるため、CSR へのアクセスは、32 ビット・アクセスが推奨されます。
- OKAY 応答のみを生成し、ERROR 応答を生成しません。

3.3 DMA コントローラ

Ethernet-MAC 内部の DMA コントローラについて説明します。Eternet-MAC は、独立した送信 DMA エンジンと受信 DMA エンジン、および CSR (Control and Status Registers)スペースを備えています。送信エンジンはデータをシステムメモリからデバイスポート(MTL)へ転送し、受信エンジンはデータをシステムメモリへ転送します。DMA コントローラはディスクリプタを利用することによってデータを転送元から転送先へ効率的に移動し、ホスト CPU の介入を最小限にします。DMA は Ethernet のフレームのようなパケット方式のデータ転送のために設計されています。DMA コントローラは、フレームの送受信完了, その他正常/エラー条件でホスト CPU に割込みを発生させるようにプログラムできます。

3.3.1 ディスクリプタ構造

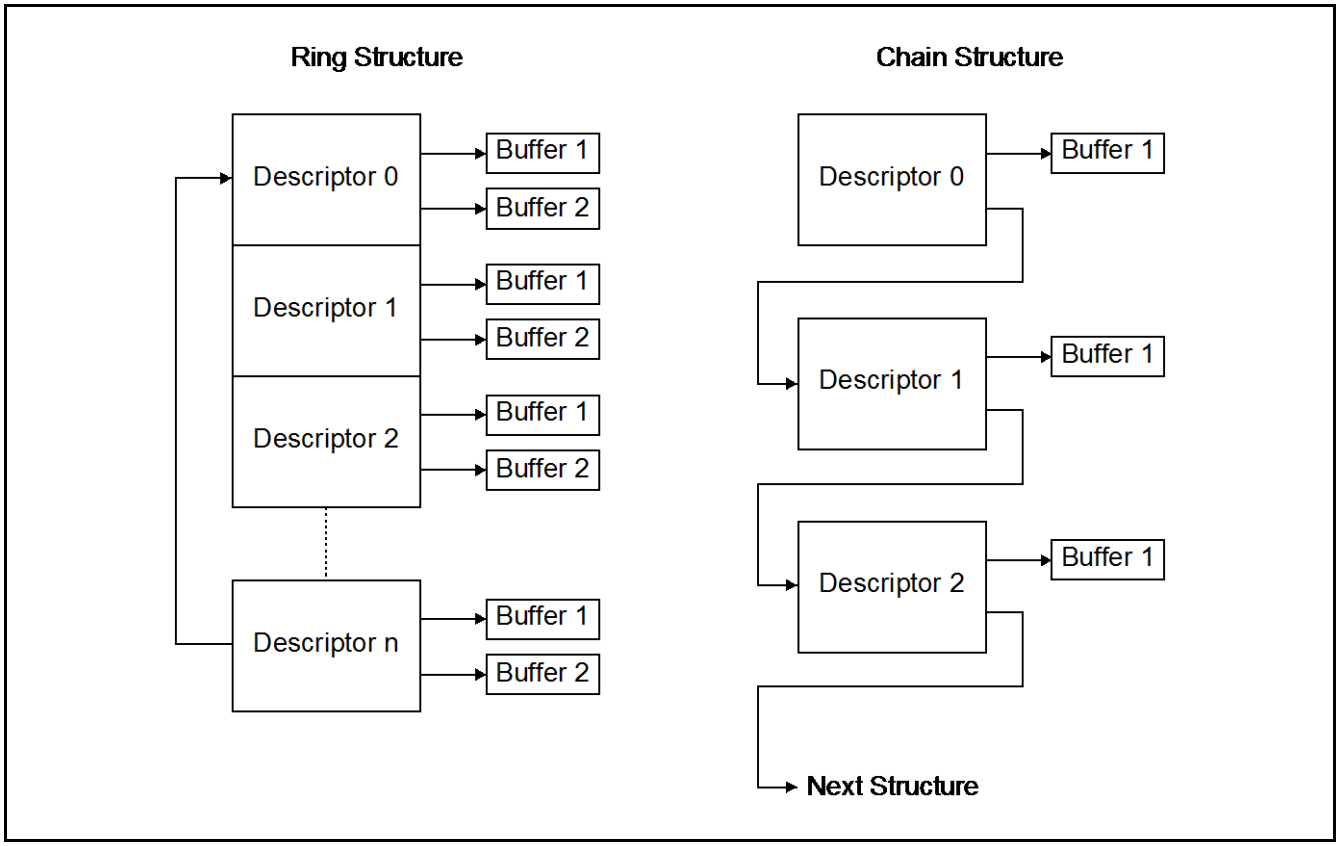
DMA とホストドライバは、下記の 2 つのデータ構造で通信します。

- Control and Status registers (CSR)
- ディスクリプタリストおよびデータバッファ

コントロール・ステータスレジスタについては「4.レジスタ」のセクションで詳細に説明しています。ディスクリプタについては「5.ディスクリプタ」のセクションで詳細に説明しています。

DMA は GMAC から受信したデータフレームをホストメモリの受信バッファへ転送し、ホストメモリの送信バッファからの送信データフレームを送信します。ホストメモリに常駐するディスクリプタはこれらのバッファのポインタとして機能します。

Figure 3-1 ディスクリプタの Ring 構造と Chain 構造



ディスクリプタは、受信用と送信用の 2 つがあります。各リストのベースアドレスをそれぞれ DMA Registers 3 および 4 へ書き込みます。ディスクリプタリストは(暗黙または明示的に)フォワードリンクされます。最後のディスクリプタが最初のエントリに戻り、Ring 構造を生成できます。ディスクリプタを明示的に Chain 化するには、受信および送信ディスクリプタの両方の 2 番目のアドレスを Chain 化します。Figure 3-1 に、ディスクリプタの Ring 構造および Chain 構造を示します。

ディスクリプタリストは、ホストメモリスペースに常駐します。各ディスクリプタは、最大 2 つのバッファをポイントできます。これによって 2 つのバッファを、メモリ内の連続するバッファとして使用するのではなく、物理アドレスを指定して使用できます。
データバッファはホストメモリスペースに常駐し、フレーム全体または一部によって構成されますが、1 つのフレームを超えることはできません。バッファはデータのみを含み、ステータスはディスクリプタに保存されます。データの Chain 化では複数のデータバッファにわたるフレームを参照します。しかし、1 つのディスクリプタは複数のフレームにわたることはできません。DMA は end-of-frame が検出されたとき、次のフレーム・バッファへスキップします。データの Chain 化を許可または禁止できます。

3.3.2 DMA コントローラの初期化

Ethernet-MAC の初期化は以下のように行います。

1. GMAC Register 0 に書込みを行い、動作モードを構成し、送信動作をイネーブルします。PS および DM ビットをオート・ネゴシエーションされた結果(PHY からリードされる)を元に設定します。
2. DMA Register 0 に書込みを行い、ホストのバスアクセス・パラメータをセットします。
3. DMA Register 7 に書込みを行い、不要な割込み原因をマスクします。
4. ソフトウェアドライバは、送信および受信ディスクリプタリストを作成します。次に、DMA Register 3 と DMA Register 4 の両方に書き込み、DMA に各リストの開始アドレスを供給します。
5. GMAC Registers 1, 2, 3 に書込みを行い、フィルタ・オプションを指定します。
6. DMA Register 6 の bit13 と bit1 をセットし、送信 DMA, 受信 DMA の動作を開始します。
7. GMAC Register 0 に書込みを行い、受信動作を許可します。 (Receiver Enable: bit2)。
   送信および受信エンジンは Running ステートに入り、それぞれのディスクリプタリストからディスクリプタを取得しようとします。次に受信および送信エンジンは、受信および送信動作の処理を開始します。送信および受信プロセスは相互に独立しており、別々に開始または停止できます。

ホスト・バス・バーストアクセス

DMA は、AHB マスタ・インタフェース上での固定長バーストを行うように構成されている場合(DMA Register 0 の FB ビット)、それを実行します。最大バースト長は PBL フィールド(DMA Register 0[13:8])によって指定および制限されます。16 バイトの読出しのために、受信および送信ディスクリプタは常に可能な最大のバーストサイズ(PBL によって制限されるか、または 16 * 8/バス幅)でアクセスされます。
送信 DMA は、MTL 送信 FIFO に構成されたバースト、または(構成されたバーストよりもサイズが小さい場合は)end of frame までのバイト数を受け入れる十分なスペースがある場合にのみデータ転送を開始します。DMA は AHB マスタ・インタフェースに開始アドレスと要求される転送の数を指示します。AHB インタフェースが固定長バースト・モードに構成されている場合、DMA は INCR4/8/16 および SINGLE トランザクションの最適の組合せを使ってデータを転送します。そうでない場合(固定長バーストを使用しない)は、INCR (不定長)と SINGLE トランザクションを使ってデータを転送します。
受信 DMA は、MTL 受信 FIFO に構成されたバーストを受け入れる十分なスペースがある場合、または(構成されたバーストよりもサイズが小さい場合)受信 FIFO で end of frame が検出された場合にのみデータ転送を開始します。DMA は AHB マスタ・インタフェースに開始アドレスと要求される転送の数を指示します。AHB インタフェースが固定長バースト・モードに構成されている場合、DMA は INCR4/8/16 および SINGLE トランザクションの最適の組合せを使ってデータを転送します。そうでない場合(DMA Register 0 の FB ビットがリセットされている)は、INCR (不定長)と SINGLE トランザクションを使ってデータを転送します。

ホスト・データバッファ・アライメント

フレームの先頭の送信および受信データバッファの開始アドレスにアライメントに関する制約はありません。例えば、32 ビット・メモリを備えたシステムでは、バッファの開始アドレスは 4 つのバイトのどれにでもアライメントできます。しかし DMA は常に 32 ビット幅にアライメントされたアドレスに、必要でないバイトレーンはダミーデータを使って、転送を行います。これは一般的には、Ethernet フレームの始まりまたは終わりの転送の際に行われます。

■ Example 3-1 バッファリード

送信バッファアドレスが 0x0FF2 で、15 バイトを送信する場合、DMA はアドレス 0x0FF0 から 5 つ 32 ビットデータをリードします。データを MTL 送信 FIFO へ転送するとき、余分のバイト(最初の 2 バイト)はドロップされ無視します。同様に、最後の転送の最後の 3 バイトも無視します。DMA は常に、end-of-frame でない限り完全な 32 ビットデータを MTL 送信 FIFO へ転送することを保証します。

■ Example 3-2 バッファライト

受信バッファアドレスが 0x0FF2 で、16 バイトの受信フレームを転送する必要がある場合、DMA はアドレス 0x0FF0 から 5 個の 32 ビットデータを書き込みます。最初の転送の最初の 2 バイト、および 5 番目の転送の最後の 2 バイトにはダミーデータを書き込みます。

バッファサイズの計算

DMA は送信および受信ディスクリプタのサイズ・フィールドを更新しません。DMA はディスクリプタのステータスフィールド(RDES および TDES)だけを更新します。ドライバはサイズ計算を実行しなければなりません。

送信 DMA は正確な数のバイト(TDES1 のバッファサイズフィールドで指定されています)を GMAC へ転送します。ディスクリプタが first とマークされている場合(TDES0 の FS ビットがセットされている)、DMA はバッファからの最初の転送に start of frame のマークを付けます。ディスクリプタが last とマークされている場合(TDES0 の LS ビットがセットされている)、DMA はそのデータバッファからの最後の転送に end-of frame to the MTL のマークを付けます。

受信 DMA は、バッファがフルになるまで、または MTL から end-of frame を受信するまで、バッファへデータを転送します。ディスクリプタに last のマーク(RDES0 の LS ビット)が付いていない場合、そのディスクリプタの対応するバッファはフルであり、バッファ内の有効なデータの量はバッファサイズフィールドで指定されているサイズから、そのディスクリプタの FS ビットがセットされているときのデータバッファポ

インタオフセットを引いた量によって正確に示されます。データバッファポインタがデータバス幅にアライメントされているとき、オフセットは 0 です。ディスクリプタが last とマークされている場合、バッファはフルでないことがあります(RDES1 のバッファサイズで示されます)。この最後のバッファの有効なデータの量を計算するには、ドライバはフレーム長をリードし(RDES0[29:16]の FL ビット)、このフレームの先行フレームのバッファサイズの合計を引く必要があります。受信 DMA は常に、次のフレームを新しいディスクリプタを使って転送します。

<注意事項>

- ***受信バッファの開始アドレスがシステムバスのデータ幅にアライメントされていない場合、システムはシステムバス幅にアライメントされたサイズの受信バッファを割り当てなければなりません。例えば、システムがアドレス 0x1000 から 1024 バイト(1 KB)の受信バッファを割り当てる場合、ソフトウェアは受信ディスクリプタの中のバッファ開始アドレスに 0x1002 のオフセットを含めるようにプログラムできます。受信 DMA はこのバッファにフレームを書き込むとき、最初の 2 つの位置(0x1000 および 0x1001)にダミーデータを挿入します。実際のフレームは位置 0x1002 から書き込まれます。したがって、バッファサイズが 1024 とプログラミングされている場合でも、開始アドレスオフセットのため、このバッファの実際の有効なスペースは 1022 バイトとなります。***

DMA アービター

DMA モジュール内のアービターは、送信および受信チャネル間の AHB マスタ・インタフェースへのアクセスのアービトレーションを実行します。round-robin と fixed-priority の 2 つのタイプのアービトレーションが可能です。DMA Register 0 の DA=0 の場合、round-robin アービトレーションが選択され、PR, TXPR の設定により、プライオリティが選択されます。DA=1

の場合、fixed-priority アービトレーションが選択され、TXPR の設定により、プライオリティが選択されます。詳細は、DMA Register 0 の「4.33. DMA Register 0 (BMR)」を参照してください。

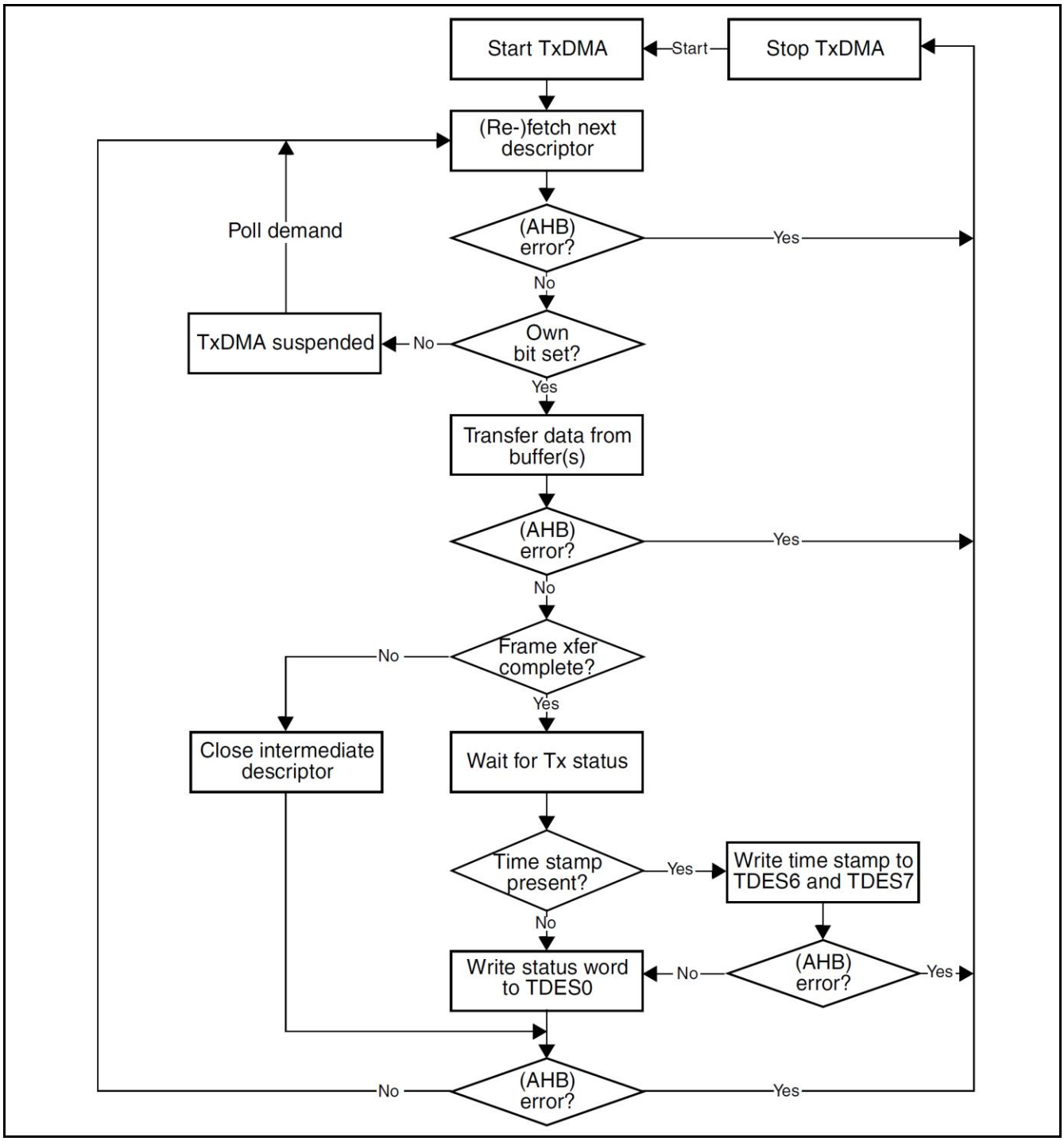
3.3.3 送信動作

送信 DMA 動作: デフォルト(Non-OSF)モード

送信 DMA エンジンは、デフォルトモードでは以下のような処理を行います。デフォルトモードでの送信 DMA 送信フローを Figure 3-2 に示します。

1. ホストは、送信ディスクリプタ(TDES0-TDES3)をセットアップし、対応するデータバッファを Ethernet フレームデータによってセットアップした後、OWN ビット(TDES0[31])をセットします。
2. ST ビット(DMA Register 6[13])をセットした後、DMA は Run ステートに入ります。
3. Run ステートのとき、DMA は、送信を要求しているフレームの送信ディスクリプタリストをポーリングします。ポーリング開始後、後続の Ring オーダ, Chain オーダのいずれかのディスクリプタを読み出します。DMA が、ホストによって所有されたフラグ付きディスクリプタを検出したとき、またはエラー条件が発生したとき、送信が一時停止され、Transmit Buffer Unavailable (DMA Register 5[2])および Normal Interrupt Summary (DMA Register 5[16])ビットの両方がセットされます。送信エンジンはステップ 9 に進みます。
4. 取得したディスクリプタに DMA の所有フラグがセットされている場合 (TDES0[31] = 1)、DMA は取得したディスクリプタから、送信データバッファアドレスをデコードします。
5. DMA はホストメモリから送信データを取得し、そのデータを送信のために MTL へ転送します。
6. Ethernet フレームが複数のディスクリプタのデータバッファに格納されている場合、DMA は中間のディスクリプタをクローズして、次のディスクリプタを取得します。end-of-Ethernet-frame データが MTL へ転送されるまでステップ 3, 4, 5 が繰り返されます。
7. フレーム送信が完了したとき、そのフレームの IEEE 1588 タイムスタンプ生成がイネーブルされている場合(送信ステータスによって示される)、MTL から取得したタイムスタンプ値が、end-of-frame バッファを含んでいる送信ディスクリプタ(TDES6 および TDES7)に書き込まれます。次に、ステータス情報がこの送信ディスクリプタ(TDES0)に書き込まれます。このステップ中に OWN ビットがクリアされるので、ホストがこのディスクリプタを所有することになります。このフレームにタイムスタンプ生成がイネーブルされていない場合、DMA は TDES6 および TDES7 の内容を変更しません。
8. 最後のディスクリプタの Interrupt on Completion (TDES0[30])がセットされているフレームの転送が完了した後、送信割込み(DMA Register 5[0])がセットされます。このとき、DMA エンジンはステップ 3 に戻ります。
9. Suspend ステートのとき、転送ポーリング要求を受信し、Underflow Interrupt Status ビットがクリアされていると、DMA はディスクリプタの再取得を試みます(その後、ステップ 3 に戻ります)。

Figure 3-2 デフォルトモードでの送信 DMA の動作



送信 DMA の動作: OSF モード

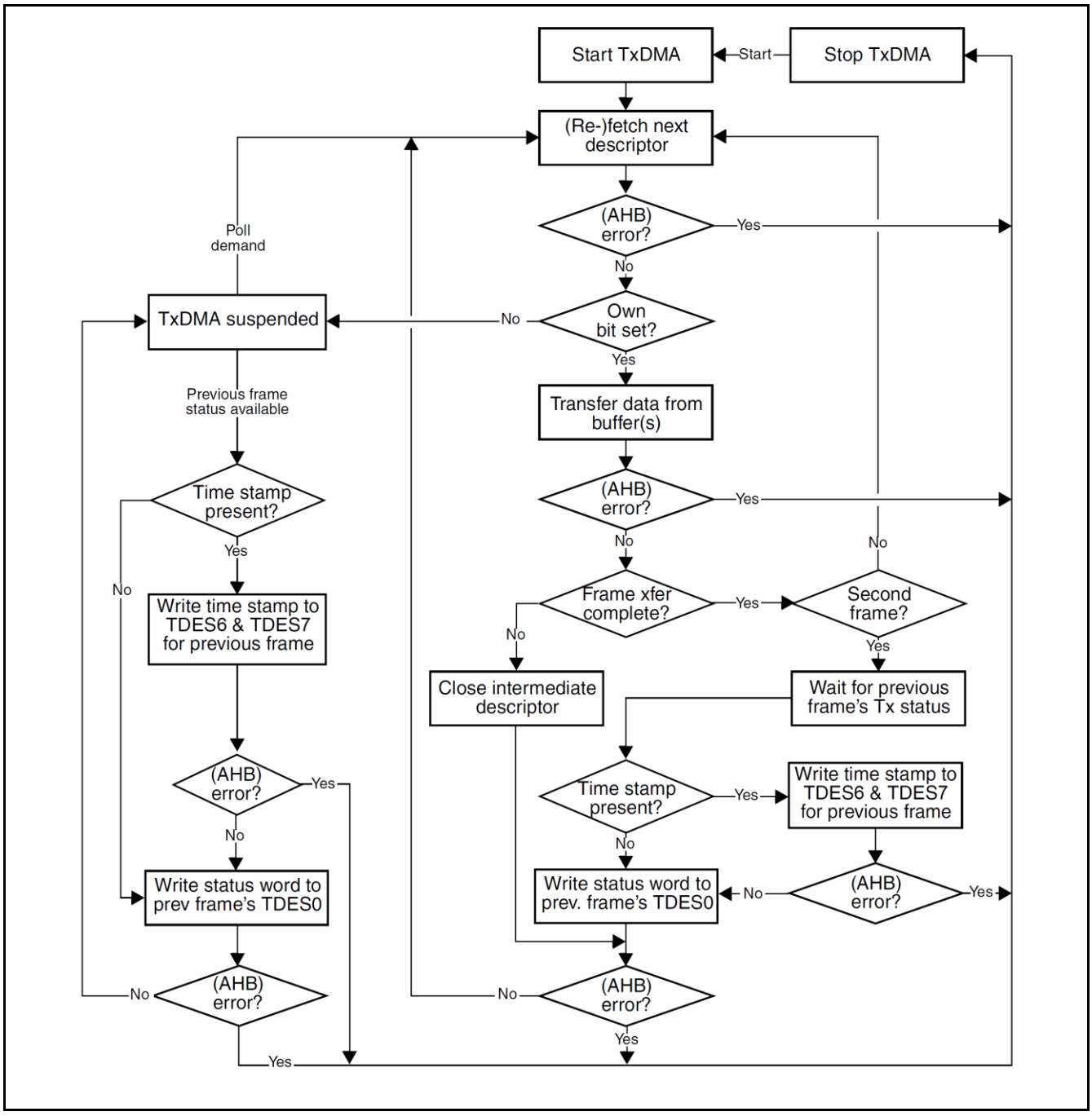
DMA Register 6[2]の OSF (Operate on Second Frame) ビットがセットされている場合、Run ステートで、送信処理は、最初のフレームのステータス・ディスクリプタをクローズすることなく 2 つのフレームを同時に取得できます。送信プロセスは、最初の送信が終了すると、即座に 2 番目のフレームの送信ディスクリプタリストをポーリングします。2 番目のフレームが有効である場合、送信プロセスは最初のフレームのステータス情報を書き込む前にこのフレームを転送します。
OSF モードでは、Run ステートの送信 DMA は以下の順序で動作します。フローを Figure 3-3 に示します。

1. DMA は送信 DMA (デフォルトモード)のステップ 1-6 に示すように動作します。
2. DMA は、前のフレームの最後のディスクリプタをクローズすることなしに次のディスクリプタを取り込みます。
3. DMA が取得したディスクリプタを所有する場合、DMA はこのディスクリプタ内の送信バッファアドレスをデコードします。DMA がディスクリプタを所有しない場合、DMA は Suspend モードに入り、ステップ 7 へスキップします。
4. DMA はホストメモリから送信フレームを取り込み、End-of-Frame データが転送されるまで MTL へフレームを転送し、このフレームが複数のディスクリプタにわたる場合は中間のディスクリプタをクローズします。
5. DMA は前のフレームのフレーム送信ステータスおよびタイムスタンプを待ちます。ステータスが入手でき、(ステータスビットによって示される)タイムスタンプが取得された場合、DMA はそのタイムスタンプを TDES6 および TDES7 に書き込みます。次に DMA は、OWN ビットをクリアしたステータスを対応する TDES0 へ書き込み、それによってディスクリプタをクローズします。前のフレームでタイムスタンプの生成がイネーブルされていない場合、DMA は TDES6 および TDES7 の内容を変更しません。
6. 送信割込みがイネーブルされていれば、送信割込みをセットします。DMA は次のディスクリプタを取り込み、ステップ 3 に進みます(ステータスが正常の場合)。前の送信のステータスがアンダフローエラーを示す場合、DMA は Suspend モードに入ります(ステップ 7) 。
7. Suspend モードでは、MTL から保留中のステータスおよびタイムスタンプを受信した場合、DMA はタイムスタンプ(現在のフレームでイネーブルされている場合)を TDES6 および TDES7 に書き込み、次にステータスを対応する TDES0 に書き込みます。その後に、関連する割込みをセットし、Suspend モードに戻ります。
8. DMA は送信ポーリング要求(DMA Register 1)を受信した後でのみ、Suspend モードを終了して Run ステートに入ることができます(保留中のステータスに応じてステップ 1 またはステップ 2 へ進みます)。

＜注意事項＞

- *DMA は現在のディスクリプタをクローズする前に次のディスクリプタを取り込むため、ディスクリプタ Chain が正しく、適切に処理されるためには 3 つ以上の異なるディスクリプタが必要です。*

Figure 3-3 OSF モードの送信 DMA の動作



送信フレーム処理

送信 DMA はデータバッファがプリアンブル, パッドバイト, FCS フィールドを除いた完全な Ethernet フレームを含んでいると想定します。DA, SA, Type/Len フィールドは有効なデータを含んでいる必要があります。送信ディスクリプタによって、CRC, PAD の挿入をディセーブルするよう指示されている場合、バッファは CRC バイトを含む完全な Ethernet フレーム(プリアンブルを除く)を含んでいなければなりません。フレームはデータの Chain 化が可能で、複数のバッファも使用できます。フレームは FS(TDES0[28])と LS(TDES0[29])によって区切らなければなりません。送信を開始するとき、最初のディスクリ

プタの FS(TDES0[28])はセットされていなければなりません。このとき、フレームデータがホストのバッファから MTL 送信 FIFO へ転送されます。同時に、そのフレームの LS(TDES0[29])がクリアされている場合、送信プロセスは次のディスクリプタの取得を試みます。送信プロセスはこのディスクリプタの FS(TDES0[28])がクリアされていると想定します。
LS(TDES0[29])がクリアされていれば、それは中間バッファを表します。LS(TDES0[29])がセットされていれば、それはフレームの最後のバッファを表します。フレームの最後のバッファが送信された後、DMA はステータス情報を、LS(TDES0[29])がセットされているディスクリプタの TDES0 ワードに書き戻します。このとき、Interrupt on Completion (TDES0[30])がセットされている場合、Transmit Interrupt (DMA Register 5[0])がセットされ、新規ディスクリプタが取り込まれ、このプロセスが繰り返されます。実際のフレーム送信は、MTL 送信 FIFO がプログラム可能な送信しきい値(DMA Register 6[16:14])に到達した、または完全なフレームが FIFO に入った後 (Store and Forward Mode (DMA Register 6[21])のオプション) に開始されます。
PHY インタフェースからフレームの送出が終了したとき、最後のディスクリプタが開放されます(OWN ビット TDES0[31]がクリアされます)。

送信ポーリング一時停止

以下のいずれかの条件で、送信ポーリングが一時停止されます。

- DMA がホストによって所有されるディスクリプタ(TDES0[31]=0)を検出したとき。再開するにはドライバはディスクリプタの所有権を DMA に渡し、Poll Demand コマンドを発行しなければなりません。
- アンダフローによる送信エラーが検出されたため、フレーム送信がアボートされたとき。適切な Transmit Descriptor 0 (TDES0)ビットがセットされます。

上の 2 番目の条件が発生したとき、Abnormal Interrupt Summary (DMA Register 5[15])と Transmit Underflow ビット(DMA Register 5 [5])の両方がセットされ、その情報が Transmit Descriptor 0 に書き込まれ、一時停止になります。1 番目の条件で DMA が Suspend ステートに入った場合、Normal Interrupt Summary (DMA Register 5 [16])と Transmit Buffer Unavailable (DMA Register 5 [2])がセットされます。
いずれの場合も、送信リスト内での位置は保持されます。保持される位置は、最後に DMA によってクローズされたディスクリプタの位置です。ドライバは一時停止の原因を是正した後、明示的に Transmit Poll Demand コマンドを発行しなければなりません。

3.3.4 受信動作

受信 DMA エンジンの受信シーケンスを Figure 3-4 および以下に手順を示します。

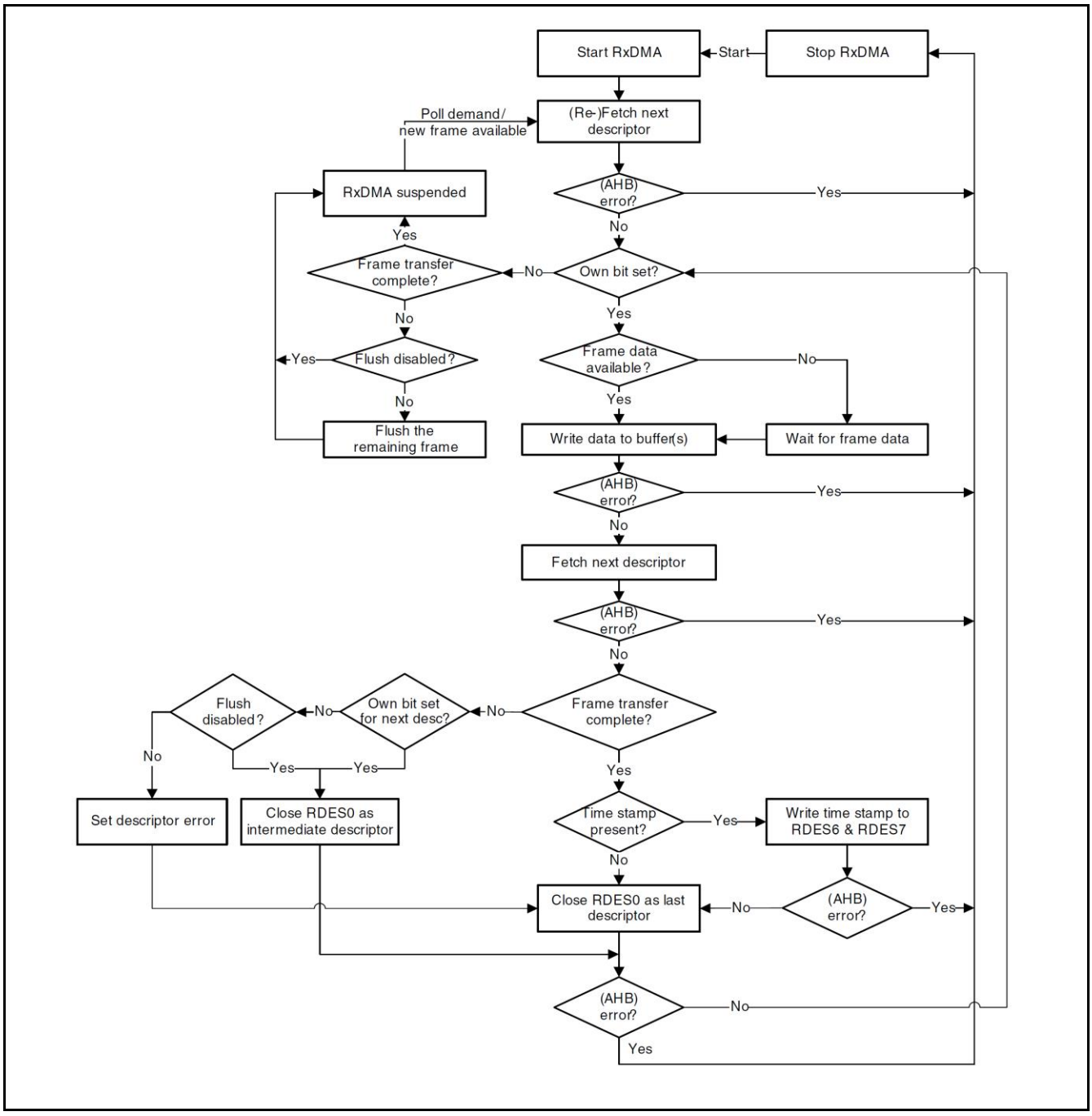
1. ホストは受信ディスクリプタ(RDES0-RDES3)をセットアップし、OWN ビット(RDES0[31])をセットします。
2. SR (DMA Register 6[1])ビットがセットされると、DMA は Run ステートに入ります。Run ステートでは DMA は受信ディスクリプタリストをポーリングし、使用可能なディスクリプタの取得を試みます。取り込んだディスクリプタが使用可能でない(ホストによって所有されている)場合、DMA は Suspend ステートに入り、ステップ 9 にジャンプします。
3. DMA は取得されたディスクリプタからの受信データバッファアドレスをデコードします。
4. 着信フレームが処理され、取得したディスクリプタのデータバッファに入れられます。
5. バッファがフルになるかフレーム転送が完了したとき、受信エンジンは次のディスクリプタを取り込みます。
6. 現在のフレーム転送が完了している場合、DMA はステップ 7 に進みます。DMA が次に取り込まれるディスクリプタを所有しておらず、フレーム転送が完了していない(EOF がまだ転送されていない)場合、DMA は RDES0 の Descriptor Error ビットをセットします(フラッシュがディセーブルされていない場合)。DMA は現在のディスクリプタをクローズし(OWN ビットをクリアします)、RDES0 の Last Segment (LS)ビットをクリアすることによって中間ディスクリプタとしてマークし(フラッシュがディセーブルされていない場合は Last Descriptor としてマークし)、ステップ 8 へ進みます。DMA が次のディスクリプタを所有しているが、現在のフレーム転送が完全でない場合、DMA は現在のディスクリプタを中間ディスクリプタとしてクローズし、ステップ 4 に戻ります。
7. IEEE 1588 タイムスタンプ生成がイネーブルされている場合、DMA はタイムスタンプ(利用可能な場合)を現在のディスクリプタの RDES6 および RDES7 に書き込みます。次に DMA は、MTL から受信フレームのステータスを取得し、ス

テータスワードを現在のディスクリプタの RDES0 に書き込み、OWN ビットをクリアし Last Segment ビットをセットします。

8. 受信エンジンは最新のディスクリプタの OWN ビットをチェックします。ホストがディスクリプタを所有している場合(OWN ビットが 0)、Receive Buffer Unavailable ビット(Register 5[7])がセットされ、DMA 受信エンジンは Suspended ステートに入ります(ステップ 9)。DMA がディスクリプタを所有する場合、エンジンはステップ 4 に戻り、次のフレームを待ちます。
9. Receive エンジンが Suspend ステートに入る前に、受信 FIFO から部分的なフレームがフラッシュされます(DMA Register 6 の bit24 を使用してフラッシュをコントロールできます)。
10. Receive Poll 要求が発行されるか、MTL 受信 FIFO からの次のフレームの転送開始が可能になったとき、受信 DMA は Suspend ステートを終了します。受信エンジンはステップ 2 へ進み、次のディスクリプタを再び取り込みます。

DMA は、タイムスタンプの書戻しを完了し、ディスクリプタへのステータスの書戻しを実行する準備ができるまで MTL からのステータスの受信をアクノリッジしません。
ソフトウェアが CSR を通じたタイムスタンプ生成をイネーブルしている場合、フレームの有効なタイムスタンプ値が入手不能であるとき(例えば、タイムスタンプを書き込むことができるようになる前に受信 FIFO がフルになったため)、DMA は RDES6 および RDES7 にオール 1 を書き込みます。そうでない場合(タイムスタンプがイネーブルされていない場合)、RDES6 および RDES7 は変更されません。

Figure 3-4 受信 DMA の動作



受信ディスクリプタの取得

受信エンジンは常に、着信フレームを予測して余分のディスクリプタを取得しようとします。以下のいずれかの条件が満たされると、ディスクリプタを取得しようとします。

- Receive Start/Stop ビット(DMA Register 6[1])がセットされ、Run ステートに入った直後。
- 現在のフレームの転送を終了する前に、ディスクリプタのデータバッファがフルになった。

- フレーム受信を完了しているが、現在の受信ディスクリプタがまだクローズしていない。
- ホストが所有するバッファ(RDES0[31]= 0)により、受信プロセスが一時停止されている状態で、新規フレームを受信した。
- 受信 Poll Demand コマンドが発行された。

受信フレーム処理

GMAC は、受信フレームがアドレスフィルタを通過し、フレームサイズが MTL 受信 FIFO のしきい値バイト数より大きいか等しいとき、または完全なフレームが Store-and-Forward モードで FIFO に書き込まれたときにのみ、受信フレームをホストメモリに転送します。
フレームがアドレスフィルタを通過しなかった場合、GMAC ブロックの中で破棄されます(GMAC Register 1 [31] :Receive All ビットがセットされていない場合)。衝突や終了が未完了のために 64 バイト未満になっているフレームを MTL 受信 FIFO からパージできます。
64 バイト(構成可能なしきい値)が受信された後、MTL ブロックは DMA ブロックに対してフレームデータを現在のディスクリプタによって指定されている受信バッファへ転送するよう要求します。DMA は、AHB Interface がデータ転送の準備完了状態になった後(DMA がホストからの送信データを取り込んでいない場合)、フレームを区切るために First Descriptor (RDES0[9])をセットします。
ディスクリプタは、データバッファがフルになるか、またはフレームの最後のセグメントが受信バッファへ転送されて OWN (RDES[31])ビットが 0 にリセットされたときに開放されます。
フレームが 1 つのディスクリプタに収まっている場合、LS (RDES0[8])と FS (RDES0[9])の両方をセットします。DMA は次のディスクリプタを取り込み、LS (RDES0[8])ビットをセットし、前のフレームディスクリプタの RDES0 ステータスビットを開放します。次に DMA は Receive Interrupt (DMA Register 5[6])をセットします。DMA がホストによって所有されていることを示すフラグが付いているディスクリプタを検出するまで、同じプロセスが繰り返されます。このようなディスクリプタが検出されたとき、受信プロセスは Receive Buffer Unavailable (DMA Register 5[7])をセットし、Suspend ステートに入ります。受信リスト内の位置は保持されます。

受信プロセスの一時停止

受信プロセスが Suspend ステートのときに新しい受信フレームが到着した場合、DMA はホストメモリの現在のディスクリプタを再び取り込みます。ディスクリプタが現在 DMA によって所有されている場合、受信プロセスは再び Run ステートに入り、フレームの受信を開始します。ディスクリプタが依然としてホストによって所有されている場合、デフォルトでは DMA は現在のフレームを MTL 受信 FIFO の最上位で廃棄し、受信されなかったフレームカウンタを増分します。MTL 受信 FIFO に 2 つ以上のフレームが格納されている場合、このプロセスが繰り返されます。Operation Mode レジストの bit24 (DFF)をセットすることによって、MTL 受信 FIFO の最上位のフレームの廃棄またはフラッシュを防止できます。このとき、受信プロセスは Receive Buffer Unavailable ステータスをセットし、Suspend ステートに戻ります。

3.3.5 割込み

割込みは種々のイベントにより発生します。DMA Register 5 は、割込みの原因になるすべてのビットを含んでいます。DMA Register 7 は、割込みの原因になる各イベントのイネーブルビットを含んでいます。DMA Register 5 で記述されているように、割込みには正常割込みと異常割込みの 2 つの種類があります。割込みは、対応するビット位置に 1 を書き込むことによってクリアされます。グループ内のすべてのイネーブルされているビットがクリアされたとき、対応するサマリビットがクリアされます。両方のサマリビットがクリアされたとき、割込み信号 INT_SBD がデアサートされます。Ethernet-MAC が割込みをアサートした原因である場合、DMA Register 5 の GLI, GMI, GPI のいずれかのビットが 1 にセットされています。

＜注意事項＞

- *DMA Register 5 は(割込み)ステータスレジスタです。割込み信号(INT_SBD)がこのステータスレジスタの何らかのイベントによってアサートされるのは、DMA Register 7 で対応するイネーブルビットがセットされている場合だけです。*

割込みは待ち行列制御を行いません。ドライバが応答する前に別の割込みイベントが発生した場合に、追加の割込みは発生しません。例えば、Receive Interrupt (DMA Register 5[6])は、1 つ以上のフレームがホストバッファへ転送されたことを示します。ドライバは DMA によって所有されるすべてのディスクリプタを、最後の記録位置から最初の記録位置までスキャンしなければなりません。

同時に複数の割込みイベントが発生した場合でも割込みは 1 回だけ生成されます。ドライバは割込みの原因を調べるために DMA Register 5 をスキャンしなければなりません。ドライバが DMA Register 5 の該当ビットをクリアした後は、新しい割込みイベントが発生するまで、割込みが再度生成されることはありません。例えば、コントローラが Receive Interrupt (DMA Register 5[6])を生成し、ドライバが DMA Register 5 のリードを開始します。次に Receive Buffer Unavailable (DMA Register 5[7]) が発生します。ドライバは Receive Interrupt をクリアします。この時点でも、Receive Buffer Unavailable 割込みがアクティブまたは保留中であるために INT_SBD 信号はデアサートされません。

Receive Interrupt (DMA register 5[6])の柔軟なコントロールのために、割込みタイマ("4.42. DMA Register 9 (RIWTR) "を参照) が提供されています。この割込みタイマは、非 0 値にプログラムされていて、DMA Register 7 で対応する受信割込みがイネーブルされていて、対応する DIC(RDES1[31])で割込みがディセーブルされている場合、受信 DMA が受信フレームのシステムメモリへの転送を完了したとき、受信割込みをアサートさせず、カウントを開始します。このタイマがタイムアウトすると、RI ビットがセットされ割込みがアサートされます。このタイマは、そのディスクリプタに対して DIC(RDES1[31])で割込みがイネーブルされている場合は、フレームがメモリへ転送されても、カウントを開始せず停止したままで、割込みを発生させません。

3.3.6 DMA へのエラー応答

DMA チャネルによって開始されるすべてのデータ転送で、バスエラーが発生した場合、その DMA はすべての動作を停止し、エラービットと Status register (DMA Register 5)の Fatal Bus Error ビットを更新します。その場合、DMA コントローラにソフトリセットまたはハードリセットを行い、DMA が再初期化された後にのみ、動作を再開できます。

3.3.7 CRC

DMA は、Ethernet フレームの FCS フィールドの CRC を生成します。DMA は Ethernet フレームの FCS フィールドの 32 ビット CRC を計算します。エンコードは下記の生成多項式によって定義されます。

$G(x) = x32 + x26 + x23 + x22 + x16 + x12 + x11 + x10 + x8 + x7 + x5 + x4 + x2 + x + 1$

3.4 チェックサムエンジン

TCP や UDP などの通信プロトコルはチェックサムフィールドを導入しており、それを使ってネットワーク上で送信されるデータの完全性が判定されます。Ethernet の最も広範な利用方法は、IP データグラム上で TCP および UDP をカプセル化することであるため、GMAC は、オプションの Checksum Offload Engine (COE)を導入し、チェックサム計算, 送信パスへの挿入, 受信パスでのエラー検出をサポートしています。

3.4.1 送信チェックサムオフロードエンジン

送信チェックサムオフロードエンジン (COE) は 2 つのタイプのチェックサム計算と挿入をサポートします。CIC ビット (TDES0[23:22])をセットすることによって、各フレームについてこのチェックサムエンジンをコントロールできます。

IP ヘッダチェックサムエンジン

IPv4 データグラムでは、ヘッダフィールドの完全性は 16 ビットのヘッダチェックサムフィールド(IPv4 データグラムの 11, 12 番目のバイト)によって示されます。COE は Ethernet フレームの Type フィールドの値が 0x0800 で、IP データグラムの Version フィールドの値が 0x4 のときに IPv4 データグラムを検出します。送信時、フレームのチェックサムフィールドは、計算時は無視され、計算値に置換されます。IPv6 ヘッダにはチェックサムフィールドがないため、COE は IPv6 ヘッダフィールドを変更しません。この IP ヘッダチェックサム計算の結果は送信ステータスの IP Header Error ステータスビット(TDES0[16])によって示されます。このステータスビットは、Ethernet フレームの Type フィールドと IP ヘッダの Version フィールドの値が一致しないとき、または Ethernet フレームに IP header Length フィールドで指定されている十分なデータがないときにセットされます。
つまり、このビットは以下のいずれかの状況で IP ヘッダエラーがアサートされているときセットされます。

IPv4 データグラム：

- Ethernet type が 0x0800 で、IP ヘッダの Version フィールドが 0x4 でない
- IPv4 Header Length フィールドが、0x5 (20 バイト)未満の値を示す
- 合計フレーム長が IPv4 Header Length フィールドで指定されている値より小さい

IPv6 データグラム：

- Ethernet タイプが 0x86DD で、IP header Version フィールドが 0x6 でない
- IPv6 ヘッダ(40 バイト)または拡張ヘッダ(拡張ヘッダの対応する Header Length フィールドで指定されています)が完全に受信される前にフレームが終了した。

COE は、このような IP ヘッダエラーを検出したときでも、Ethernet フレームの Type フィールドが IPv4 ペイロードを示している場合は IPv4 ヘッダチェックサムを挿入します。

TCP/UDP/ICMP チェックサムエンジン

TCP/UDP/ICMP チェックサムエンジンは、IPv4 または IPv6 ヘッダ(拡張ヘッダを含む)を処理し、カプセル化されたペイロードが TCP, UDP, ICMP のいずれかを判定します。

<注意事項>

- *TCP, UDP, ICMP/ICMPv6 以外のペイロードについては、このチェックサムエンジンはバイパスされ、フレームでそれ以上の変更は行われません。*
  断片化された IP フレーム(IPv4 または IPv6)とセキュリティ機能を備えた IP フレーム(認証ヘッダ, カプセル化されたセキュリティペイロードなど)は、このエンジンでは処理されないため、バイパスしなければなりません。つまり、このようなフレームではペイロードチェックサムの挿入をイネーブルしてはいけません。
  IPv6 認証ヘッダを使用する Ethernet 環境下では、チェックサムエンジンはイネーブルできません。送信拡張ディスクリプタ 0 の CIC フィールドは、00 または 01 にしてください。

TCP, UDP または ICMP ペイロードのチェックサムが計算され、ヘッダ内の対応するビットに挿入されます。このエンジンは下記の 2 つのモードで動作できます。

- 最初のモードは、TCP, UDP または ICMPv6 擬似ヘッダに対するチェックサム計算を行わず、入力フレームの Checksum フィールドに (その計算結果が) 存在すると見なします。エンジンは、Checksum フィールドの値を含めてチェックサム計算を行い、Checksum フィールドを最終的なチェックサムで更新します。
- 2 番目のモードは、エンジンは Checksum フィールドを無視します。TCP, UDP または ICMPv6 擬似ヘッダデータをチェックサム計算に含め、Checksum フィールドを最終的なチェックサムによって上書きします。

＜注意事項＞

- *IPv4 上の ICMP 上のパケットは、擬似ヘッダは定義されないため、ICMP パケットの Checksum フィールドはどちらのモードでも常に 0x0000 でなければなりません。この値が 0x0000 でない場合、正しくないチェックサムがパケットに挿入されることがあります。*

この操作の結果は、TDES0[12]の IP Payload Error ビットによって示されます。このエンジンは、フレームが Store-and-Forward モードで GMAC トランスミッタに転送され、end-of-frame が FIFO に書き込まれていないことを検出したとき、または IP ヘッダの Payload Length フィールドで指定されているバイト数を受信する前にパケットが終了したときに、IP Payload Error をセットします。パケットが指定されているペイロードの長さより長い場合、COE はそれらを埋め込みバイトとして無視し、エラーを報告しません。このエンジンは、最初のタイプのエラーを検出したときには TCP, UDP または ICMP ヘッダを変更しません。2 番目のエラー・タイプでは、計算によって得られたチェックサムを対応するヘッダフィールドへ挿入します。

＜注意事項＞

- *TCP, UDP, ICMP のチェックサムは完全なフレームについて計算され、次にその対応するヘッダフィールドに挿入されます。この要件のため、この機能は送信 FIFO が Store-and-Forward モードに構成されている場合(DMA Register 6 TSF=1)にのみイネーブルされます。GMAC が Threshold (cut-through) モードに構成されている場合、送信 COE はバイパスされます。*

  Store-and-Forward モードであっても、TCP, UDP, ICMP のチェックサムを挿入する際は、下記のバイト数の条件を満たすフレームに対してだけイネーブルする必要があります。

  フレーム長 (Byte) ＜ 2036 (Byte) － PBL ×4 (Byte)

  PBL は、DMA Register 0 で指定する送信時のバースト長になります。送信時、PBL ×4 の設定可能最大値は、512 のため、通常の Ethernet フレームであれば問題はありません。しかしこの条件を満たさないフレーム長の場合、チェックサム挿入エンジンは失敗し、その結果、不適切なリカバリのために先行するすべてのフレームが破損する可能性があります。

3.4.2 受信チェックサムオフロードエンジン

受信 Ethernet フレームの中の IPv4 フレームと IPv6 フレームの両方が検出され、データの完全性チェックを行います。このモジュールは GMAC Configuration レジスタの IPC ビットをセットすることで有効になります。GMAC レシーバは、受信 Ethernet フレームの Type フィールドの値 0x0800 または 0x86DD をチェックし、それぞれ IPv4 または IPv6 フレームを識別します。この識別は VLAN タグ付フレームにも適用されます。受信チェックサムオフロードエンジンは、IPv4 ヘッダチェックサムを計算し、それが受信 IPv4 ヘッダのチェックサムと一致することをチェックします。指定されたペイロードタイプ(Ethernet Type フィールド)と IP ヘッダバージョンが一致しない、または受信したフレームに IPv4 ヘッダの Length フィールドで指定されている十分なバイト数がない (または IPv4 または IPv6 ヘッダのバイト数が 20 未満である) ときに IP Header Error ビットがセットされます。また、受信 IP データグラム(IPv4 または IPv6)の中の TCP, UDP または ICMP ペイロードを識別し、TCP, UDP または ICMP 仕様で定義されている方法でそのペイロードのチェックサムを適切に計算します。このエンジンは、チェックサム計算のための TCP/UDP/ICMPv6 擬似ヘッダバイトを含んでおり、受信した Checksum フィールドが計算値と一致することをチェックします。この演算の結果は受信ステータスワードの Payload Checksum Error ビットに示されます。また、このステータスビットは TCP, UDP または ICMP のペイロード長が IP ヘッダで指定されている想定ペイロード長と一致しない場合にもセットされます。"TCP/UDP/ICMP チェックサムエンジン"で記述しているように、このエンジンは、断片化された IP データグラム, セキュリティ機能を備えた IP データグラム, および TCP, UDP, ICMP 以外のペイロードをバイパスします。この情報は(チェックサムエンジンがバイパスされるかどうかに関わりなく)受信ステータスで与えられます。

＜注意事項＞

- *GMAC が IPv6 Routing 拡張ヘッダの Segment Left フィールドに 0 以外の値を持ったパケットを受信した場合、チェックサムエンジンは、受信パケットをバイパスし、無視します。*
  IPv6 認証ヘッダを使用する Ethernet 環境下では、チェックサムエンジンはイネーブルできません。GMAC Register0(MCR)の IPC(bit10)は、0 にしてください。

3.5 Energy Efficient Ethernet

Energy Efficient Ethernet (EEE)は、IEEE 802.3 Media Access Control (MAC)サブレイヤと物理レイヤを Low-Power Idle (LPI)モードで動作することを可能にするオプションの動作モードです。
LPI モードでは、送受信するデータがないときに通信装置の機能の一部をオフに切り換えることによって電力の消費を削減できます。リンクの両側のシステムで、リンク利用率が低い時間に一部の機能をディセーブルすることによって消費電力を削減できます。GMAC はシステムが LPI モードに入るか LPI モードを終了するかをコントロールし、それを PHY に知らせます。
EEE は、リンクパートナーが EEE をサポートするかどうかを決定するネゴシエーション方法を規定し、次に両方の装置に共通のパラメータセットを選択します。

LPI モードは、GMAC が MII を使い、全二重モードでのみサポートされます。

送信パス機能

送信パスで、ソフトウェアは、GMAC Register 12 (LPI Control and Status Register)の bit16 (LPIEN)をセットすることによって、GMAC に対して送信を停止し、LPI プロトコルを開始するよう指示しなければなりません。GMAC は進行中の送信を完了し、送信ステータスを生成します。次に、Interframe gap (IFG)の間、IDLE パターンの代わりに LPI パターンの送信を開始します。

PHY を LPI ステートにするため、GMAC は以下のタスクを実行します。

1. TX_EN をディアサートします。
2. TX_ER をアサートします。
3. TXD[3:0]を 0x1 (100 Mbps の場合)にセットします。
   GMAC は PHY が LPI ステートに入っている全期間にわたって TX_EN, TX_ER, TXD 信号を同じ状態に維持します。
4. GMAC Register 12 (LPI Control and Status Register)の bit0 を更新し、割込みを生成します。

PHY の LPI ステートを終了する、つまりソフトウェアが LPIEN ビットをリセットすると、GMAC は以下のタスクを実行します。

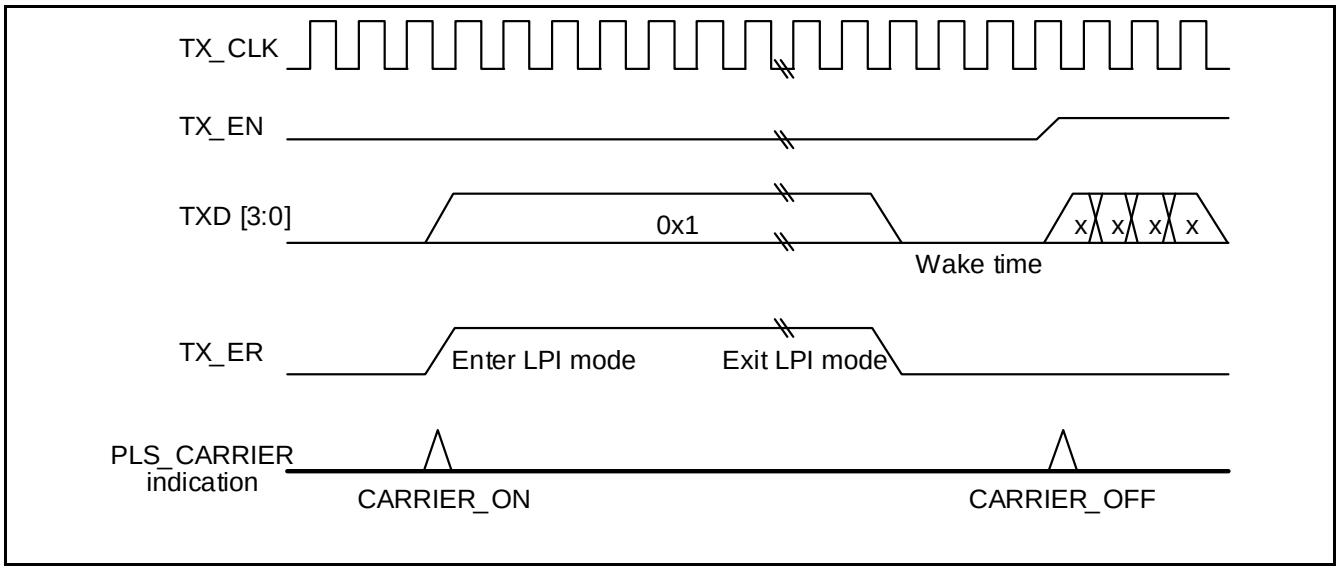
1. LPI パターンの送信を停止し、IDLE パターンの送信を開始します。
2. LPI TW TIMER をスタートします。
   GMAC は、PHY のウェイクアップ時間が経過するまで送信を開始できません。オート・ネゴシエーション・ウェイクアップ間隔は、GMAC Register 13 (LPI Timers Control Register)の bit[15:0] (TWT: LPI TW TIMER)にプログラムされます。
3. LPI 終了ステータス GMAC Register 12 (LPI Control and Status Register)の bit1 を更新し、割込みを生成します。

<注意事項>

- *LPI モード中に TX_CLK の供給が停止される場合、GMAC Register 12 (LPI Control and Status Register)の bit19 (LPITXA)を使用できません。Energy Efficient Ethernet 標準(802.3az)によると、PHY は MII(10 Mbps/100 Mbps)モードで LPI ステートの間、TXCLK クロックを停止できません。*
  GMAC が送信 LPI モードで、TX_CLK の供給が停止しているとき、アプリケーションは TX_CLK ドメインに同期されている CSR レジスタに書き込んではいけません。
  GMAC が LPI モードで、ホストがソフトリセットまたはハードリセットを発行したとき、GMAC トランスミッタは LPI モードを終了します。

Figure 3-5 に、LPI モード遷移中の TX_EN, TX_ER および TXD[3:0]信号の動作を示します。

Figure 3-5 LPI の遷移(送信)



受信バス機能

受信パスでは、PHY がリンクパートナーから LPI ステートに入る信号を受信したとき、PHY および GMAC は以下のタスクを実行します。

1. PHY が RX_ER をアサートします。
2. PHY が RXD[3:0]を 0x1 にセットします。
3. PHY が RX_DV をデアサートします。
   PHY は LPI ステートに入っている全期間、RX_ER, RXD, RX_DV 信号を同じステートに保持します。
4. GMAC が、GMAC Register 12 (LPI Control and Status Register)の bit2 (RLPIEN)を更新し、即座に割込みを生成します。

PHY がリンクパートナーから LPI ステートを終了する信号を受信したとき、PHY および GMAC は以下のタスクを実行します。

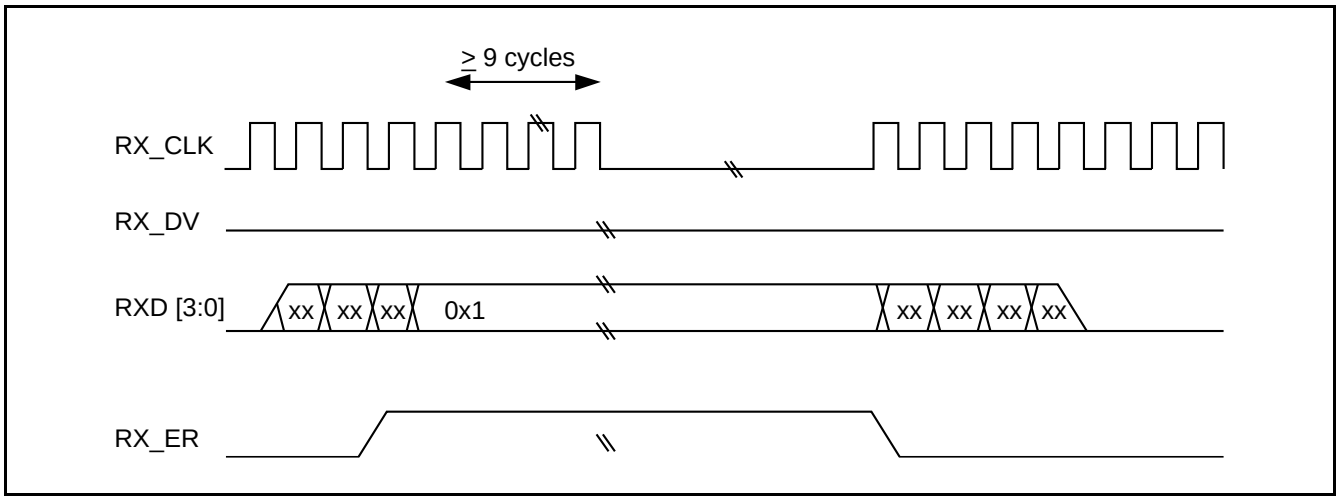
1. PHY が RX_ER をデアサートし、通常のフレーム間ステートに戻ります。
2. GMAC が、GMAC Register 12 (LPI Control and Status Register)の bit3 (RLPIEX)を更新し、割込みを生成します。この場合、割込み信号 INT_LPI (synchronous to Rx clock)もアサートされます。

Figure 3-6 に、LPI モード遷移中の RX_ER, RX_DV および RXD[3:0]信号の動作を示します。

<注意事項>

- *MDIO を通じて書き込まれる PHY レジスタの RX_CLK_stoppable ビットがアサートされている場合、Figure 3-6 に示すように、PHY が GMAC に LPI を指示しているとき、PHY は LPI ステートの開始から 9 クロックサイクル以上後に、随時 RX_CLK の供給を停止できます。*
  GMAC が LPI モードのときに、ホストがソフトリセットまたはハードリセットを発行した場合、GMAC レシーバは、リセット中に LPI モードを終了します。リセットがデアサートされた後にまだ LPI パターンが受信される場合、GMAC レシーバは再び LPI ステートに入ります。
  受信 LPI モードで RX クロックが停止した場合、ホストは RX クロックドメインに同期されている CSR レジスタへの書込みを行うことはできません。

Figure 3-6 LPI 遷移(受信)



LPI タイマ

GMAC トランスミッタは、下記の 2 つのタイマを動作させます。タイマには GMAC Register 13 (LPI Timers Control Register) からそれぞれの値がロードされます。

■ LPI LS TIMER：

LPI LS TIMER は、リンクステータスが Up になって以降の経過時間(ms 単位)をカウントします。このタイマは、リンクが Down になるたびにクリアされ、リンクが Up になると、ソフトウェアによってプログラムされているターミナルカウントに到達するまで、再びインクリメントします。PHY インタフェースは、ターミナルカウントに到達しない限り LPI パターンをアサートしません。これによって、リモートステーションとのリンクが確立されてから LPI パターンがアサートされるまでの最小限の時間が確保されます。この時間は IEEE standard 802.3-az-2010 によって 1 秒と規定されています。LPI LS TIMER は 10 ビット幅です。したがってソフトウェアは最大 1023 ms までプログラミングできます。

■ LPI TW TIMER：

LPI TW TIMER は LPI がデアサートされて以降の経過時間(µs 単位)をカウントします。このタイマのターミナルカウントは、Transmit TW を確保するための値で、オート・ネゴシエーション後、GMAC が通常の送信動作を再開できるまでの時間です。GMAC は LPI TWTIMER をマイクロ秒単位でサポートします。LPI TW TIMER は 16 ビット幅です。したがってソフトウェアは最大 65535 µs までプログラミングできます。

PHY からの TX_CLK 供給が停止する場合は、GMAC Register 12 (LPI Control and Status Register)の bit17 (PLS)を 0 にプログラムします。それによって内部タイマがリセットされます。LPI LS TIMER または LPI TW TIMER がカウント中に、TX_CLK 周波数が変化すると、不適切なタイムアウトが発生することがあります。

3.6 MAC マネジメントカウンタ

MAC マネジメントカウンタ (MMC)は、受信および送信フレームに関する統計収集のためレジスタのセットです。レジスタは、レジスタの動作をコントロールするためのコントロールレジスタ, 収集された割込みを表示する 3 つの 32 ビットレジスタ, 割込みをマスクする 3 つの 32 ビットレジスタおよび統計カウンタレジスタで構成されています。

MMC の動作

MMC モジュール内のレジスタは AHB スレーブ・インタフェースを通じてアプリケーションからアクセスできます。各レジスタは、32 ビット幅です。ワードアクセス (32 ビット幅) のアクセスのみが許容されます。
それぞれの統計カウンタの機能(カウントする内容)を「4.48. MMC Register List の Table 4-1 Ethernet-MAC レジスタ一覧」にて、説明しています。
受信 MMC カウンタは、Address Filter (AFM)ブロックによってフレームが渡されたときに更新されます。AFM モジュールによって破棄されたフレームの統計は、それらのフレームが 6 バイト未満の小さなフレーム(DA バイトが完全には受信されない)でない限り更新されません。受信 IPC カウンタは、送信先アドレスフィルタを通過したフレームに対してのみ更新されます。受信 IPC カウンタは、受信 Ethernet フレーム内のカプセル化された IPv4, IPv6, TCP, UDP, または ICMP ペイロードに関する統計を収集します。
MMC コントロールレジスタにより、マネジメントカウンタのオペレーションモードを設定できます。カウンタが規定値に達するイベントにより、割込みを発生させることができます。MMC 受信割込みレジスタ, MMC 送信割込みレジスタ, MMC 受信チェックサムオフロード割込みレジスタにより、割込みの状態を確認できます。MMC 受信割込みマスクレジスタ, MMC 送信割込みマスクレジスタ, MMC 受信チェックサムオフロード割込みマスクレジスタにより、各割込みを個別にマスクできます。
MMC レジスタの命名規則は以下のとおりです。

- tx: 送信に関連付けられたカウンタを示す接頭辞または接尾辞
- rx: 受信に関連付けられたカウンタを示す接頭辞または接尾辞
- _g: 正常なフレームのみをカウントするレジスタを示す接尾辞
- _gb: フレームが正常か不良かに関わりなくフレームをカウントするレジスタを示す接尾辞

送信が正常に完了した場合、送信されたフレームは「正常」と見なされます。つまり、以下のいずれかのエラーのためにフレーム送信が中止された場合以外は、送信フレームは正常なフレームです。

- Jabber Timeout
- No Carrier/Loss of Carrier
- Late Collision
- Frame Underflow
- Excessive Deferral
- Excessive Collision

受信フレームは、以下のどのエラーも存在しない場合、「正常な」フレームと見なされます。

- CRC error
- Runt Frame (64 バイト未満)
- Alignment error (10/100 Mbps のみ)
- Length error (Type フレーム以外)
- Out of Range (Type フレームのみ、最大サイズより長い)
- MII_RXER Input error

最大フレームサイズは以下のようにフレームの種類により異なります。

- Untagged frame maxsize = 1518
- VLAN Frame maxsize = 1522
- Jumbo Frame maxsize = 9018
- JumboVLAN Frame maxsize = 9022

3.7 Station Management Agent

Station Management Agent (SMA)モジュールは、2 ワイヤ Station Management Interface を通じて任意の PHY レジスタにアクセスできるようにします。このインタフェースは、最大 32 個の PHY のアクセスをサポートします。アプリケーションは 32 個の PHY から 1 つの PHY と、その PHY 内の 32 個のレジスタから 1 つのレジスタを選択し、コントロールデータの送信, またはステータス情報の受信ができます。

機能

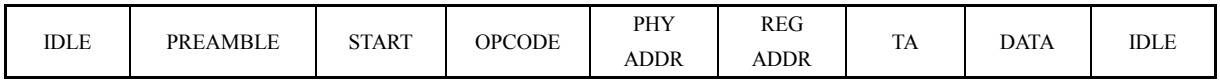
マネジメント動作で使用する MDC クロック信号は、SYS_CLK からの分周クロックです。分周比は GMAC Register 4 の MDC クロック範囲の設定で決定されます。SYS_CLK の周波数に基づき、MDC クロックを Table 3-3 のように設定します。

Table 3-3 MDC の生成

GMAC Register 4 の CR[3:0]	SYS_CLK	MDC クロック
0000	60 MHz ～ 100 MHz	SYS_CLK/42
0001	100 MHz ～ 150 MHz	SYS_CLK/62
0010	20 MHz ～ 35 MHz	SYS_CLK/16
0011	35 MHz ～ 60 MHz	SYS_CLK/26
0100	150 MHz ～ 250 MHz	SYS_CLK/102
0101	250 MHz ～ 300 MHz	SYS_CLK/124
0110, 0111	予約	

マネジメント動作の入出力データは、MDIO 信号に入出力されます。MDIO のフレーム構造を Figure 3-7 に示します。

Figure 3-7 MDIO のフレーム構造



IDLE: Hi-Z 状態になります。 (MDC 上にクロックはありません)
PREAMBLE: 32 ビットの連続した 1
START: フレームの開始は 01
OPCODE: 読出しでは 10, 書込みでは 01
PHY ADDR: 32 個の PHY から 1 つを選択する 5 ビットアドレス
REG ADDR: 選択した PHY 内のレジスタアドレス
TA: 書込み時、Ethernet-MAC が 10 をドライブします。
読出し時、1 ビット目は Hi-Z 状態、2 ビット目は PHY が 0 をドライブします。
DATA: 書込み時、Ethernet-MAC がライトデータをドライブします。
読出し時、PHY がリードデータをドライブします。

マネジメントライト動作

ユーザが GMII Write ビット(GW)および Busy ビット(GB)(「4.4. GMAC Register 4 (GAR)」を参照)をセットすると、SMA モジュールは、PHY アドレス, PHY 内のレジスタアドレス, およびライトデータを使用し、PHY レジスタへのライト動作を開始します。トランザクションの進行中にアプリケーションは GMII アドレスレジスタの内容および GMII データレジスタを変更できません。ライト動作が完了した後、Busy ビット(GB)がリセットされます。
SMA モジュールは、SYS_CLK をプログラムされたクロック分周比(GMII アドレスレジスタの CR ビット)によって分周し、このインタフェースの MDC クロックを生成します。GMAC はフレームの持続時間全体にわたって MDIO ラインをドライブします。ライト動作のフレームフォーマットを Figure 3-8 に示します。

Figure 3-8 ライト時のフレームフォーマット

IDLE	PREAMBLE	START	OPCODE	PHY ADDR	REG ADDR	TA	DATA	IDLE
Z	1111…11	01	01	AAAAA	RRRRR	01	DDD…DDD	Z

マネジメントリード動作

ユーザが GMII Busy ビット(GB)(「4.4. GMAC Register 4 (GAR)」を参照)をセットし、GMII Write ビットが 0 のとき、SMA モジュールは、PHY アドレスおよび PHY 内のレジスタアドレスを使用して、PHY レジスタのリード動作を開始します。トランザクションが進行中にアプリケーションは GMII アドレスレジスタの内容および GMII データレジスタを変更できません。リード動作が完了した後、Busy ビットがリセットされます。さらに、PHY から読み出されたデータによって GMII データレジスタが更新されます。リード時のフレームフォーマットを Figure 3-9 に示します。

Figure 3-9 リード時のフレームフォーマット

IDLE	PREAMBLE	START	OPCODE	PHY ADDR	REG ADDR	TA	DATA	IDLE
Z	1111…11	01	10	AAAAA	RRRRR	Z0	DDD…DDD	Z

3.8 IEEE1588

IEEE1588 機能について説明します。

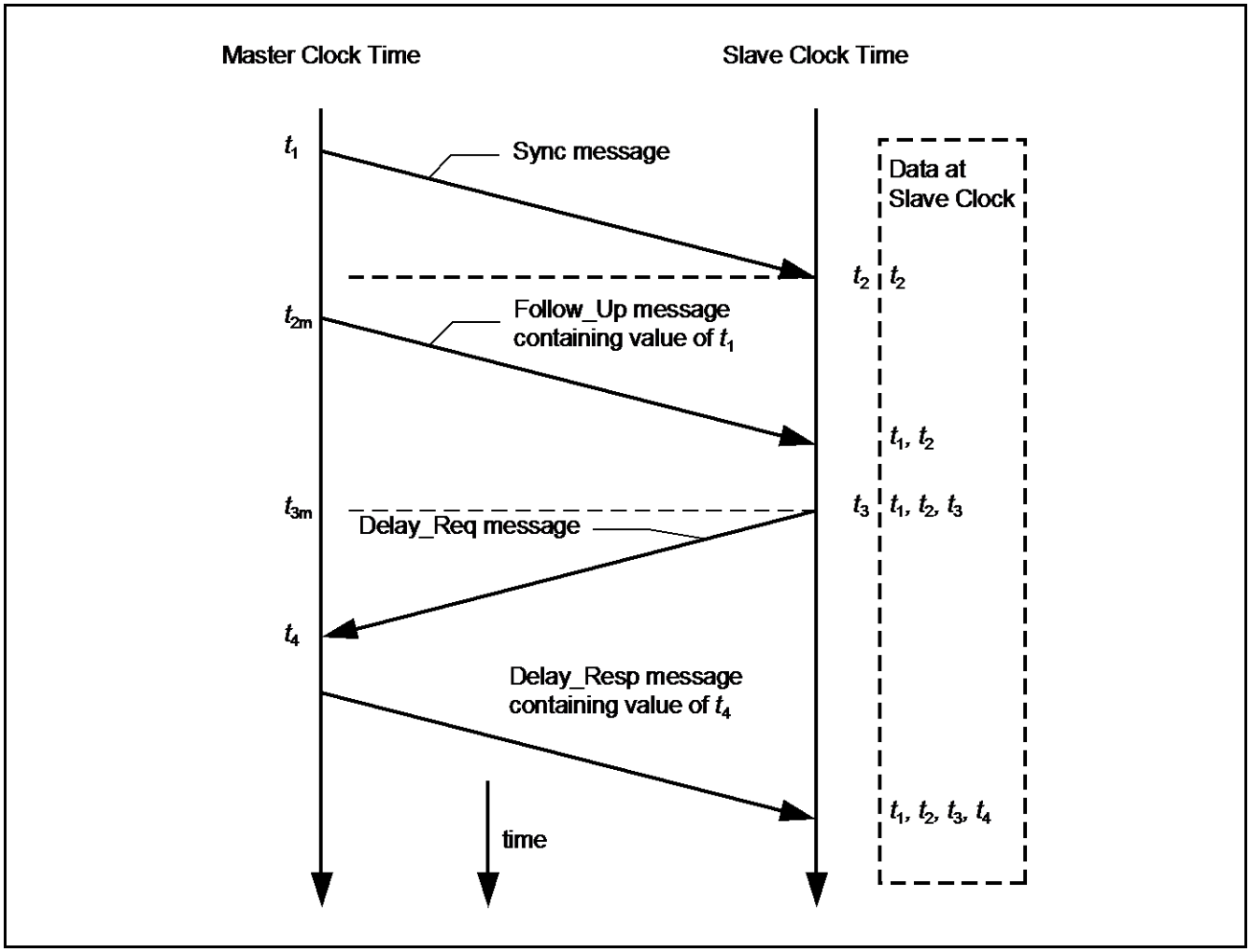
IEEE1588 の概要

IEEE 1588 標準は、ネットワーク通信, ローカル・コンピューティング, 分散オブジェクトなどの技術を活用する計測および制御システムにおけるクロックの正確な同期を可能にするプロトコルを定義しています。このプロトコルは、Ethernet またはそのほかの、マルチキャスト・メッセージングをサポートするローカルエリアネットワークによって通信するシステムに適用されます。このプロトコルは精度, 分解能, 安定性が異なる種々のクロックを含む異種システムの同期を可能にします。このプロトコルは、最小限のネットワークおよびローカルのクロック・コンピューティング・リソースを使って、システム全体にわたってマイクロ秒以下の同期精度をサポートします。

Precision Time Protocol (PTP)と呼ばれるメッセージベースのプロトコルが UDP/IP 上で転送されます。タイミング/クロック情報の分配においては、システムまたはネットワークはマスタノードとスレーブノードに分類されます。このプロトコルで PTP メッセージを交換することによって

スレーブノードをマスタノードに同期する技術を Figure 3-10 に示しています。

Figure 3-10 ネットワークの時間同期



1. マスタは PTP Sync メッセージをすべてのノードにブロードキャストします。Sync メッセージにはマスタのリファレンス時間情報が含まれます。このメッセージがマスタのシステムを出る時刻は t1 で、これは Ethernet ポートのために PHY インタフェースでキャプチャされなければなりません。
2. スレーブは Sync メッセージを受信し、そのタイミングリファレンスを使って正確な時刻 t2 をキャプチャします。
3. 次にマスタはスレーブに t1 情報を含む Follow_up メッセージを送信します(この情報は後で使用されます)。
4. スレーブはマスタに、このフレームが MII を出る正確な時刻 t3 を記載した Delay_Req メッセージを送信します。
5. マスタはこのメッセージを受信し、それがシステムに入る正確な時刻 t4 をキャプチャします。
6. マスタは t4 情報を Delay_Resp メッセージに入れてスレーブに送信します。
7. スレーブは t1, t2, t3, t4 の 4 つの値を使ってそのローカル・タイミングリファレンスをマスタのタイミングリファレンスに同期します。

プロトコルの実装の大部分はソフトウェアで、UDP の上のレイヤで行われます。しかし、上記のように、特定の PTP パケットが Ethernet ポートに入る、またはそのポートを出る正確な時刻をキャプチャするためにハードウェア・サポートが必要とされます。高精度の PTP を適切に実装するには、このタイミング情報がキャプチャされ、ソフトウェアへ返されなければなりません。

リファレンスタイミング・ソース

Ethernet-MAC は、リファレンスクロックを入力として、内部的にリファレンス時間(「システム時間」ともよばれます)を生成し、それを使ってタイムスタンプをキャプチャします。システム時間の生成, 更新, 変更については「システム時間レジスタモジュール」を参照してください。
時間のスナップショットを生成するために、64 ビット形式のリファレンスタイムを必要とします(2 つの 32 ビット・チャネルに分割され、上位 32 ビットが秒単位の時間, 下位 32 ビットが ns 単位の時間を示します)。IEEE 1588-2008 標準は時間の上位フィールドを 48 ビット幅と定義しています。タイムスタンプのフィールドは下記のとおりです。

- UInteger48- seconds フィールド
- UInteger32-nanoseconds フィールド

"seconds"フィールドはタイムスタンプの整数部で、秒単位です。"nanoseconds"フィールドはタイムスタンプの小数部で、ns 単位です。例：2.000000001 秒は secondsField = 0x0000_0000_0002, nanoSeconds = 0x0000_0001 で表されます。したがって、この形式での nanoseconds フィールドの最大値は 0x3B9A_C9FF((10e9-1) ns)となります。これはデジタルロールオーバ動作モードとして定義されます。また、値 0x7FFF_FFFF の後 nano-seconds フィールドがロールオーバし、seconds フィールドを増分するレガシーモードもサポートされます(精度は最大 0.466ns/ビット)。これはバイナリ・ロールオーバ・モードとして定義されます。これらのモードは、Time Stamp Control レジスタ"TSDB: Time Stamp Digital or Binary rollover control"ビットを使ってコントロールされます。
Ethernet-MAC に保持される時間は 64 ビット幅です。なぜなら、seconds レジスタの上位 16 ビットへのオーバフローは、130 年に 1 回しか発生しないからです。seconds フィールドの上位 16 ビットの値は GMAC レジスタからのみ取得できます。32 ビットの"seconds"フィールドでオーバフローが発生したときにそれを知らせるステータスビットが提供されています。
また、1 秒のインターバル(デフォルト)を示すために pulse-per-second 出力も供給されます。インターバルを変更するオプションも提供されています。詳細については「4.30. GMAC Register 459 (PPSCR)」を参照してください。

送信パス機能

フレームの SFD が PHY インタフェースに出力されたときに、タイムスタンプがキャプチャされます。フレームのタイムスタンプをキャプチャする必要があるか否かはフレームごとに制御可能です。つまり、各送信フレームにタイムスタンプをキャプチャする必要があるか否かを示すマークを付けることができます。
PTP フレームを識別するための処理は実行されません。フレーム単位のコントロールは、送信ディスクリプタのコントロールビットを通じて実行されます。

キャプチャされたタイムスタンプは、フレームのステータスが供給されるのと同様の方法で、アプリケーションに返されます。タイムスタンプとフレームの送信ステータスが供給されます。タイムスタンプは対応する送信ディスクリプタによってソフトウェアに返され、それによってタイムスタンプが自動的に特定の PTP フレームにコネクトされます。64 ビットのタイムスタンプ情報が TDES6 および TDES7 フィールドに書き戻され、TDES6 がタイムスタンプの下位 32 ビットを保持します。

受信パス機能

Ethernet-MAC は、対応する受信ディスクリプタによってタイムスタンプをソフトウェアに返します。64 ビットのタイムスタンプ情報が RDES6 および RDES7 フィールドに書き戻され、RDES6 がタイムスタンプの下位 32 ビットを保持します。タイムスタンプは、Last Descriptor ステータスフィールドが 1 (EOF マーカ)にセットされている受信ディスクリプタにのみ書き込まれます。タイムスタンプが生成できない場合(例えば、受信 FIFO オーバフローのため)、ディスクリプタ(RDES6 および RDES7)には"オール 1"パターンが書き込まれ、タイムスタンプが正しくないことを示します。ソフトウェアがコントロールレジスタ・ビットを使ってタイムスタンプをディセーブルしている場合、DMA は RDES6 および RDES7 を変更しません。

タイムスタンプ・エラー・マージン

IEEE 1588 仕様によると、タイムスタンプは PHY インタフェースで送信および受信フレームの SFD でキャプチャされなければなりません。リファレンスタイミング・ソースが PHY インタフェースのクロックと異なるため、非同期クロックドメイン間の情報転送によって小さなエラー・マージンが発生します。
送信パスではキャプチャおよびレポートされるタイムスタンプの最大エラー・マージンは 2 PTP クロックです。つまり、キャプチャされるタイムスタンプには、PHY インタフェース上で SFD が送信されてから 2 クロック以内のリファレンスタイム・ソースの値が与えられます。同様に、受信パスでは、エラー・マージンは 3 PHY インタフェースクロックと最大 2 PTP クロックの和となります。SFD データが PHY インタフェースに到達する前に 3 PHY インタフェースクロックに起因するエラー・マージンがシステム(またはリンク)に存在していたと想定すれば、この固定の遅延は無視できます。

リファレンスタイミングクロックの周波数範囲

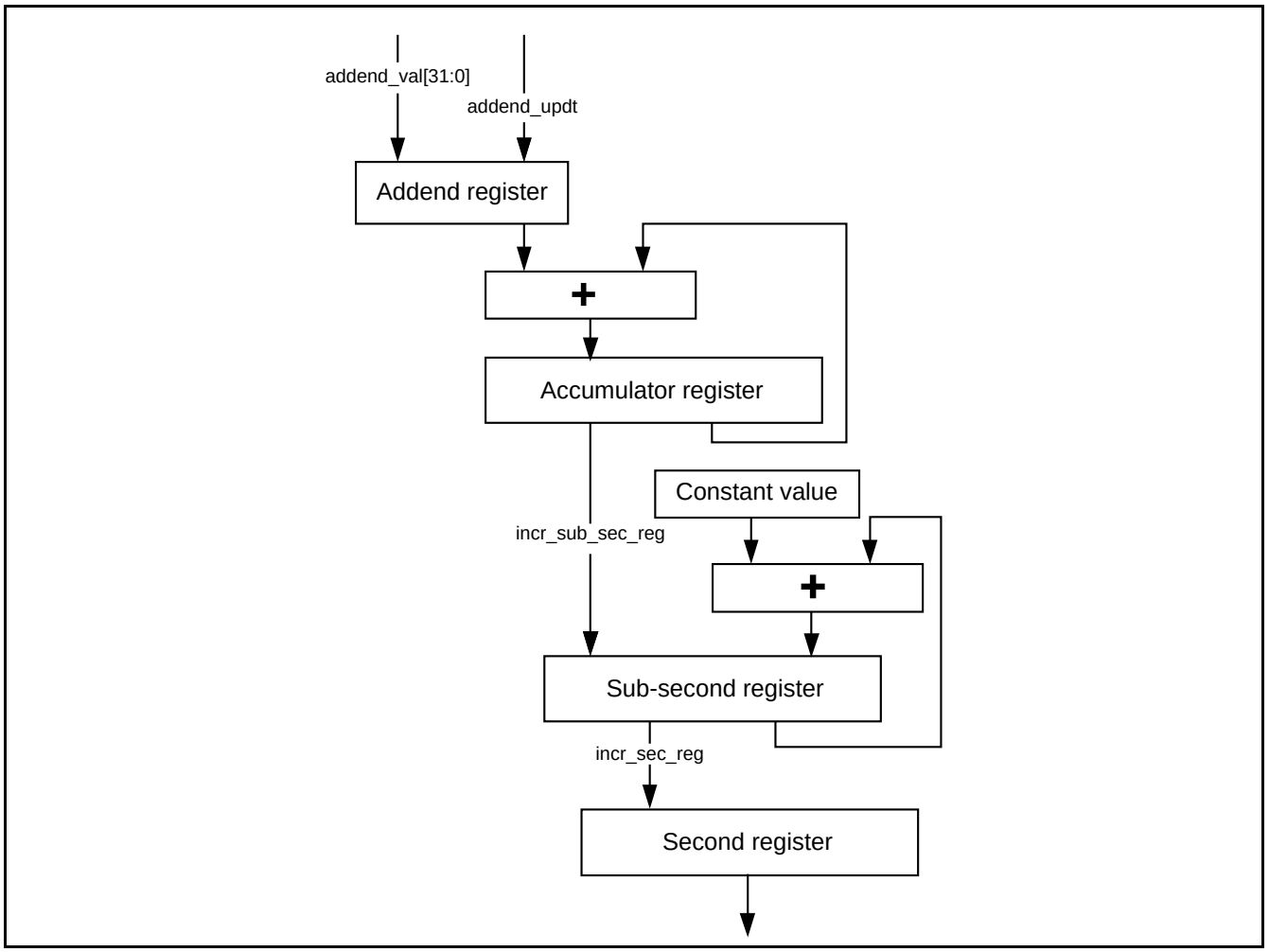
クロックドメイン間のタイムスタンプ情報の転送には非同期ロジックが関係するため、2 つの連続するタイムスタンプ・キャプチャの間に最小限の遅延が必要とされます。この遅延は PHY インタフェースおよび PTP クロックの両方の 3 クロックサイクルです。間隔がこれより短いと、Ethernet-MAC は 2 番目のフレームのタイムスタンプスナップショットを生成しません。最大 PTP クロック周波数は、リファレンスタイムの最大分解能(1 GHz では 1 ns)と PTP クロック上で動作するロジックで利用できるタイミング抑制によって制限されます。考慮すべきもう 1 つの要因は、リファレンスタイム・ソースの分解能(または「粒度」)が同期の精度を決定するということです。そのため、PTP クロックの周波数を高くするとシステムパフォーマンスが向上します。最小 PTP クロック周波数は 2 つの連続する SFD バイトの間で要求される時間に依存します。PHY インタフェースクロック周波数は IEEE 仕様によって固定されるため、適切な動作のために要求される最小 PTP クロック周波数は Ethernet-MAC の動作モードおよび動作速度に依存します。
100 Mbps 全二重動作では 2 つの SFD の間の最小間隔は 160 PHY インタフェースクロックです。(64 バイト・フレームのための 128 クロックと、最小 IFG のための 24 クロック, およびプリアンブルのための 8 クロック)
この例では、(3 × PTP) + (3 × PHY) ≦ 160 × PHY です。したがって、最小 PTP クロック周波数は約 0.5 MHz ((160 - 3) × 40ns ÷ 3 = 2093 ns)です。

システム時間レジスタモジュール

64 ビットの時間はこのモジュールに保持され、入力リファレンスクロック(PTP_CLK) (注：FM3 ファミリ マイコンの Ethernet-MAC は、入力リファレンスクロック(PTP_CLK)に AHB のシステムバスクロック：HCLK が接続されています。)を使って更新されます。この時間は送信または受信している Ethernet フレームのスナップショット(タイムスタンプ)を生成するためのソースとなります。
System Time カウンタは、Coarse correction (粗修正)方法を使って初期化または修正できます。この方法では、初期値または

オフセット値が Time Stamp Update レジスタに書き込まれます。初期化の場合は System Time カウンタに Time Stamp Update レジスタの値が書き込まれ、システム時間修正の場合はシステム時間にオフセット値が加算または減算されます。
Fine correction (微修正)方法では、スレーブクロック(PTP_CLK)のマスタクロックに対する周波数ドリフト(IEEE 1588 で定義されています)が、粗修正の場合のように 1 クロック内ではなく一定時間の間に修正されます。これは直線的な時間を維持するのに役立ち、PTP Sync メッセージ間隔間でリファレンス時間が大幅に変動する(ジッタを大きくする)ことはありません。このメソッドでは、アキュムレータは Figure 3-11 に示すように Addend レジスタの内容を集計します。アキュムレータが生成する算術的繰上げがシステムタイムカウンタを増分するためのパルスとして使われます。アキュムレータと Addend は 32 ビットレジスタです。したがって、アキュムレータは高精度周波数乗算器または除算器として動作します。このアルゴリズムを Figure 3-11 に示します。

Figure 3-11 微修正メソッドによるシステム時間の更新



System Time Update ロジックは 20ns の精度を実現するために 50 MHz のクロック周波数を必要とします。周波数分割はリファレンスクロック周波数の、要求されるクロック周波数に対する比です。したがって、例えばリファレンスクロック(PTP_CLK)が 66 MHz のとき、この比は 66 MHz/50 MHz = 1.32 となります。その結果、レジスタでセットするデフォルトの Addend 値は 2^{32}/1.32 または 0xC1F07C1F です。
リファレンスクロックが低い値にドリフトする場合、例えば 65 MHz の場合、この比は 65 / 50、つまり 1.3 となり、Addend レジスタで設定される値は 2^{32}/1.30 または 0xC4EC4EC4 です。クロックが高い値にドリフトする場合、例えば 67 MHz の場合、Addend レジスタは 0xBF0B7672 にセットされなければなりません。クロックのドリフトが 0 である場合、デフォルトの Addend の値 0xC1F07C1F (2^{32}/1.32)をプログラムしなければなりません。
ソフトウェアは、Sync メッセージを元にして周波数のドリフトを計算し、それに基づいて Addend レジスタを更新しなけれ

ばなりません。
最初に、スレーブクロックは、Addend レジスタに FreqCompensationValue0 をセットします。この値は以下のとおりです。

FreqCompensationValue0 = 2^{32} / FreqDivisionRatio

最初は、MasterToSlaveDelay が後続の Sync メッセージと同じであると想定し、下記のアルゴリズムを適用しなければなりません。いくつかの Sync サイクルの後に、周波数がロックします。このとき、スレーブクロックは正確な MasterToSlaveDelay 値を決定し、新しい値を使ってマスタと再同期できます。
このアルゴリズムは以下のとおりです。

- MasterSyncTimen の時刻にマスタはスレーブクロックに Sync メッセージを送信します。スレーブは、そのローカルクロックが SlaveClockTimen のときにこのメッセージを受信し、次のように MasterClockTimen を計算します。

MasterClockTimen = MasterSyncTimen + MasterToSlaveDelayn

- マスタクロックは現在の Sync サイクルをカウントします。MasterClockCountn は次の式によって求められます。

MasterClockCountn = MasterClockTimen - MasterClockTimenn-1 (Sync サイクル n と n - 1 で MasterToSlaveDelay が同じと想定します)

- スレーブクロックは現在の Sync サイクルをカウントします。SlaveClockCountn は次のように求められます。

SlaveClockCountn = SlaveClockTimen - SlaveClockTimen - 1

- 現在の Sync サイクルのマスタとスレーブのクロック・カウントの差 (ClockDiffCountn) は次の式によって求められます。

ClockDiffCountn = MasterClockCountn - SlaveClockCountn

- スレーブクロックの周波数スケーリング係数 (FreqScaleFactorn) は次の式によって求められます。

FreqScaleFactorn = (MasterClockCountn + ClockDiffCountn) / SlaveClockCountn

- Addend レジスタの周波数補正値 (FreqCompensationValuen) は次の式によって求められます。

FreqCompensationValuen = FreqScaleFactorn * FreqCompensationValuen - 1

理論的には、このアルゴリズムによって 1 つの Sync サイクルでロックが得られますが、ネットワーク伝搬遅延や動作条件の変動のために、いくつかのサイクルを必要とする場合があります。
このアルゴリズムは自動修正型です。何らかの理由でスレーブクロックの初期値が間違ったマスタからの値にセットされている場合、このアルゴリズムは、さらにいくつかの Sync サイクルを使ってそれを修正します。

PTP の処理およびコントロール

PTP メッセージの共通メッセージヘッダを Table 3-4 に示しています。この形式は IEEE 標準 1588-2008 に準拠しています。

Table 3-4 IEEE 1588-2008 で定義されているメッセージ形式

ビット		オクテット	オフセット
transportSpecific	messageType	1	0
予約	versionPTP	1	1
messageLength		2	2
domainNumber		1	4
予約		1	5
flagField		2	6
correctionField		8	8
予約		4	16
sourcePortIdentity		10	20
sequenceId		2	30
messageType field		1	32
logMessageInterva		1	33

PTP フレームの一部のフィールドは、タイプを検出し、生成するスナップショットをコントロールするために使用されます。これは Ethernet 上で直接に送信される PTP フレーム, UDP / IPv4 上で送信される PTP フレーム, UDP / IPv6 上で送信される PTP フレームで異なります。「IPv4 上の PTP フレーム」では、スナップショットの生成をコントロールするために使用するフィールドに関する情報を示します。

IPv4 上の PTP フレーム

Table 3-5 は、IEEE 1588 version 1 および 2 に対応する UDP over IPv4 上で送信される PTP パケットのスナップショットをコントロールするために照合されるフィールドに関する情報を示しています。タグ・フレームのオクテット位置のオフセットは 4 です。これは IEEE 1588-2008 標準の Annex D および Table 3-4 で定義されているメッセージ形式をベースとしています。

Table 3-5 コントロールとステータスのために要求される IPv4-UDP PTP フレームのフィールド

対応するフィールド	オクテット位置	対応する値	説明
MAC Frame Type	12, 13	0x0800	IPv4 データグラム
IP Version and Header Length	14	0x45	IP バージョンが IPv4
Layer 4 protocol	23	0x11	UDP
IP Multicast address	30, 31, 32, 33	0xE0, 0x00, 0x01, 0x81 0xE0, 0x00, 0x00, 0x6B	PTP-primarymulticast address allowed 224.0.1.129 224.0.1.130 224.0.1.131 224.0.1.132
UDP destination port	36, 37	0x013F, 0x0140	0x013F : PTP event message (*) 0x0140 : PTPgeneral message
PTP Control Field (IEEE version 1)	74	0x00, 0x01, 0x02, 0x03, 0x04	0x00 : SYNC 0x01 : Delay_Req 0x02 : Follow_Up 0x03 : Delay_Resp 0x04 : Management
PTP Message Type Field (IEEE version 2)	42	0x0, 0x1, 0x2, 0x3, 0x8, 0x9, 0xA, 0xB, 0xC, 0xD	0x0 : SYNC 0x1 : Delay_Req 0x2 : Pdelay_Req 0x3 : Pdelay_Resp 0x8 : Follow_Up 0x9 : Delay_Resp 0xA : Pdelay_Resp_Follow_Up 0xB : Announce 0xC : Signaling 0xD : Management
PTP version field	43 (nibble)	0x1, 0x2	0x1 : PTP version 1 サポート 0x2 : PTP version 2 サポート

(*) PTP イベント・メッセージは SYNC, Delay_Req (IEEE 1588 version 1 および 2)または Pdelay_Req, Pdelay_Resp (IEEE 1588 version 2 のみ)です。

IPv6 上の PTP フレーム

Table 3-6 は、PTP パケットのスナップショットをコントロールするために照合されるフィールドに関する情報を示しています。この情報は、IEEE 1588 version 1 および 2 に対応する UDP over IPv6 上で送信されます。タグ・フレームのオクテット位置のオフセットは 4 です。これは IEEE 1588-2008 標準の Annex D および Table 3-4 で定義されているメッセージ形式をベースとしています。

Table 3-6 コントロールとステータスのために要求される IPv6-UDP PTP フレームのフィールド

対応するフィールド	オクテット位置	対応する値	説明
MAC Frame Type	12, 13	0x86DD	IPv4 データグラム
IP version	14 (bits [7:4])	0x6	IP バージョンが IPv6
Layer 4 protocol	20 (*)	0x11	UDP
PTP Multicast address	38-53	FF0x:0:0:0:0:0:0:181 (Hex) FF02:0:0:0:0:0:0:6B (Hex)	PTP primary multicast address: FF0x: 0:0:0:0:0:0:181 (Hex) PTP Pdelay multicast address: FF02: 0:0:0:0:0:0:6B (Hex)
UDP destination port	56, 57	0x013F, 0x140	0x013F : PTP event message 0x0140 : PTP general message
PTP Control Field (IEEE version 1)	93 (*)	0x00, 0x01, 0x02, 0x03, 0x04	0x00 : SYNC 0x01 : Delay_Req 0x02 : Follow_Up 0x03 : Delay_Resp 0x04 : 管理(バージョン 1)
PTP Message Type Field (IEEE version 2)	74 (*)(nibble)	0x0, 0x1, 0x2, 0x3, 0x8, 0x9, 0xA, 0xB, 0xC, 0xD	0x0 : SYNC 0x1 : Delay_Req 0x2 : Pdelay_Req 0x3 : Pdelay_Resp 0x8 : Follow_Up 0x9 : Delay_Resp 0xA : Pdelay_Resp_Follow_Up 0xB : Announce 0xC : Signaling 0xD : Management
PTP version field	75 (nibble)	0x1, 0x2	0x1 : PTP version 1 をサポート 0x2 : PTP version 2 をサポート

(*) PTP パケットの拡張ヘッダは定義されていません。

Ethernet 上の PTP フレーム

Table 3-7 は、Ethernet 上で送信される PTP パケットのスナップショットをコントロールするために照合されるフィールドの詳細を示しています。タグ・フレームのオクテット位置のオフセットは 4 です。

Table 3-7 コントロールとステータスのために要求される Ethernet PTP フレームのフィールド

対応するフィールド	オクテット位置	対応する値	説明
MAC Destination Multicast Address (*a)	0-5	01-1B-19-00-00-00 01-80-C2-00-00-0E	すべての PTP メッセージは以下のマルチキャストアドレス(*b)のどれでも使用できます。 01-1B-19-00-00-00 01-80-C2-00-00-0E (*c)
MAC Frame Type	12, 13	0x86F7	PTP Ethernet フレーム
PTP Control Field (IEEE version 1)	45	0x00, 0x01, 0x02, 0x03, 0x04	0x00: SYNC 0x01: Delay_Req 0x02: Follow_Up 0x03: Delay_Resp 0x04: Management
PTP Message Type Field (IEEE version 2)	14 (nibble)	0x0, 0x1, 0x2, 0x3, 0x8, 0x9, 0xA, 0xB, 0xC, 0xD	0x0: SYNC 0x1: Delay_Req 0x2: Pdelay_Req 0x3: Pdelay_Resp 0x8: Follow_Up 0x9: Delay_Resp 0xA: Pdelay_Resp_Follow_Up 0xB: Announce 0xC: Signaling 0xD: Management
PTP version field	15 (nibble)	0x1, 0x2	0x1: PTP version 1 をサポート 0x2: PTP version 2 をサポート

*a: Timestamp Control レジスタのコントロールビット 18 (TSENMF: Enable MAC address for PTP frame filtering)がセットされている場合、MAC アドレス 0～31 でプログラミングされている宛先アドレス(DA)のアドレス・マッチが使用されます。

*b: IEEE 標準 1588-2008, Annex F

*c: GMAC-UNIV は Peer delay マルチキャストアドレス(01-80-C2-00-00- 0E)が指定されている PTP version メッセージを有効な PTP メッセージと見なしません。

4. レジスタ

Ethernet-MAC のレジスタ機能について説明します。

レジスタマップ

Ethernet-MAC のレジスター一覧を Table 4-1 に示します。

Table 4-1 Ethernet-MAC レジスター一覧

アドレス	レジスタ番号	レジスタ略称	レジスタ名	参照先
0x0000	GMAC Reg. 0	MCR	MAC Configuration Register	4.1
0x0004	GMAC Reg. 1	MFFR	MAC Frame Filter Register	4.2
0x0008	GMAC Reg. 2	MHTRH	MAC Hash Table Register (High)	4.3
0x000C	GMAC Reg. 3	MHTRL	MAC Hash Table Register (Low)	4.3
0x0010	GMAC Reg. 4	GAR	GMII Address Register	4.4
0x0014	GMAC Reg. 5	GDR	GMII Data Register	4.5
0x0018	GMAC Reg. 6	FCR	Flow Control Register	4.6
0x001C	GMAC Reg. 7	VTR	VLAN Tag Register	4.7
0x0020 - 0x0024	-	-	予約	-
0x0028	GMAC Reg. 10	RWFFR	Remote Wake-up Frame Filter Register	4.8
0x002C	GMAC Reg. 11	PMTR	PMT Register	4.9
0x0030	GMAC Reg. 12	LPICSR	LPI Control and Status Register	4.10
0x0034	GMAC Reg. 13	LPITCR	LPI Timers Control Register	4.11
0x0038	GMAC Reg. 14	ISR	Interrupt Status Register	4.12
0x003C	GMAC Reg. 15	IMR	Interrupt Mask Register	4.13
0x0040	GMAC Reg. 16	MAR0H	MAC Address0 Register (High)	4.14
0x0044	GMAC Reg. 17	MAR0L	MAC Address0 Register (Low)	4.15
0x0048	GMAC Reg. 18	MAR1H	MAC Address1 Register (High)	4.16
0x004C	GMAC Reg. 19	MAR1L	MAC Address1 Register (Low)	4.17
0x0050	GMAC Reg. 20	MAR2H	MAC Address2 Register (High)	4.16
0x0054	GMAC Reg. 21	MAR2L	MAC Address2 Register (Low)	4.17
0x0058	GMAC Reg. 22	MAR3H	MAC Address3 Register (High)	4.16
0x005C	GMAC Reg. 23	MAR3L	MAC Address3 Register (Low)	4.17
0x0060	GMAC Reg. 24	MAR4H	MAC Address4 Register (High)	4.16
0x0064	GMAC Reg. 25	MAR4L	MAC Address4 Register (Low)	4.17
0x0068	GMAC Reg. 26	MAR5H	MAC Address5 Register (High)	4.16
0x006C	GMAC Reg. 27	MAR5L	MAC Address5 Register (Low)	4.17
0x0070	GMAC Reg. 28	MAR6H	MAC Address6 Register (High)	4.16
0x0074	GMAC Reg. 29	MAR6L	MAC Address6 Register (Low)	4.17
0x0078	GMAC Reg. 30	MAR7H	MAC Address7 Register (High)	4.16
0x007C	GMAC Reg. 31	MAR7L	MAC Address7 Register (Low)	4.17
0x0080	GMAC Reg. 32	MAR8H	MAC Address8 Register (High)	4.16
0x0084	GMAC Reg. 33	MAR8L	MAC Address8 Register (Low)	4.17
0x0088	GMAC Reg. 34	MAR9H	MAC Address9 Register (High)	4.16
0x008C	GMAC Reg. 35	MAR9L	MAC Address9 Register (Low)	4.17
0x0090	GMAC Reg. 36	MAR10H	MAC Address10 Register (High)	4.16
0x0094	GMAC Reg. 37	MAR10L	MAC Address10 Register (Low)	4.17
0x0098	GMAC Reg. 38	MAR11H	MAC Address11 Register (High)	4.16

アドレス	レジスタ番号	レジスタ略称	レジスタ名	参照先
0x009C	GMAC Reg. 39	MAR11L	MAC Address11 Register (Low)	4.17
0x00A0	GMAC Reg. 40	MAR12H	MAC Address12 Register (High)	4.16
0x00A4	GMAC Reg. 41	MAR12L	MAC Address12 Register (Low)	4.17
0x00A8	GMAC Reg. 42	MAR13H	MAC Address13 Register (High)	4.16
0x00AC	GMAC Reg. 43	MAR13L	MAC Address13 Register (Low)	4.17
0x00B0	GMAC Reg. 44	MAR14H	MAC Address14 Register (High)	4.16
0x00B4	GMAC Reg. 45	MAR14L	MAC Address14 Register (Low)	4.17
0x00B8	GMAC Reg. 46	MAR15H	MAC Address15 Register (High)	4.16
0x00BC	GMAC Reg. 47	MAR15L	MAC Address15 Register (Low)	4.17
0x00C0 - 0x00D0	-	-	予約	-
0x00D8	GMAC Reg. 54	RGSR	RGMII Status Register	4.18
0x00DC - 0x00FC	-	-	予約	-
0x0100	GMAC Reg. 64	mmc_cntl	MMC Control Register	4.49
0x0104	GMAC Reg. 65	mmc_intr_rx	MMC Receive Interrupt Register	4.50
0x0108	GMAC Reg. 66	mmc_intr_tx	MMC Transmit Interrupt Register	4.51
0x010C	GMAC Reg. 67	mmc_intr_mask_rx	MMC Receive Interrupt Mask Register	4.52
0x0110	GMAC Reg. 68	mmc_intr_mask_tx	MMC Transmit Interrupt Mask Register	4.53
0x0114	GMAC Reg. 69	txoctetcount_gb	MMC Counters	4.48
0x0118	GMAC Reg. 70	txframecount_gb		
0x011C	GMAC Reg. 71	txbroadcastframes_g		
0x0120	GMAC Reg. 72	txmulticastframes_g		
0x0124	GMAC Reg. 73	tx64octets_gb		
0x0128	GMAC Reg. 74	tx65to127octets_gb		
0x012C	GMAC Reg. 75	tx128to255octets_gb		
0x0130	GMAC Reg. 76	tx256to511octets_gb		
0x0134	GMAC Reg. 77	tx512to1023octets_gb		
0x0138	GMAC Reg. 78	tx1024tomaxoctets_gb		
0x013C	GMAC Reg. 79	txunicastframes_gb		
0x0140	GMAC Reg. 80	txmulticastframes_gb		
0x0144	GMAC Reg. 81	txbroadcastframes_gb		
0x0148	GMAC Reg. 82	txunderflowerror		
0x014C	GMAC Reg. 83	txsinglecol_g		
0x0150	GMAC Reg. 84	txmulticol_g		
0x0154	GMAC Reg. 85	txdeferred		
0x0158	GMAC Reg. 86	txlatecol		
0x015C	GMAC Reg. 87	txexesscol		
0x0160	GMAC Reg. 88	txcarriererrror		
0x0164	GMAC Reg. 89	txoctetcount_g		
0x0168	GMAC Reg. 90	txframecount_g		
0x016C	GMAC Reg. 91	txexecessdef		
0x0170	GMAC Reg. 92	txpauseframes		
0x0174	GMAC Reg. 93	txvlanframes_g		
0x0178, 0x017C	-	-	予約	-

アドレス	レジスタ番号	レジスタ略称	レジスタ名	参照先
0x0180	GMAC Reg. 96	rxframecount_gb	MMC Counters	4.48
0x0184	GMAC Reg. 97	rxoctetcount_gb		
0x0188	GMAC Reg. 98	rxoctetcount_g		
0x018C	GMAC Reg. 99	rxbroadcastframes_g		
0x0190	GMAC Reg. 100	rxmulticastframes_g		
0x0194	GMAC Reg. 101	rxcrcerror		
0x0198	GMAC Reg. 102	rxallignmenterror		
0x019C	GMAC Reg. 103	rxrunterror		
0x01A0	GMAC Reg. 104	rxjabbererror		
0x01A4	GMAC Reg. 105	rxundersize_g		
0x01A8	GMAC Reg. 106	rxoversize_g		
0x01AC	GMAC Reg. 107	rx64octets_gb		
0x01B0	GMAC Reg. 108	rx65to127octets_gb		
0x01B4	GMAC Reg. 109	rx128to255octets_gb		
0x01B8	GMAC Reg. 110	rx256to511octets_gb		
0x01BC	GMAC Reg. 111	rx512to1023octets_gb		
0x01C0	GMAC Reg. 112	rx1024tomaxoctets_gb		
0x01C4	GMAC Reg. 113	rxunicastframes_g		
0x01C8	GMAC Reg. 114	rxlengtherror		
0x01CC	GMAC Reg. 115	rxoutofrangetype		
0x01D0	GMAC Reg. 116	rxpauseframes		
0x01D4	GMAC Reg. 117	rxfifooverflow		
0x01D8	GMAC Reg. 118	rxvlanframes_gb		
0x01DC	GMAC Reg. 119	rxwatchdogerror		
0x01E0 - 0x01FC	-	-	予約	-
0x0200	GMAC Reg. 128	mmc_ipc_intr_mask_rx	MMC Receive Checksum Offload Interrupt Mask Register	4.54
0x0204	-	-	予約	-
0x0208	GMAC Reg. 130	mmc_ipc_intr_rx	MMC Receive Checksum Offload Interrupt Register	4.55
0x020C	-	-	予約	-
0x0210	GMAC Reg. 132	rxipv4_gd_frms	MMC Counters	4.48
0x0214	GMAC Reg. 133	rxipv4_hdrerr_frms		
0x0218	GMAC Reg. 134	rxipv4_nopay_frms		
0x021C	GMAC Reg. 135	rxipv4_frag_frms		
0x0220	GMAC Reg. 136	rxipv4_udsbl_frms		
0x0224	GMAC Reg. 137	rxipv6_gd_frms		
0x0228	GMAC Reg. 138	rxipv6_hdrerr_frms		
0x022C	GMAC Reg. 139	rxipv6_nopay_frms		
0x0230	GMAC Reg. 140	rxudp_gd_frms		
0x0234	GMAC Reg. 141	rxudp_err_frms		
0x0238	GMAC Reg. 142	rxtcp_gd_frms		
0x023C	GMAC Reg. 143	rxtcp_err_frms		
0x0240	GMAC Reg. 144	rxicmp_gd_frms		
0x0244	GMAC Reg. 145	rxicmp_err_frms		

アドレス	レジスタ番号	レジスタ略称	レジスタ名	参照先
0x0248, 0x024C	-	-	予約	-
0x0250	GMAC Reg. 148	rxipv4_gd_octets	MMC Counters	4.48
0x0254	GMAC Reg. 149	rxipv4_hdrerr_octets		
0x0258	GMAC Reg. 150	rxipv4_nopay_octets		
0x025C	GMAC Reg. 151	rxipv4_frag_octets		
0x0260	GMAC Reg. 152	rxipv4_udsbl_octets		
0x0264	GMAC Reg. 153	rxipv6_gd_octets		
0x0268	GMAC Reg. 154	rxipv6_hdrerr_octets		
0x026C	GMAC Reg. 155	rxipv6_nopay_octets		
0x0270	GMAC Reg. 156	rxudp_gd_octets		
0x0274	GMAC Reg. 157	rxudp_err_octets		
0x0278	GMAC Reg. 158	rxtcp_gd_octets		
0x027C	GMAC Reg. 159	rxtcp_err_octets		
0x0280	GMAC Reg. 160	rxicmp_gd_octets		
0x0284	GMAC Reg. 161	rxicmp_err_octets		
0x0288, 0x06FC	-	-	予約	-
0x0700	GMAC Reg. 448	TSCR	Time Stamp Control Register	4.19
0x0704	GMAC Reg. 449	SSIR	Sub-Second Increment Register	4.20
0x0708	GMAC Reg. 450	STSR	System Time - Seconds Register	4.21
0x070C	GMAC Reg. 451	STNR	System Time - Nanoseconds Register	4.22
0x0710	GMAC Reg. 452	STSUR	System Time - Seconds Update Register	4.23
0x0714	GMAC Reg. 453	STSNUR	System Time - Nanoseconds Update Register	4.24
0x0718	GMAC Reg. 454	TSAR	Time Stamp Addend Register	4.25
0x071C	GMAC Reg. 455	TTSR	Target Time Seconds Register	4.26
0x0720	GMAC Reg. 456	TTNR	Target Time Nanoseconds Register	4.27
0x0724	GMAC Reg. 457	STHWSR	System Time - High Word Seconds Register	4.28
0x0728	GMAC Reg. 458	TSR	Time Stamp Status Register	4.29
0x072C	GMAC Reg. 459	PPSCR	PPC Control Register	4.30
0x0730	GMAC Reg. 460	ATNR	Auxiliary Time Stamp-Nanosecond Register	4.31
0x0734	GMAC Reg. 461	ATSR	Auxiliary Time Stamp-Seconds Register	4.32
0x0738 - 0x07FC	-	-	予約	-
0x0800	GMAC Reg. 512	MAR16H	MAC Address16 Register (High)	4.16
0x0804	GMAC Reg. 513	MAR16L	MAC Address16 Register (Low)	4.17
0x0808	GMAC Reg. 514	MAR17H	MAC Address17 Register (High)	4.16
0x080C	GMAC Reg. 515	MAR17L	MAC Address17 Register (Low)	4.17
0x0810	GMAC Reg. 516	MAR18H	MAC Address18 Register (High)	4.16
0x0814	GMAC Reg. 517	MAR18L	MAC Address18 Register (Low)	4.17
0x0818	GMAC Reg. 518	MAR19H	MAC Address19 Register (High)	4.16
0x081C	GMAC Reg. 519	MAR19L	MAC Address19 Register (Low)	4.17
0x0820	GMAC Reg. 520	MAR20H	MAC Address20 Register (High)	4.16
0x0824	GMAC Reg. 521	MAR20L	MAC Address20 Register (Low)	4.17
0x0828	GMAC Reg. 522	MAR21H	MAC Address21 Register (High)	4.16
0x082C	GMAC Reg. 523	MAR21L	MAC Address21 Register (Low)	4.17

アドレス	レジスタ番号	レジスタ略称	レジスタ名	参照先
0x0830	GMAC Reg. 524	MAR22H	MAC Address22 Register (High)	4.16
0x0834	GMAC Reg. 525	MAR22L	MAC Address22 Register (Low)	4.17
0x0838	GMAC Reg. 526	MAR23H	MAC Address23 Register (High)	4.16
0x083C	GMAC Reg. 527	MAR23L	MAC Address23 Register (Low)	4.17
0x0840	GMAC Reg. 528	MAR24H	MAC Address24 Register (High)	4.16
0x0844	GMAC Reg. 529	MAR24L	MAC Address24 Register (Low)	4.17
0x0848	GMAC Reg. 530	MAR25H	MAC Address25 Register (High)	4.16
0x084C	GMAC Reg. 531	MAR25L	MAC Address25 Register (Low)	4.17
0x0850	GMAC Reg. 532	MAR26H	MAC Address26 Register (High)	4.16
0x0854	GMAC Reg. 533	MAR26L	MAC Address26 Register (Low)	4.17
0x0858	GMAC Reg. 534	MAR27H	MAC Address27 Register (High)	4.16
0x085C	GMAC Reg. 535	MAR27L	MAC Address27 Register (Low)	4.17
0x0860	GMAC Reg. 536	MAR28H	MAC Address28 Register (High)	4.16
0x0864	GMAC Reg. 537	MAR28L	MAC Address28 Register (Low)	4.17
0x0868	GMAC Reg. 538	MAR29H	MAC Address29 Register (High)	4.16
0x086C	GMAC Reg. 539	MAR29L	MAC Address29 Register (Low)	4.17
0x0870	GMAC Reg. 540	MAR30H	MAC Address30 Register (High)	4.16
0x0874	GMAC Reg. 541	MAR30L	MAC Address30 Register (Low)	4.17
0x0878	GMAC Reg. 542	MAR31H	MAC Address31 Register (High)	4.16
0x087C	GMAC Reg. 543	MAR31L	MAC Address31 Register (Low)	4.17
0x0880 - 0x0FFC	-	-	予約	-
0x1000	DMA Reg. 0	BMR	BUS Mode Register	4.33
0x1004	DMA Reg. 1	TPDR	Transmit Poll Demand Register	4.34
0x1008	DMA Reg. 2	RPDR	Receive Poll Demand Register	4.35
0x100C	DMA Reg. 3	RDLAR	Receive Descriptor List Address Register	4.36
0x1010	DMA Reg. 4	TDLAR	Transmit Descriptor List Address Register	4.37
0x1014	DMA Reg. 5	SR	Status Register	4.38
0x1018	DMA Reg. 6	OMR	Operation Mode Register	4.39
0x101C	DMA Reg. 7	IER	Interrupt Enable Register	4.40
0x1020	DMA Reg. 8	MFBOCR	Missed Frame and Buffer Overflow Counter Register	4.41
0x1024	DMA Reg. 9	RIWTR	Receive Interrupt Watchdog Timer Register	4.42
0x1028	-	-	予約	-
0x102C	DMA Reg. 11	AHBSR	AHB Status Register	4.43
0x1030 - 0x1044	-	-	予約	-
0x1048	DMA Reg. 18	CHTDR	Current Host Transmit Descriptor Register	4.44
0x104C	DMA Reg. 19	CHRDR	Current Host Receive Descriptor Register	4.45
0x1050	DMA Reg. 20	CHTBAR	Current Host Transmit Buffer Address Register	4.46
0x1054	DMA Reg. 21	CHRBAR	Current Host Receive Buffer Address Register	4.47
0x1058 - 0xFFFF	-	-	予約	-

レジスタマップの見方(例)

Register Abbreviation (Register full name) Offset address xxxxh

bit	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
Field	A	B	C	D	E	F	G	H
属性	R/W	R/W	R	R	R	R	R	R/WSC
初期値	0	0	0	0	0	0	0	0

Register Abbreviation: レジスタ略称を記載します。
Register full name: レジスタ・フルネームを記載します。
Offset address: レジスタ・オフセットアドレスを記載します。
Bits: ビット位置を示します。
Field: ビット・フィールド名称を記載します。
属性: レジスタのリード/ライト属性を記載します。記載記号の意味は以下です。

- R: リード可能
- W: ライト可能
- R/W_SC: アプリケーションからのリード/ライト (Read and Write) ができます。GMAC/DMA によって 0 にクリア(Self Clear)されます。GMAC/DMA がこのフィールドをクリアする条件については、フィールドの説明を参照してください。
- R_SS_WC: アプリケーションからのリード(Read) ができます。特定の内部イベントで GMAC/DMA によって 1 にセット(Self Set)されます。アプリケーションからの 1 ライトにより、0 にクリア(Write Clear)できます。アプリケーションからの 0 ライトは何も行いません。GMAC/DMA がこのフィールドをセットする条件については、フィールドの説明を参照してください(例：割込みビット)。
- R/WS_SC: アプリケーションからのリード(Read)と、アプリケーションからの 1 ライト(Write Set)ができます。アプリケーションからの 0 ライトは無効です。GMAC/DMA によって 0 にクリア(Self Clear)されます。GMAC/DMA がこのフィールドをクリアする条件については、フィールドの説明を参照してください(例：リセット信号)。
- R/SS_SC_WC: アプリケーションからのリード(Read)ができます。特定の内部イベントで GMAC/DMA によって 1 にセット(Self Set)、または 0 にクリア(Self Clear)されます。アプリケーションからの 0 ライトにより、0 にクリア(Write Clear) できます。アプリケーションからの 1 ライトは何も行いません。GMAC/DMA がこのフィールドをセットまたはクリアする条件については、フィールドの説明を参照してください。
- R/WT: アプリケーションからのリード (Read) ができます。何らかのデータ値でライト動作が実行 (Write Trigger) されたとき、フィールドの説明に示すように、イベントがトリガされます(例:受信 Poll Demand レジスタ)。
- R_SS_RC: アプリケーションからのリード(Read)ができます。特定の内部イベントで GMAC/DMA によって 1 にセット(Self Set)されます。アプリケーションからのリードで自動的に 0 にクリア(Read Clear) されます。アプリケーションからの 0 ライトは、何も行いません。GMAC/DMA がこのフィールドをセットする条件については、フィールドの説明を参照してください(例：オーバフローカウンタ)
- R/WSU: アプリケーションによるリード/ライト (Read and Write) ができます。レジスタフィールドはイベントに基づいて自動的に更新(Self Update)されます。これは、例えば、PTP 構成におけるシステム時間に使用されます。
- Reserved: レジスタフィールドの値は変更できません。変更を行った場合、GMAC/DMA は、予期せぬ動作を行う可能性があります。
- 初期値: レジスタのリセット後の値を示します。

4.1 GMAC Register 0 (MCR)

MCR レジスタは、送受信のオペレーティングモードを設定します。

MCR (MAC Configuration Register) Address 0000h

bit	31	30	29	28	27	26	25	24
Field	予約						CST	TC
属性	R	R	R	R	R	R	R/W	R/W
初期値	-	-	-	-	-	-	0	0

bit	23	22	21	20	19	18	17	16
Field	WD	JD	BE	JE	IFG[2:0]			DCRS
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	PS	FES	DO	LM	DM	IPC	DR	LUD
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	ACS	BL[1:0]		DC	TE	RE	予約	
属性	R/W	R/W	R/W	R/W	R/W	R/W	R	R
初期値	0	0	0	0	0	0	-	-

[bit25] CST (CRC stripping for Type frames)

このビットを 1 にセットすると、フレームをアプリケーションに転送する前に、Ether タイプ(タイプフィールド 0x0600 より大きい)のすべてのフレームの最後の 4 バイト(FCS)が除去し、廃棄します。

[bit24] TC (Transmit Configuration in RGMII)

MII/RMII モード時は、Reserved ビットです。このビットにライトする場合は、常に 0 の書込みを行ってください。

[bit23] WD (Watchdog Disable)

このビットを 1 にセットすると、GMAC はレシーバの Watchdog Timer を無効にし、最大 16384 バイトのフレームを受信できるようになります。このビットを 0 にリセットすると、GMAC は受信フレームを 2048 バイトまで(JE ビットが High にセットされている場合は 10240 まで)のみ許可し、それ以降のすべてのバイトを切り捨てます。

[bit22] JD (Jabber Disable)

このビットを 1 にセットすると、GMAC はトランスミッタの Jabber Timer を無効にし、最大 16384 バイトのフレームを送信できるようになります。このビットを 0 にリセットすると、送信中にアプリケーションが 2048 バイト(JE ビットが High にセットされている場合は 10240 バイト)を超えるデータを送信した場合に GMAC はトランスミッタを停止します。

[bit21] BE (Frame Burst Enable)

MII/RMII モード時は、Reserved ビットです。このビットにライトする場合は、常に 0 の書込みを行ってください。

[bit20] JE (Jumbo Frame Enable)

このビットを 1 にセットすると、GMAC は最大 9018 バイト(VLAN tagged フレームの場合は 9022 バイト)を受信できます。

[bit19:17] IFG (Inter-Frame GAP)
これらのビットは、送信時のフレーム間の最小 IFG をコントロールします。
000 : 96 ビットタイム
001 : 88 ビットタイム
010 : 80 ビットタイム
011 : 72 ビットタイム
100 : 64 ビットタイム
101 : 56 ビットタイム
110 : 48 ビットタイム
111 : 40 ビットタイム
半二重モードでは、最小 IFG は 64 ビットタイム(IFG = 100)にのみ設定できます。64 ビットタイム未満の値は無視されます。

[bit16] DCRS(Disable Carrier Sense During Transaction)
このビットを 1 にセットすると、GMAC トランスミッタは、半二重モードでのフレーム送信中に、CRS 信号を無視します。その結果、そのような送信中には Loss of Carrier または No Carrier によるエラーは生成されません。このビットが 0 のとき、GMAC トランスミッタは、Carrier Sense によってエラーを生成して送信を中止します。

[bit15] PS (Port Select)
このビットの初期値は 0 です。GMAC の初期化時に必ず 1 を書き込んでください。PHY のインタフェースとして、MII または、RMII が選択されます。
このビットにライトする場合は、常に 1 の書込みを行ってください。

[bit14] FES(Speed)
RMII モードの場合の通信速度を指定します。
0 : 10 Mbps
1 : 100 Mbps

このビットは、RMII のときのみ有効です。

[bit13] DO (Disable Receive Own)
このビットを 1 にセットすると、半二重モードで TX_EN がアサートされたときに、フレームの受信を無効にします。このビットを 0 にリセットすると、GMAC は送信中に PHY から来るパケットをすべて受信します。GMAC が全二重モードで動作しているときは、このビットは無効です。

[bit12] LM (Loop-back Mode)
このビットを 1 にセットすると、GMAC はループバックモードで動作します。
ループバックが適切に機能するためには、受信クロック入力(RX_CLK/REF_CLK)が必要です。
*このビットがセットされているとき、PAUSE フレームは送信できません。

[bit11] DM (Duplex mode)
このビットを 1 にセットすると、GMAC は全二重モードで動作します。全二重モードでは受信と送信を同時に行うことができます。

[bit10] IPC (Checksum Offload)
このビットを 1 にセットすると、受信フレームに対する IPv4 チェックサムのチェック, ペイロードの TCP/UDP/ICMP ヘッダのチェックが有効になります。このビットを 0 にリセットすると、レシーバの COE 機能が禁止され、対応する PCE および IP HCE ステータスビットが常にクリアされます。

<注意事項>

- *IPv6 認証ヘッダを使用する Ethernet 環境下では、必ず IPC は 0 にリセットしてください。*

[bit9] DR (Disable Retry)
このビットを 1 にセットすると、GMAC は 1 回だけ送信を試行します。PHY インタフェースでコリジョンが発生したとき、GMAC は現在のフレーム送信を無視し、フレームアボートおよび過剰コリジョンの送信フレームステータスを通知します。このビットを 0 にリセットすると、GMAC は BL でセットした条件に従ってリトライします。このビットは半二重モードでのみ有効です。

[bit8] LUD(Link Up/Down in RGMII)
MII/RMII モード時は、Reserved ビットです。このビットにライトする場合は、常に 0 の書込みを行ってください。

[bit7] ACS (Automatic Pad/CRC Stripping)
このビットをセットすると、GMAC は、受信フレームの Length のフィールドの値が 1500 バイト以下である場合のみ、受信フレームの Pad/FCS フィールドを取り除きます。1501 バイト以上の場合は、Pad/FCS フィールドを取り除くことなく、アプリケーションに渡します。このビットをリセットすると、GMAC は、すべての受信フレームを変更なしで Host に渡します。

＜注意事項＞

- *ACS ビットを 1 として使用する際は、必ず Receive-Store-Forward モードで使用してください。*
  Receive-Store-Forward モードは、RSF (DMA Register 6 (OMR) bit25) を 1 にすることで指定できます。

[bit6,5] BL (Back-off Limit)
バックオフリミットは、スロットタイム遅延をランダムな整数 (r) で設定します(10/100 Mbps では 512 ビットタイム)。GMAC は、コリジョンの後のリトライ中に、送信試行が再スケジュールされるまで待機します。このビットは半二重モードでのみ有効です。

00 : k = min (n, 10)　(デフォルト)
01 : k = min (n, 8)
10 : k = min (n, 4)
11 : k = min (n, 1)

ここで、n = 送信リトライ回数。ランダムな整数 r は、0 ≦ r < 2^K の範囲から選択されます。

[bit4] DC (Deferral Check)
このビットをセットすると、GMAC で延期 (Deferral) チェック機能が許可されます。10/100 Mbps モードで送信ステートマシンが 24288 ビットタイムを超えて延期された場合、GMAC はフレームアボートステータスを発行し、送信フレームステータスに過剰延期エラービットをセットします。10/100 Mbps モードで Jumbo フレームモードがイネーブルされている場合、延期のしきい値は 155680 ビットタイムになります。延期は、トランスミッタが送信可能な状態で、PHY インタフェースのアクティブな CRS(キャリア検出)信号のために送信できないときに始まります。延期時間は累積ではありません。送信が 10000 ビットタイムの間延期されると、送信を行い、コリジョンが発生し、バックオフします。バックオフの完了後に再び延期が必要とされた場合、延期タイマは 0 にリセットされ、リスタートされます。このビットをリセットすると、延期チェック機能が禁止され、GMAC は CRS 信号が非アクティブになるまで延期します。このビットは半二重モードでのみ有効です。

[bit3] TE (Transmitter Enable)
このビットを 1 にセットすると、GMAC 送信ステートマシンの PHY インタフェースへの送信動作が許可されます。このビットを 0 にリセットすると、現在のフレームの送信完了後に GMAC の送信ステートマシンの送信動作が禁止され、以降のフレームを送信しません。

[bit2] RE (Receiver Enable)
このビットを 1 にセットすると、GMAC 受信ステートマシンの PHY インタフェースからのフレーム受信動作が許可されます。このビットを 0 にリセットすると、GMAC 受信ステートマシンの受信動作が禁止され、PHY インタフェースからのフレームは受信されません。

4.2 GMAC Register 1 (MFFR)

MFFR レジスタは、受信フレームのフィルタリング制御を行います。

このレジスタのコントロールの一部は、GMAC のアドレスチェックブロックに進み、そこでアドレスフィルタリングの最初のレベルを実行します。フィルタリングの 2 番目のレベルは、受信フレームに対して、ほかのコントロール(Pass Bad Frames, Pass Control Frames など)をベースとして実行されます。

MFFR (MAC Frame Filter Register) Address 00004h

bit	31	30	29	28	27	26	25	24
Field	RA	予約						
属性	R/W	R	R	R	R	R	R	R
初期値	0	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約					HPF	SAF	SAIF
属性	R	R	R	R	R	R/W	R/W	R/W
初期値	-	-	-	-	-	-	0	0

bit	7	6	5	4	3	2	1	0
Field	PCF[1: 0]		DB	PM	DAIF	HMC	HUC	PR
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit31] RA (Receive All)

このビットをセットすると、GMAC レシーバモジュールは、フレームがアドレスフィルタをパスしたかどうかに関わりなく、受信したすべてのフレームをアプリケーションに渡します。SA/DA フィルタリングの結果により、受信ステータスの対応するビットが更新されます(パスまたはフェイル)。このビットをリセットすると、レシーバモジュールは、SA/DA アドレスフィルタをパスしたフレームのみをアプリケーションに渡します。

[bit10] HPF (Hash or Perfect Filter)

このビットを 1 にセットすると、フレームはパーフェクトフィルタまたはハッシュフィルタ(HMC/HUC ビットによって設定されている場合) のいずれかとマッチする場合に、アドレスフィルタをパスします。このビットを 0 にリセットすると、フレームはハッシュフィルタ(HUC/HMC ビットによって設定されている場合)とマッチする場合にだけパスします。

[bit9] SAF (Source Address Filter)

GMAC は、受信フレームの SA フィールドと、イネーブルされている SA レジスタでプログラムされている値を比較します。このビットを 1 にセットすると、SA フィルタフェイル時、そのフレームを廃棄します。このビットを 0 にリセットすると、そのフレームをアプリケーションに転送します。

[bit8] SAIF (Source Address Inverse Filter)

このビットをセットすると、Address Check ブロックが SA アドレス比較のために、逆フィルタリングモードで動作します。SA が SA レジスタとマッチするフレームには、SA Address フィルタをフェイルしたことを示すマークが付けられます。
このビットをリセットすると、SA が SA レジスタとマッチしないフレームに、SA Address フィルタをフェイルしたことを示すマークが付けられます。

[bit7, 6] PCF (Pass Control Frames)

このビットは、以下のすべてのコントロールフレーム(ユニキャストおよびマルチキャスト PAUSE フレームを含む)の転送をコントロールします。

00: すべてのコントロールフレームをフィルタリングし、アプリケーションに到達しないようにします。

01: PAUSE を除くすべてのコントロールフレームを、Address フィルタをフェイルした場合でも、アプリケーションに転送します。

10: すべてのコントロールフレームを、Address フィルタをフェイルした場合でも、アプリケーションに転送します。

11: Address フィルタをパスしたコントロールフレームを転送します。

PAUSE コントロールフレームを処理するためには、以下の条件が真である必要があります。

条件 1: MAC が全二重モードで、GMAC Register 6 (Flow Control Register)の bit2 (RFE)を 1 にセットすることによってフローコントロールが許可されている。

条件 2: GMAC Register 6 (Flow Control Register)の bit3 (UP)がセットされているとき、受信フレームのディスティネーション・アドレス(DA)が特別のマルチキャストアドレスまたは MAC Address 0 と一致する。

条件 3: 受信フレームの Type フィールドが 0x8808 で、OPCODE フィールドが 0x0001 である。

<注意事項>

- *PCF フィールドは、条件 1 が真であるとき、つまり、MAC が全二重モードで動作するようにプログラムされており、RFE ビットがイネーブルされている場合にのみ 01 にセットします。*
  そうでない場合は、PAUSE フレームのフィルタリングに矛盾が生じることがあります。条件 1 が偽である場合、PAUSE フレームは汎用コントロールフレームと見なされます。そのため、全二重モードでフローコントロールが有効でない場合は、PAUSE コントロールフレームを含むすべてのコントロールフレームをパスさせるために、PCF フィールドを(アプリケーションの要求に応じて)10 または 11 にセットする必要があります。

[bit5] DB (Disable Broadcast Frames)

このビットを 1 にセットすると、すべての受信したブロードキャストフレームをフィルタします。
このビットを 0 にリセットすると、すべての受信したブロードキャストフレームをパスさせます。

[bit4] PM (Pass All Multicast)

このビットを 1 にセットすると、マルチキャスト・ディスティネーション・アドレスを持つすべての受信フレーム(ディスティネーション・アドレス・フィールドの最初のビットが 1)をパスさせます。
このビットを 0 にリセットすると、マルチキャストフレームのフィルタリングは HMC ビットに依存します。

[bit3] DAIF (DA Inverse Filtering)

このビットを 1 にセットすると、Address Check ブロックはユニキャストフレームとマルチキャストフレームの両方に対して、逆フィルタリングモードで DA アドレス比較を行います。
このビットを 0 にリセットすると、通常のフレーム・フィルタリングが実行されます。
*フレームがコントロールフレームであるとき、この機能は無効です。

[bit2] HMC (Hash Multicast)

このビットを 1 にセットすると、受信したマルチキャストフレームのディスティネーション・アドレス・フィルタリングをハッシュテーブルに従って実行します。このビットを 0 にリセットすると、マルチキャストフレームに対する完全なディスティネーション・アドレス・フィルタリングを実行します。つまり、DA フィールドと DA レジスタにプログラムされている値を比較します。

[bit1] HUC (Hash Unicast)

このビットを 1 にセットすると、ユニキャストフレームのディスティネーション・アドレス・フィルタリングをハッシュテーブルに従って実行します。
このビットを 0 にリセットすると、ユニキャストフレームに対する完全なディスティネーション・アドレス・フィルタリングを実行します。つまり、DA フィールドと、DA レジスタにプログラムされている値を比較します。

[bit0] PR (Promiscuous Mode)

このビットをセットすると、Address Filter モジュールは、ディスティネーション・アドレスおよびソース・アドレスに関わりなくすべての受信フレームをパスさせます。PR がセットされているとき、受信ステータスワードの SA/DA フィルタフェイルステータスは、常にクリアされます。

Table 4-2, Table 4-3 に、受信フレームタイプごとの SA/DA フィルタリング内容をまとめて示します。

Table 4-2 ディスティネーション・アドレス・フィルタリングテーブル

フレームタイプ	PR	HPF	HCU	DAIF	HMC	PM	DB	DA フィルタ動作
ブロードキャスト	1	X	X	X	X	X	X	パス
	0	X	X	X	X	X	0	パス
	0	X	X	X	X	X	1	フェイル
ユニキャスト	1	X	X	X	X	X	X	すべてのフレームをパス
	0	X	0	0	X	X	X	Perfect/Group フィルタマッチ時パス
	0	X	0	1	X	X	X	Perfect/Group フィルタマッチ時フェイル
	0	0	1	0	X	X	X	Hash フィルタマッチ時パス
	0	0	1	1	X	X	X	Hash フィルタマッチ時フェイル
	0	1	1	0	X	X	X	Hash または Perfect/Group フィルタマッチ時パス
	0	1	1	1	X	X	X	Hash または Perfect/Group フィルタマッチ時フェイル
マルチキャスト	1	X	X	X	X	X	X	すべてのフレームをパス
	X	X	X	X	X	1	X	すべてのフレームをパス
	0	X	X	0	0	0	X	Perfect/Group フィルタマッチ時パス PCF = 0x 時、PAUSE コントロールフレーム廃棄
	0	0	X	0	1	0	X	Hash フィルタマッチ時パス PCF = 0x 時、PAUSE コントロールフレーム廃棄
	0	1	X	0	1	0	X	Hash または Perfect/Group フィルタマッチ時パス PCF = 0x 時、PAUSE コントロールフレーム廃棄
	0	X	X	1	0	0	X	Perfect/Group フィルタマッチ時フェイル PCF = 0x 時、PAUSE コントロールフレーム廃棄
	0	0	X	1	1	0	X	Hash フィルタマッチ時フェイル PCF = 0x 時、PAUSE コントロールフレーム廃棄
	0	1	X	1	1	0	X	Hash または Perfect/Group フィルタマッチ時フェイル PCF = 0x 時、PAUSE コントロールフレーム廃棄

Table 4-3 ソース・アドレス・フィルタリング・テーブル

フレームタイプ	PR	SAIF	SAF	SA フィルタ動作
ユニキャスト	1	X	X	すべてのフレームをパス。
	0	0	0	Perfect/Group フィルタマッチ時にパスステータスを返却。 フレームを廃棄しない
	0	1	0	Perfect/Group フィルタマッチ時にフェイルステータスを返却。 フレームを廃棄しない
	0	0	1	Perfect/Group フィルタマッチ時パス。 フェイルしたフレームを廃棄
	0	1	1	Perfect/Group フィルタマッチ時フェイル。 フェイルしたフレームを廃棄

4.3 GMAC Register 2, 3 (MHTRH, MHTRL)

MHTRH および MHTRL レジスタは、グループアドレス・フィルタリングを設定します。

HTH フィールドにはハッシュテーブルの上位 32 ビットを設定します。
HTL フィールドにはハッシュテーブルの下位 32 ビットを設定します。

MHTRH (MAC Hash Table Register (High))	Address 0008h
bit	31 ～ 0
Field	HTH[31:0]
属性	R/W
初期値	0

MHTRL (MAC Hash Table Register (Low))	Address 000Ch
bit	31 ～ 0
Field	HTL[31:0]
属性	R/W
初期値	0

[bit31:0] HTH, HTL

64 ビットハッシュテーブルは、グループアドレスのフィルタリングのために使用します。ハッシュフィルタリングでは、受信フレームのディスティネーション・アドレスの内容が CRC ロジックに渡され、CRC 演算結果の上位 6 ビットのビット反転値をハッシュテーブル内容のインデックスとして使用します。インデックスの最上位ビットは使用するレジスタ(HTH/HTL)を決定します。ほかの 5 ビットは各レジスタ内のビット位置を決定します。00000 は選択したレジスタの bit0 を指定し、11111 は選択したレジスタの bit31 を指定します。
HTH/HTL レジスタの対応するビット値が 1 の場合は、受信フレームは受け入れられます。そうでない場合は、フレームは拒否されます。GMAC Register1 の PM (Pass All Multicast)ビットがセットされている場合、マルチキャストハッシュ値に関係なく、すべてのマルチキャストフレームが受け入れられます。

例えば、受信フレームの DA が 0x1F52419CB6AF(0x1F が受信された最初のバイト)である場合、CRC により計算された 6 ビット値は 0x2C となり、HTH レジスタビット[12]に設定された値が、フィルタリングのために使用されます。受信フレームの DA が 0xA00A98000045 である場合、CRC により計算された 6 ビット値は 0x07 となり、HTL レジスタビット[7] に設定された値が、フィルタリングのために使用されます。

上記 CRC は以下を使用します。

CRC_32 : $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$
POLYNOMIAL : 0x04C11DB7
INITIAL_REMAINDER : 0xFFFFFFFF
FINAL_XOR_VALUE : CRC の 1 の補数

＜注意事項＞

- *このレジスタへのライトアクセスは、32 ビット幅で行ってください。一度値を書き込んだ後、異なる値に書換えを行う場合は、PHY インタフェースクロックで、4 サイクルクロック時間以上経過してから、書換え処理を行う必要があります。*

4.4 GMAC Register 4 (GAR)

GMII Address レジスタは、マネジメントインタフェースを通じて外部 PHY へのマネジメントサイクルをコントロールします。

GAR (GMII/MII Address Register) Address 0010h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	PA[4:0]					GR[4:2]		
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	GR[1:0]		CR[3:0]				GW	GB
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/WS_SC
初期値	0	0	0	0	0	0	0	0

[bit15:11] PA (Physical Layer Address)

このフィールドは、接続する 32 個の PHY デバイスの PHY アドレスを指定します。

[bit10:6] GR (GMII Register)

このビットは、選択した PHY デバイス内のアクセスするレジスタアドレスを指定します。

[bit5:2] CR (Application Clock Range)

CR により、Ethernet-MAC に供給されている SYS_CLK 周波数に従って MDC クロックの周波数を決定します。bit5 = 0 のときは、下記の SYS_CLK 周波数レンジの値を適用すれば、MDC クロックの周波数レンジが約 1.25MHz～2.5MHz の範囲の値に設定されます。

CR	SYS_CLK	MDC クロック
0000	60 MHz -100 MHz	SYS_CLK/42
0001	100 MHz -150 MHz	SYS_CLK/62
0010	20 MHz – 35 MHz	SYS_CLK/16
0011	35 MHz – 60 MHz	SYS_CLK/26
0100	150 MHz – 250 MHz	SYS_CLK/102
0101	250 MHz – 300 MHz	SYS_CLK/122

上記外：Reserved

[bit1] GW (GMII/MII Write)

このビットを 1 にセットすると、PHY に対し、GMAC Register5 (GMII Data)の値のライト動作を行います。このビットを 0 にリセットすると、PHY に対し、リード動作を行い、リード値が GMAC Register5 (GMII Data)に格納されます。

[bit0] GB (GMII/MII Busy)

アプリケーションによるこのビットへの 1 書込みにより、PHY レジスタへのアクセスが開始され、このビットが 1 の間は、PHY へのリード動作, ライト動作が進行中であることを示します。PHY へのアクセスが完了すると、このビットは自動的に 0 にクリアされます。

GMAC Register 4 および GMAC Register5 への書込みを行う前に、このビットから 0 が読み出せることを確認する必要があります。PHY へのアクセス進行中に、GMAC Register4, GMAC Register5 を書き換えることはできません。PHY へのリード動作が完了するまで、GMAC Register5 (GMII Data)の値は無効です。

4.5 GMAC Register 5 (GDR)

GMII Data レジスタは、PHY レジスタにライトデータを設定します。また、PHY レジスタからのリードデータが格納されます。

GDR (GMII/MII Data Register) Address 0014h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	GD[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	GD[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit15:0] GD (GMII/MII Data Register)

マネジメントライト動作前に、PHY に書き込む 16 ビットデータ値を設定します。
マネジメントリード動作後に、PHY から読み出された 16 ビットデータ値を格納します。

4.6 GMAC Register 6 (FCR)

Flow Control レジスタは、GMAC のフローコントロールモジュールによる Control (Pause Command)フレームの生成および受信をコントロールします。

FCR (Flow Control Register) Address 0018h

bit	31	30	29	28	27	26	25	24
Field	PT[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	PT[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	DZPQ	予約	PLT[1:0]		UP	RFE	TFE	FCB/BPA
属性	R/W	R	R/W	R/W	R/W	R/W	R/W	R/WS_SC/R/W
初期値	0	-	0	0	0	0	0	0

Flow Control レジスタは、GMAC のフローコントロールモジュールによる Control (Pause Command)フレームの生成および受信をコントロールします。Busy ビットを 1 にセットしレジスタへのライト動作することにより、Flow Control ブロックによる Pause Control フレームの生成がトリガされます。コントロールフレームのフィールドは、802.3x 仕様で指定されているとおり選択され、このレジスタからの Pause Time 値は、コントロールフレームの Pause Time フィールドで使用されます。Busy ビットは、コントロールフレームがケーブル上に転送されるまでセットされたままです。Host は、レジスタへのライト動作の前に、Busy ビットがクリアされていることを確認しなければなりません。

[bit31:16] PT (Pause Time)

このフィールドは、送信コントロールフレームの Pause Time フィールドで使用する値を保持します。このレジスタへ、一度値を書き込んだ後、異なる値に書換えを行う場合は、PHY インタフェースクロックで、4 サイクルクロック時間以上経過してから、書換え処理を行う必要があります。

[bit7] DZPQ (Disable Zero-Quanta Pause)

このビットは、Reserved ビットです。

[bit5,4] PLT (Pause Low Threshold)

このビットは、Reserved ビットです。

[bit3] UP (Unicast Pause Frame detect)

このビットを 1 にセットすると、GMAC は、MAC Address0 High Register および MAC Address0 Low Register で指定されたステーション内のユニキャストアドレスをもつ Pause フレームを検出し、また一意なマルチキャストアドレスをもつ Pause フレームを検出します。
このビットをリセットすると、GMAC は、802.3x 標準で指定された一意なマルチキャストアドレスをもつ Pause フレームを検出します。

[bit2] RFE (Receive Flow Control Enable)

このビットをセットすると、GMAC は、受信した Pause フレームをデコードし、指定された期間、そのトランスミッタをディセーブルします。
このビットをリセットすると、Pause フレームのデコード機能が禁止されます。

[bit1] TFE (Transmit Flow Control Enable)

全二重モードでは、このビットを 1 にセットすると、GMAC はフローコントロール動作を許可し、Pause フレームを送信します。このビットをリセットすると、GMAC のフローコントロール動作が禁止され、GMAC は、Pause フレームを送信しなくなります。
半二重モードでは、このビットをセットすると、GMAC はバックプレッシャ動作を許可します。このビットをリセットするとバックプレッシャ機能が禁止されます。

[bit0] FCB/BPA (Flow Control Busy/Backpressure Activate)

このビットは、全二重モードで Pause Control フレームを開始し、TFE ビットがセットされている場合に半二重モードでバックプレッシャ機能を起動します。全二重モードでは、Flow Control レジスタへの書込みの前にこのビットが 0 になっている必要があります。Pause Control フレームを開始するためには、アプリケーションはこのビットを 1 にセットしなければなりません。Control Frame フレームの送信中、このビットはセットされたままになり、フレーム送信が進行中であることを示します。Pause Control フレーム送信の完了後、GMAC はこのビットを 0 にリセットします。このビットがクリアされるまで Flow Control レジスタへの書込みはできません。
半二重モードでは、このビットをセットすると(TFE もセット)、バックプレッシャが GMAC によってアサートされます。バックプレッシャ中に GMAC が新しいフレームを受信したとき、トランスミッタは JAM パターンの送信を開始し、その結果コリジョンが発生します。GMAC が半二重モードに設定されている場合、BPA は自動的にディセーブルされます。

4.7 GMAC Register 7 (VTR)

VLAN Tag レジスタは、VLAN フレームを識別する値を設定します。

VTR (VLAN TAG Register) Address 001Ch

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							ETV
属性	R	R	R	R	R	R	R	R/W
初期値	-	-	-	-	-	-	-	0

bit	15	14	13	12	11	10	9	8
Field	VL[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	VL[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

VLAN Tag レジスタは、VLAN フレームを識別するための IEEE 802.1Q VLAN タグを含んでいます。GMAC は受信フレームの 13～14 番目のバイト(Length/Type)を 0x8100 と比較し、その次の 2 バイトを VLAN タグと比較します。マッチした場合、受信フレームステータスの受信 VLAN ビットをセットします。フレームの有効な長さは 1518 バイトから 1522 バイトに増えます。

[bit16] ETV (Enable 12-Bit VLAN Tag Comparison)

このビットをセットすると、比較およびフィルタリングのために、完全な 16 ビット VLAN タグではなく、12 ビット VLAN 識別子が使用されます。VLAN タグの bit[11:0]が、受信した VLAN タグ付きフレーム内の対応するフィールドと比較されます。このビットをリセットすると、受信した VLAN フレームの 15～16 番目のバイトの 16 ビットすべてが比較に使用されます。

[bit15:0] VL (VLAN Tag Identifier)

このフィールドは、VLAN フレームを識別するための 802.1Q VLAN タグを含んでおり、VLAN フレームで受信されるフレームの 15～16 番目のバイトと比較されます。bit[15:13]は User Priority, bit [12]は Canonical Format Indicator (CFI)、bit[11:0]は VLAN タグの VLAN Identifier (VID)フィールドです。

ETV ビットをセットすると、比較のために VID(bit[11:0])のみが使用されます。VL (ETV がセットされている場合は VL[11：0])がすべてゼロである場合、GMAC は、VLAN タグ比較で 15～16 番目のバイトをチェックせず、Type フィールド値が 0x8100 のすべてのフレームを VLAN フレームとして宣言します。

＜注意事項＞

- ***このレジスタへのライトアクセスは、32 ビット幅で行ってください。一度値を書き込んだ後、異なる値に書換えを行う場合は、PHY インタフェースクロックで、4 サイクルクロック時間以上経過してから、書換え処理を行う必要があります。***

4.8 GMAC Register 10 (RWFFR)

このレジスタは、Wake-up フレームの Filter パターンを設定します。

RWFFR (Remote Wake-up Frame Filter Register) Address 0028h

bit	31	30	29	28	27	26	25	24
Field	RWFFR[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	-	-	-	-	-	-	-	-
bit	23	22	21	20	19	18	17	16
Field	RWFFR[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	-	-	-	-	-	-	-	-
bit	15	14	13	12	11	10	9	8
Field	RWFFR[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0
bit	7	6	5	4	3	2	1	0
Field	RWFFR[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit[31:0] RWFFR[31:0] (Remote Wake-up Frame Filter Register)

このレジスタは、Wake-up フレームの Filter パターンを設定します。本品種では、4 種類の Filter パターンをプログラムすることが可能です。RWFFR は、Figure 4-1 のように 8 個のレジスタで構成されています。書き込むときは、8 回連続でレジスタ 0, レジスタ 1 の順番で 0x0028 に書き込みます。読み込む場合は、8 回連続でレジスタ 0, レジスタ 1 の順番で 0x0028 から読出しを行います。

RWFFRPR (PMTR レジスタの bit31) に 1 を書き込むことにより、本レジスタアクセスのポインタを初期化することができます。

Figure 4-1 RWFFR のレジスタ構成

RWFFR_0	Filter 0 Byte Mask							
RWFFR_1	Filter 1 Byte Mask							
RWFFR_2	Filter 2 Byte Mask							
RWFFR_3	Filter 3 Byte Mask							
RWFFR_4	予約	Filter 3 コマンド	予約	Filter 2 コマンド	予約	Filter 1 コマンド	予約	Filter 0 コマンド
RWFFR_5	Filter3 Offset		Filter2 Offset		Filter1 Offset		Filter0 Offset	
RWFFR_6	Filter 1 CRC-16				Filter 0 CRC-16			
RWFFR_7	Filter 3 CRC-16				Filter 2 CRC-16			

Filter i Byte Mask

このレジスタは、フレームが Wake-up フレームであるかどうかを決定するために、フィルタ i (0, 1, 2, 3)でフレームのどのバイトを調べるかを定義します。MSB (31 番目のビット)をゼロにする必要があります。ビット j [30:0]は、バイトマスクです。バイトマスクのビット j (バイト目)がセットされた場合、受信フレームの Filter i Offset + j が CRC ブロックによって処理されます。そうでない場合は、Filter i Offset + j は無視されます。

Filter i Command

この 4 ビットのコマンドは、フィルタ i の動作をコントロールします。bit3 は、パターンのディスティネーション・アドレスタイプを定義するアドレスタイプを指定します。このビットをセットすると、パターンはマルチキャストフレームにのみ適用します。このビットをリセットすると、パターンはユニキャストフレームにのみ適用します。bit2 および bit1 は予約されています。bit0 でフィルタ i はイネーブルされます。bit0 がセットされていない場合、フィルタ i はディセーブルされます。

Filter i Offset

このレジスタは、フレームの何バイト目からフィルタ i によってチェックするかのオフセットを定義します。この 8 ビットパターンオフセットは、チェックするフィルタ i の最初のバイトのオフセットです。最小設定は 12 です。これはフレームの 13 バイト目です(オフセット値 0 はフレームの最初のバイトです)。

Filter i CRC-16

このレジスタは、パターンから計算された CRC_16 値と、Wake-up フィルタレジスタブロックにプログラムされたバイトマスクを含んでいます。

CRC_16:	$x^{16}+x^{15}+x^2+1$
POLYNOMIAL:	0x8005
INITIAL_REMAINDER:	0xFFFF
FINAL_XOR_VALUE:	0x0000

4.9 GMAC Register 11 (PMTR)

このレジスタは、Wake-up リクエスト・イベントをプログラムし、Wake-up イベントをモニタします。

PMTR (PMT Register) Address 002Ch

bit	31	30	29	28	27	26	25	24
Field	RWFFRPR	予約						
属性	R/WS_SC	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約						GU	予約
属性	R	R	R	R	R	R	R/W	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約	WPR	MPR	予約		WFE	MPE	PD
属性	R	R_SS_RC	R_SS_RC	R	R	R/W	R/W	R/WS_SC
初期値	0	0	0	0	0	0	0	0

[bit31] RWFFRPR (Remote Wake-up Frame Filter Register Pointer Reset)

このビットをセットすると、Remote Wake-up Frame Filter レジスタポインタを 000 にリセットします。このビットは、1 クロックサイクル後に自動的にクリアされます。

[bit9] GU (Global Unicast)

このビットをセットすると、DA フィルタを通過したユニキャストパケットを Wake-up フレームとして認識します。

[bit6] WPR (Wake Up Frame Receive)

このビットがセットされると、Wake-up フレームの受信によりパワーマネジメントイベントが生成されたことを示します。このビットはこのレジスタへのリードによってクリアされます。

[bit5] MPR (Magic Packet Received)

このビットがセットされると、Magic Packet の受信によりパワーマネジメントイベントが生成されたことを示します。このビットはこのレジスタへのリードによってクリアされます。

[bit2] WFE (Wake-Up Frame Enable)

このビットをセットすると、Wake-up フレームの受信によるパワーマネジメントイベントの生成を許可します。

[bit1] MPE (Magic Packet Enable)

このビットをセットすると、Magic Packet の受信によるパワーマネジメントイベントの生成を許可します。

[bit0] PD (Power Down)

このビットをセットすると、すべての受信フレームが廃棄されます。このビットは、Magic Packet または Wake-up フレームが受信され、Power-Down モードがディセーブルされたとき、自動的にクリアされます。このビットがクリアされた後に受

信したフレームはアプリケーションに転送されます。このビットは、Magic Packet Enable, Global, Unicast, Wake- Up Frame Enable ビットが"1"にセットされているときのみセットできます。

* パワーマネジメントモード時に SYS_CLK を停止できます。しかし、SYS_CLK が停止しているときは、このレジスタに対し、リード・ライト動作は実行できません。したがってソフトウェアはこのビットをクリアできません。

4.10 GMAC Register 12 (LPICSR)

このレジスタは、LPI の機能をコントロールし、LPI 割込みステータスを提供します。

LPICSR (LPI Control and Status Register) Address 0030h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約				LPITXA	PLSEN	PLS	LPIEN
属性	R	R	R	R	R/W	R/W	R/W	R/W
初期値	-	-	-	-	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約						RLPIST	TLPIST
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	0	0

bit	7	6	5	4	3	2	1	0
Field	予約				RLPIEX	RLPIEN	TLPIEX	TLPIEN
属性	R	R	R	R	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	0	0	0	0	0	0	0	0

[bit19] LPITXA (LPI TX Automate)

このビットは、GMAC が送信側で LPI モードに入るかこのモードを終了するときの動作をコントロールします。
LPITXA および LPIEN ビットが 1 にセットされている場合、GMAC は、すべての未解決のフレームおよび保留中のフレームが送信された後でのみ LPI モードに入ります。GMAC は、アプリケーションが送信フレームを送信したとき、またはアプリケーションが TX FIFO Flush コマンドを発行したときに LPI モードを終了します。
GMAC は LPI ステートを終了したとき、LPIEN ビットを自動的にクリアします。
このビットが 0 のとき、GMAC が LPI モードに入るかこのモードを終了するときの動作は LPIEN ビットによって直接にコントロールされます。

[bit18] PLSEN (PHY Link Status Enable)

MII/RMII モード時は、Reserved ビットです。このビットにライトする場合は、常に 0 の書込みを行ってください。

[bit17] PLS (PHY Link Status)

このビットは、PHY のリンクステータスをライトするビットです。GMAC トランスミッタは、リンクステータスが LPI LS TIMER によって指定された時間以上の間 Up (OK)になっているときにのみ LPI パターンをアサートします。このビットをセットすると、リンクは OK(Up)と見なされ、リセットするとリンクは Down と見なされます。

[bit16] LPIEN (LPI Enable)

このビットをセットすると、GMAC トランスミッタに LPI ステートに入るよう指示します。このビットをリセットすると、GMAC に LPI ステートを終了して通常の送信を再開するよう指示します。

[bit9] RLPIST (Receive LPI State)

このビットがセットされると、GMAC が PHY インタフェース上で LPI パターンを受信していることを示します。

[bit8] TLPIST (Transmit LPI State)
このビットがセットされると、GMAC が PHY インタフェース上で LPI パターンを送信していることを示します。

[bit3] RLPIEX (Receive LPI Exit)
このビットがセットされると、GMAC レシーバが、PHY インタフェースで LPI パターンの受信を停止し、LPI ステートを終了し、通常の受信を再開したことを示します。このビットは、このレジスタへのリードによってクリアされます。

[bit2] RLPIEN (Receive LPI Entry)
このビットがセットされると、GMAC レシーバが、LPI パターンを受信し、LPI ステートに入ったことを示します。このビットは、このレジスタへのリードによってクリアされます。

＜注意事項＞

- *RLPIEN と RLPIEX の両方が 1 のとき、受信 LPI が終了していると見なす必要があります。*

[bit1] TLPIEX (Transmit LPI Exit)
このビットがセットされると、ユーザが LPIEN ビットをクリアし、LPI TW Timer が時間切れになった後、GMAC トランスミッタが LPI ステートを終了したことを示します。このビットはこのレジスタへのリードによってクリアされます。

[bit0] TLPIEN (Transmit LPI Entry)
このビットがセットされると、GMAC が LPIEN ビットのセットによって LPI ステートに入ったことを示します。このビットはこのレジスタへのリードによってクリアされます。

4.11 GMAC Register 13 (LPITCR)

このレジスタは、LPI ステートのタイムアウト値を設定します。GMAC が LPI パターンを送信する時間、および通常の送信を再開するまで待機する時間を指定します。

LPITCR (LPI Timers Control Register) Address 0034h

bit	31	30	29	28	27	26	25	24
Field	予約						LIT[9:8]	
属性	R	R	R	R	R	R	R/W	R/W
初期値	-	-	-	-	-	-	1	1

bit	23	22	21	20	19	18	17	16
Field	LIT[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	0	1	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TWT[15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TWT[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit25:16] LIT (LPI LS TIMER)

このビットは、LPI パターンを PHY に送信するために、PHY からのリンクステータスが Up (OK)になっていなければならない最小時間(単位:ms)を指定します。LPIEN ビットがセットされていても、LPI LS Timer がプログラムされたターミナルカウントに到達しない限り、GMAC は LPI パターンを送信しません。LPI LS Timer のデフォルト値は、IEEE 標準に従って 1000 (1 秒)です。

[bit15:0] TWT (LPI TW TIMER)

このビットは、GMAC が PHY への LPI パターンの送信を停止してから通常の送信を再開するまでの待機時間の最小値(単位:µs)を指定します。このタイマが時間切れになった後、TLPIEX ステータスビットがセットされます。

4.12 GMAC Register 14 (ISR)

このレジスタは、割込みステータスを表示します。

ISR (Interrupt Status Register) Address 0038h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約					LPIIS	TSIS	予約
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	COIS	TIS	RIS	MIS	PIS	予約	予約	RGIS
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit10] LPIIS (LPI Interrupt Status)

このビットは、GMAC トランスミッタまたはレシーバの LPI ステート開始または終了によりセットされます。このビットは GMAC Register 12 (LPI Control and Status Register)のバイト 0 のリードでクリアされます。

[bit9] TSIS (Time Stamp Interrupt Status)

このビットは、以下のときにセットされます。

- システムタイム値が Target Time High および Low レジスタで指定されている値、またはそれ以上になった
- 秒レジスタのオーバフローが発生した

このビットは、Time Stamp ステータスレジスタ(「4.29. GMAC Register 458 (TSR)」を参照)のバイト 0 のリードでクリアされます。

[bit7] COIS (MMC Receive Checksum Offload Interrupt Status)

このビットは、GMAC register 130(MMC Receive Checksum Offload Interrupt Register)で割込みが生成されたときに 1 にセットされます。このビットは、この割込みレジスタ内のすべてのビットがクリアされたときにクリアされます。

[bit6] TIS (MMC Transmit Interrupt Status)

このビットは、GMAC register 66(MMC Transmit Interrupt Register)で割込みが生成されたときに 1 にセットされます。このビットは、この割込みレジスタ内のすべてのビットがクリアされたときにクリアされます。

[bit5] RIS (MMC Receive Interrupt Status)

このビットは、GMAC register65(MMC Receive Interrupt Register)で割込みが生成されたときに 1 にセットされます。このビットは、この割込みレジスタ内のすべてのビットがクリアされたときにクリアされます。

[bit4] MIS (MMC Interrupt Status)

このビットは、COIS, TIS, RIS ビットのどれかが 1 にセットされたときに 1 にセットされ、COIS, TIS, RIS ビットのすべてが 0 になるとクリアされます。

[bit3] PIS (PMT Interrupt Status)

このビットは、Power- Down モードで Magic パケットまたは Wake-on-LAN フレームを受信したときにセットされます(GMAC Register 11（PMTR）の bit5 および bit6 を参照)。このビットは、PMT コントロール・ステータスレジスタへのリード動作により両方の bit[6:5]がクリアされたときにクリアされます。

[bit0] RGIS(RGMII Interrupt Status)

MII/RMII モード時は、Reserved ビットです。

4.13 GMAC Register 15 (IMR)

Interrupt Mask Register ビットによって、Interrupt Status Register の対応するイベントによる割込み信号(INT_SBD)をマスクできます。

IMR (Interrupt Mask Register) Address 003Ch

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約					LPIIM	TSIM	予約
属性	R	R	R	R	R	R/W	R/W	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約				PIM	予約	予約	RGIM
属性	R	R	R	R	R/W	R	R	R/W
初期値	0	0	0	0	0	0	0	0

[bit10] LPIIM(LPI Interrupt Mask)

このビットをセットすると、GMAC Register 14 で LPIIS (LPI Interrupt Status)ビットがセットされていることによる割込み信号のアサートが禁止されます。

[bit9] TSIM (Time Stamp Interrupt Mask)

このビットをセットすると、GMAC Register 14 で TSIS (Time Stamp Interrupt Status)ビットがセットされていることによる割込み信号のアサートが禁止されます。

[bit3] PIM (PMT Interrupt Mask)

このビットをセットすると、GMAC Register 14 で PIS (PMT Interrupt Status)ビットがセットされていることによる割込み信号のアサートが禁止されます。

[bit0] RGIM(RGMII Interrupt Mask)

MII/RMII モード時は、Reserved ビットです。

4.14 GMAC Register 16 (MAR0H)

MAC Address0 High レジスタは、ステーションの最初の MAC アドレス(6 バイト)の上位 16 ビットを設定します。

MAR0H (MAC Address0 Register (High)) Address 0040h

bit	31	30	29	28	27	26	25	24
Field	MO	予約						
属性	R	R	R	R	R	R	R	R
初期値	1	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	A0[47:40]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	7	6	5	4	3	2	1	0
Field	A0[39:32]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

PHY インタフェースで受信する送信先アドレスの最初のバイトは、MAC Address Low レジスタの LS バイト(bit[7:0])に対応します。例えば、PHY インタフェースでディスティネーション・アドレスとして 0x11:0x22:0x33:0x44:0x55:0x66 が受信された場合(0x11 が最初のバイト)、MAC Address0 Register [47:0]が 0x665544332211 と比較されます。

MAC Address[47:0] : UU:VV:WW:XX:YY:ZZ の場合の例を示します。

[7:0] = UU, [15:8] = VV, [23:16] = WW, [31:24] = XX, [39:32] = YY, [47:40] = ZZ
MARL : 0xXXWWVVYYUU
MARH : 0x8000ZZYY

[bit31] MO (常に 1)

常に 1 を設定します。

[bit15:0] A0[47:32] (MAC Address0[47:32])

このフィールドは、MAC アドレス 0 の上位 16 ビット[47:32]を設定します。GMAC はこの値を使用して受信フレームをフィルタリングし、Transmit Flow Control (PAUSE) Frames に MAC アドレスを挿入します。

<注意事項>

このレジスタへのライトアクセスは、32 ビット幅で行ってください。MAC アドレスを設定する際は、必ず、MAC アドレス上位レジスタ(MARxH)の書込みを行い、その次に MAC アドレス下位レジスタ(MARxL)の書込みを行います。MAC アドレス上位レジスタ(MARxH)への書込みだけを行った場合、あるいは書き込む順番を逆にした場合は、書き込んだ値が正しく認識されません。

4.15 GMAC Register 17 (MAR0L)

MAC Address0 Low レジスタは、ステーションの最初の MAC アドレス(6 バイト)の下位 32 ビットを設定します。

MAR0L (MAC Address0 Register (Low)) Address 0044h

bit	31	30	29	28	27	26	25	24
Field	A0[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	23	22	21	20	19	18	17	16
Field	A0[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	15	14	13	12	11	10	9	8
Field	A0[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	7	6	5	4	3	2	1	0
Field	A0[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

[bit31:0] A0[31:0] (MAC Address0[31:0])

このフィールドは、MAC アドレス 0 の下位 32 ビットを設定します。GMAC はこれを使って受信フレームをフィルタリングし、Transmit Flow Control (PAUSE) Frames に MAC アドレスを挿入します。

<注意事項>

- *このレジスタへのライトアクセスは、32 ビット幅で行ってください。このレジスタに一度値を書き込んだ後、異なる値に書換えを行う場合は、PHY インタフェースクロックで、4 サイクルクロック時間以上経過してから、書換え処理を行う必要があります。*

4.16 GMAC Register 18, 20, 22, …, 542 (MAR1H, 2H, 3H, …, 31H)

MAC Address High レジスタ(1-31)は、ステーションの 2～32 番目の MAC アドレス(6 バイト)の上位 16 ビットを設定します。

MAR1H～MAR31H　(MAC Address1～31 Register -High)

Address　0048h, 0050h, 0058h, 0060h, 0068h, 0070h, 0078h, 0080h, 0088h, 0090h, 0098h, 00A0h, 00A8h, 00B0h, 00B8h, 0800h, 0808h, 0810h, 0818h, 0820h, 0828h, 0830h, 0838h, 0840h, 0848h, 0850h, 0858h, 0860h, 0868h, 0870h, 0878h

bit	31	30	29	28	27	26	25	24
Field	AE	SA	MBC					
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	A [47:40]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	7	6	5	4	3	2	1	0
Field	A [39:32]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

[bit31] AE (Address Enable)

このビットをセットすると、アドレスフィルタは、MAC アドレス 1-31 を使用します。このビットをリセットすると、アドレスフィルタは、MAC アドレス 1-31 を無視します。

[bit30] SA (Source Address)

このビットをセットすると、MAC Address1[47:0]を受信フレームの SA フィールドとの比較に使用します。
このビットをリセットすると、MAC Address1[47:0]を受信フレームの DA フィールドとの比較に使用します。

[bit29:24] MBC(Mask Byte Control)

このビットは、各 MAC アドレスバイトの比較のためのマスクコントロールビットです。1 にセットすると、GMAC は、受信した DA/SA の対応するバイトを MAC Address*レジスタの内容と比較しません。各ビットは、以下のようにバイトのマスキングをコントロールします。

bit29:　MAC Address*[47:40]
bit28:　MAC Address*[39:32]
　　…
bit24:　MAC Address*[7:0]

[bit15:0] A

このフィールドは、2～32 番目の MAC アドレス(6 バイト)の上位 16 ビット(47:32)を設定します。

＜注意事項＞

- *このレジスタへのライトアクセスは、32 ビット幅で行ってください。MAC アドレスを設定する際は、必ず、MAC アドレス上位レジスタ(MARxH)の書込みを行い、その次に MAC アドレス下位レジスタ(MARxL)の書込みを行います。MAC アドレス上位レジスタ(MARxH)への書込みだけを行った場合、あるいは書き込む順番を逆にした場合は、書き込んだ値が正しく認識されません。AE, SA, MBC の各フィールドへの書込み値は、アドレス上位レジスタ(MARxH)への書込みだけでは有効にならず、次の MAC アドレス下位レジスタ(MARxL)の書き込み後に有効になります。*

4.17 GMAC Register 19, 21, 23, …, 543 (MAR1L, 2L, 3L, …, 31L)

MAC Address Low レジスタ(1-31)は、ステーションの 2～32 番目の MAC アドレス(6 バイト)の下位 32 ビットを設定します。

MAR1L～MAR31L (MAC Address1～31 Register -Low)

Address 004Ch, 0054h, 005Ch, 0064h, 006Ch, 0074h, 007Ch, 0084h, 008Ch, 0094h, 009Ch, 00A4h, 00ACh, 00B4h, 00BCh, 0804h, 080Ch, 0814h, 081Ch, 0824h, 082Ch, 0834h, 083Ch, 0844h, 084Ch, 0854h, 085Ch, 0864h, 086Ch, 0874h, 087Ch

bit	31	30	29	28	27	26	25	24
Field	A [31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	23	22	21	20	19	18	17	16
Field	A [23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	15	14	13	12	11	10	9	8
Field	A [15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

bit	7	6	5	4	3	2	1	0
Field	A [7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	1	1	1	1	1	1	1	1

[bit31:0] A [31:0]

このフィールドは、2～32 番目の MAC アドレス(6 バイトの)の下位 32 ビットを含みます。このフィールドの内容は、初期化プロセスの後アプリケーションによってロードされるまで未定義のままです。

＜注意事項＞

- *このレジスタへのライトアクセスは、32 ビット幅で行ってください。このレジスタに一度値を書き込んだ後、異なる値に書換えを行う場合は、PHY インタフェースクロックで、4 サイクルクロック時間以上経過してから、書換え処理を行う必要があります。*

4.18 GMAC Register 54 (RGSR)

RGMII ステータスレジスタは、RGMII が PHY から受信したステータス信号を示します。

RGSR (RGMII Status Register) Address 00D8h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	予約				LS	LSP		LM
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	0	0	0	0

[bit3] LS (Link Status)

MII/RMII モード時は、予約ビットです。

[bit2, 1] LSP (Link Speed)

MII/RMII モード時は、予約ビットです。

[bit0] LM(Link Mode)

MII/RMII モード時は、予約ビットです。

4.19 GMAC Register 448 (TSCR)

このレジスタは、レシーバのタイムスタンプ生成のためのシステムタイムジェネレータの動作と PTP パケットのスヌーピングをコントロールします。

TSCR (Time Stamp Control Register) Address 0700h

bit	31	30	29	28	27	26	25	24
Field	予約							ATSFC
属性	R	R	R	R	R	R	R	R/WSC
初期値	-	-	-	-	-	-	-	0

bit	23	22	21	20	19	18	17	16
Field	予約					TSENMF	TSPS	
属性	R	R	R	R	R	R/W	R/W	R/W
初期値	-	-	-	-	-	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSMRM	TETSEM	TSIP4E	TSIP6E	TETSP	TSV2E	TSDB	TSEA
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	1	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約		TARU	TITE	TSU	TSI	TFCU	TSE
属性	R	R	R/WSC	R/WSC	R/WSC	R/WSC	R/W	R/W
初期値	-	-	0	0	0	0	0	0

[bit24] ATSFC (Auxiliary Snapshot FIFO Clear)

本レジスタは、Reserved です。書込みの際は 0 を書き込んでください。

[bit18] TSENMF(Enable MAC address for PTP frame filtering)

このビットをセットすると、Ethernet 上に直接送信された PTP フレームのフィルタリングのために DA MAC アドレス(MAC Address レジスタ 0-31 とマッチする) が使用されます。

[bit17,16] TSPS (Select PTP packet for taking snapshots)

これらのビットは、TSMRM, TETSEM ビットとの組合せによって、スナップショットを生成する必要がある PTP パケットタイプのセットを決定します。これらのビットのエンコードを Table 4-3 に示しています。

[bit15] TSMRM(Enable Snapshot for Messages Relevant to Master)

このビットをセットすると、マスタノードに関連するメッセージのスナップショットだけが生成されます。セットしていない場合は、スレーブノードに関連するメッセージのスナップショットが生成されます。これは通常クロックノードと境界クロックノードにのみ有効です。

[bit14] TETSEM(Enable Time Stamp Snapshot for Event Messages)

このビットをセットすると、イベント・メッセージ(SYNC, Delay_Req, Pdelay_Req, または Pdelay_Resp)のタイムスタンプスナップショットだけが生成されます。このビットをリセットすると、Announce, Management, および Signaling を除くほかのすべてのメッセージのスナップショットが生成されます。

[bit13] TSIP4E (Enable Time Stamp Snapshot for IPv4 frames)

このビットをセットすると、IPv4 フレームのタイムスタンプスナップショットが生成されます。

[bit12] TSIP6E (Enable Time Stamp Snapshot for IPv6 frames)
このビットをセットすると、IPv6 フレームのタイムスタンプスナップショットが生成されます。

[bit11] TETSP (Enable Time Stamp Snapshot for PTP over Ethernet frames)
このビットをセットすると、Ethernet フレーム(Ethernet 上の PTP)に PTP メッセージがあるフレームのタイムスタンプスナップショットも生成されます。デフォルトでは、UDP-IP-Ethernet PTP パケットのタイムスタンプスナップショットが生成されます。

[bit10] TSV2E (Enable PTP packet snooping for version 2 format)
このビットをセットすると、PTP パケットは 1588 バージョン 2 フォーマットを使用してスヌープされます。このビットをリセットすると、バージョン 1 フォーマットを使用してスヌープされます。

[bit9] TSDB (Time Stamp Digital or Binary rollover control)
このビットをセットすると、Time Stamp Low レジスタは 0x3B9A_C9FF 値(精度 1 ns)後でロールオーバし、Time Stamp (High) の秒数をインクリメントします。このビットをリセットすると、秒未満レジスタのロールオーバ値は 0x7FFF_FFFF となります。秒未満の増分は、PTP リファレンスクロック周波数およびこのビットの値に従って正確にプログラムされる必要があります。

[bit8] TSEA (Enable Time Stamp for All Frames)
このビットをセットすると、GMAC が受信したすべてのフレームのタイムスタンプスナップショットが許可されます。

[bit5] TARU (Addend Register Update)
このビットをセットすると、Fine correction のために Time Stamp Addend レジスタの PTP ブロックの内容が更新されます。更新が完了したとき、このビットがクリアされます。このレジスタビットは、セットする前に 0 にしておく必要があります。これは coarse correction オプションだけが選択されているときは予約ビットとなります。

[bit4] TITE (Time Stamp Interrupt Trigger Enable)
このビットをセットすると、System Time が Target Time レジスタに書き込まれている値よりも大きくなったとき、Time Stamp 割込みが生成されます。このビットは、Time Stamp Trigger Interrupt が生成された後にリセットされます。

[bit3] TSU (Time Stamp Update)
このビットをセットすると、システムタイムが Time Stamp High Update および Time Stamp Low Update レジスタによって指定された値によって更新(加算/減算)されます。このレジスタビットは、更新される前は 0 でなければなりません。ハードウェアで更新が完了した後、このビットがリセットされます。Time Stamp Higher Word レジスタは更新されません。

[bit2] TSI (Time Stamp Initialize)
このビットをセットすると、システムタイムが Time Stamp High Update および Time Stamp Low Update レジスタによって指定された値によって更新(上書き)されます。このレジスタビットは、更新される前は 0 でなければなりません。初期化が完了した後、このビットは、リセットされます。Time Stamp Higher Word レジスタは初期化することのみができます。

[bit1] TFCU (Time Stamp Fine or Coarse Update)
このビットをセットすると、システムタイムの更新が Fine 更新方法を使って行われます。このビットをリセットすると、システムタイムスタンプの更新が Coarse 更新方法を使って行われます。

[bit0] TSE (Time Stamp Enable)
このビットをセットすると、送信および受信フレームのシステムタイムスタンプの生成が許可されます。このビットをリセットすると、送信および受信フレームに対してタイムスタンプは生成されず、TimeStamp Generator は一時停止します。このモードをイネーブルした後、必ず TimeStamp (システムタイム)を初期化しなければなりません。

Table 4-4 タイムスタンプスナップショットとレジスタビットの関係

TSPS	TSMRM	TETSEM	スナップショットが生成されるメッセージ
00	X	0	SYNC, Follow_Up, Delay_Req, Delay_Resp
00	0	1	SYNC
00	1	1	Delay_Req
01	X	0	SYNC, Follow_Up, Delay_Req, Delay_Resp, PDelay_Req, PDelay_Resp, PDelay_Resp_Follow_Up
01	0	1	SYNC, PDelay_Req, PDelay_Resp
01	1	1	Delay_Req, Pdelay_Req, Pdelay_Resp
10	X	X	SYNC, Delay_Req
11	X	X	PDelay_Req, PDelay_Resp

4.20 GMAC Register 449 (SSIR)

Coarse Update モード(Time Stamp Control Register の TFCU ビット)では、このレジスタ内の値は PTP_CLK のクロックサイクルごとにシステムタイムに加算されます。Fine Update モードでは、このレジスタ内の値はアキュムレータがオーバフローになったときにシステムタイムに加算されます。

SSIR (Sub-Second Increment Register) Address 0704h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R/WSC	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	SSINC							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit7:0] SSINC (Sub-Second Increment Value)

このレジスタでプログラムされた値がサブセカンドレジスタの内容に応じて累積されます。例えば、20ns の精度を得るには、値 20 にプログラムします(1ns の精度が選択されている場合、50MHz リファレンスクロックで 0x14 になります)。

4.21 GMAC Register 450 (STSR)

System Time - Seconds レジスタは、System Time - Nanoseconds レジスタと合わせて、現在保持されているシステムタイムの値を示します。この値は継続的に更新されますが、クロックドメイン転送遅延(PTP_CLK から SYS_CLK)のために実際の時間からは少し遅延します。

STSR (System Time - Seconds Register) Address 0708h

bit	31	30	29	28	27	26	25	24
Field	TSS[31:24]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSS[23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSS[15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSS[7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit31:0] TSS (Time Stamp Second)

このフィールドの値は、保持されている現在の System Time の秒の値を示します。

4.22 GMAC Register 451 (STNR)

System Time - Nanoseconds レジスタは、System Time - Seconds レジスタと合わせて使用します。

STNR (System Time - Nanoseconds Register) Address 070Ch

bit	31	30	29	28	27	26	25	24
Field	予約	TSSS[30:24]						
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSSS[23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSSS[15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSSS[7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit30:0] TSSS (Time Stamp Sub-Seconds)

このフィールドの値は、時間の秒以下の部分を表し、精度は 0.46 ns です(TSDB がセットされているとき、各ビットは 1ns を表し、最大値は 0x3B9A_C9FF になり、それを超えるとロールオーバし、0 になります)。

4.23 GMAC Register 452 (STSUR)

System Time - Seconds Update レジスタは、System Time - Nanoseconds Update レジスタと合わせて、保持されているシステムタイムを初期化または更新します。Time Stamp Control レジスタで TSINIT または TSUPDT ビットをセットする前に、この両方のレジスタに書き込む必要があります。

STSUR (System Time - Seconds Update Register) Address 0710h

bit	31	30	29	28	27	26	25	24
Field	TSS[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSS[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	

bit	15	14	13	12	11	10	9	8
Field	TSS[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSS[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit31:0] TSS (Time Stamp Second)

このフィールドの値は、システムタイムの初期値またはシステムタイムに加算する時間を秒単位で示します。

4.24 GMAC Register 453 (STNUR)

System Time - Nanoseconds Update レジスタは、System Time - Seconds Update レジスタと合わせて使用します。

STSNUR (System Time - Nanoseconds Update Register) Address 0714h

bit	31	30	29	28	27	26	25	24
Field	ADDSUB	TSSS[30:24]						
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSSS[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSSS[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSSS[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit31] ADDSUB (Add or Subtract Time)

このビットをセットすると、タイム値から更新レジスタの内容が減算されます。このビットをリセットすると、タイム値に更新レジスタの内容が加算されます。

[bit30:0] TSSS (Time Stamp Sub-Seconds)

このフィールドの値は、時間の秒以下の部分を表し、精度は 0.46 ns です(タイムスタンプコントロールレジスタで TSDB がセットされているとき、各ビットは 1ns を表し、プログラムできる最大値は 0x3B9A_C9FF です)。

4.25 GMAC Register 454 (TSAR)

このレジスタ値は、システムタイムが Fine Update モードに設定されている(GMAC Register 448 の TFCU ビット)ときにのみ使用されます。PTP_CLK のクロックサイクルごとにこのレジスタの内容が 32 ビットのアキュムレータに加算され、アキュムレータがオーバフローになったときにシステムタイムが更新されます。

TSAR (Time Stamp Addend Register) Address 0718h

bit	31	30	29	28	27	26	25	24
Field	TSAR[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSAR[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSAR[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSAR[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit31:0] TSAR (Time Stamp Addend Register)

このレジスタは、時間同期のためにアキュムレータレジスタに加算される 32 ビット値を示します。

4.26 GMAC Register 455 (TTSR)

Target Time Seconds register は、Target Time Nanoseconds レジスタと合わせて使用され、システムタイムがこれらのレジスタでプログラムされている値を超えたときに割込みイベント(レジスタ 458 の TSTART ビット, またはレジスタ 14[9]の TSIS ビット)をスケジュールします。

TTSR (Target Time Seconds Register) Address 071Ch

bit	31	30	29	28	27	26	25	24
Field	TSTR[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSTR[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	

bit	15	14	13	12	11	10	9	8
Field	TSTR[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSTR[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit31:0] TSTR (Target Time Stamp Seconds Register)

このレジスタは時間を秒単位で保存します。タイムスタンプ値が両方の Target Time Stamp レジスタの値と一致するかそれを超えているとき、GMAC は(イネーブルされている場合)割込みを生成します。

4.27 GMAC Register 456 (TTNR)

Target Time Nanoseconds レジスタは、Target Time Seconds register と合わせて使用します。

TTNR (Target Time Nanoseconds Register) Address 0720h

bit	31	30	29	28	27	26	25	24
Field	予約	TSTR[30:24]						
属性	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	-	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	TSTR[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TSTR[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSTR[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit30:0] TSTR (Target Time Stamp Nanoseconds Register)

このレジスタは時間を(符号付き) ns 単位で保存します。Time Stamp の値が両方の Target Time Stamp レジスタの値と一致するとき、GMAC は(イネーブルされている場合)割込みを生成します(TSDB がタイムスタンプコントロールレジスタでセットされているとき、この値は 0x3B9A_C9FF を超えてはいけません)。

4.28 GMAC Register 457 (STHWSR)

このレジスタは、タイムスタンプの 48 ビット秒値の上位 16 ビットを読み出します。

STHWSR (System Time - Higher Word Seconds Register) Address 0724h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	TSHWR[15:8]							
属性	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TSHWR[7:0]							
属性	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU	R/WSU
初期値	0	0	0	0	0	0	0	0

[bit15:0] TSHWR (Time Stamp Higher Word Register)

タイムスタンプの秒値の上位 16 ビットを含みます。値を初期化するときは、このレジスタに直接に書き込みます。このレジスタは、System Time - Seconds レジスタの 32 ビットからのオーバフローがあるとき、インクリメントされます。

4.29 GMAC Register 458 (TSR)

このレジスタは、システムタイムカウンタへの操作のステータスを表示します。

TSR (Time Stamp Status Register) Address 0728h

bit	31	30	29	28	27	26	25	24
Field	予約				ATSNS			ATSSTM
属性	R	R	R	R	R	R	R	R_SS_RC
初期値	-	-	-	-	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	予約				TRGTER	ATSTS	TSTART	TSSOVF
属性	R	R	R	R	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	-	-	-	-	-	0	0	0

このレジスタがホストによってリードされた後、[bit27:25]を除くすべてのビットがクリアされます。

[bit27:25] ATSNS(Auxiliary Time Stamp Number of Snapshots)

予約ビットです。

[bit24] ATSSTM(Auxiliary Time Stamp Snapshot Trigger Missed)

予約ビットです。

[bit3] TRGTER (Time Stamp Target Time Error)

Target Time レジスタ（GMAC Register 455(TTSR)、GMAC Register 456(TTNR)）にプログラムする際、既にその時間を経過している場合、このビットがセットされます。このビットは、アプリケーションからの読出しによりクリアされます。

[bit2] ATSTS(Auxiliary Time Stamp Trigger Snapshot)

予約ビットです。

[bit1] TSTART (Time Stamp Target Time Reached)

このビットがセットされると、システムタイムの値が Target Time Seconds および Nanoseconds レジスタで指定された値と等しいかそれより大きいことを示します。

[bit0] TSSOVF(Time Stamp Seconds Overflow)

このビットがセットされると、タイムスタンプの秒値(バージョン 2 のフォーマットをサポートする場合)が 0xFFFF_FFFF を超えてオーバフローしたことを示します。

4.30 GMAC Register 459 (PPSCR)

このレジスタは、PTPPPS 出力端子の制御を行います。

PPSCR (PPS Control Register) Address 072Ch

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	-	-	-	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	予約				PPSCTRL[3:0]			
属性	R	R	R	R	R/W	R/W	R/W	R/W
初期値	-	-	-	-	0	0	0	0

[bit3:0] PPSCTRL (Controls the frequency of the PPS output)

これらのビットは、PPS 出力(PTPPPS)信号の動作をコントロールします。PPSCTRL のデフォルト値は 0000 で、PPS 出力は 1 パルス(幅 PTP_CLK)/秒です。

PPSCTRL がほかの値のとき、PPS 出力は以下の周波数の生成クロックになります。

PPSCTRL	バイナリロールオーバ	デジタルロールオーバ
0001	2Hz	1Hz
0010	4Hz	2Hz
0011	8Hz	4Hz
0100	16Hz	8Hz
…		
1111	32.768kHz	16.384kHz

<注意事項>

バイナリロールオーバモードでは、これらの周波数での PPS 出力(PTPPPS)のデューティサイクルは 50%です。

デジタルロールオーバモードでは、PPS 出力周波数は平均値です。実際のクロックは周波数が異なり、1 秒ごとに同期されます。例えば、

- *PPSCTRL = 0001 のとき、PPS (1 Hz)は Low 時間が 537 ms で、High 時間が 463 ms です。*
- *PPSCTRL = 0010 のとき、PPS (2 Hz)は以下のシーケンスになります。*

・最初のクロックは 50%デューティサイクル、時間が 537 ms

・2 番目のクロックは時間が 463 ms (Low 時間が 268 ms, High 時間が 195 ms)

- *PPSCTRL = 0011 のとき、PPS (4 Hz)は以下のシーケンスになります。*

・最初の 3 つのクロックは 50%デューティサイクル、時間が 268 ms

・4 番目のクロックは時間が 195 ms (Low 時間が 134 ms, High 時間が 61 ms)

この動作は、GMAC Register 451 (STNR)のデジタルロールオーバモードでのビットの非直線的なトグルによるものです。

4.31 GMAC Register 460 (ATNR)

このレジスタは、Reserved レジスタです。

ATNR (Auxiliary Time Stamp - Nanoseconds Register) Address 0730h

bit	31	30	29	28	27	26	25	24
Field	予約	ATN						
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	ATN							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	ATN							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	ATN							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

このレジスタは、Reserved レジスタです。

4.32 GMAC Register 461 (ATSR)

このレジスタは、Reserved レジスタです。

ATSR (Auxiliary Time Stamp - Seconds Register) Address 0734h

bit	31	30	29	28	27	26	25	24
Field	ATS							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	ATS							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	ATS							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	ATS							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

このレジスタは、Reserved レジスタです。

4.33 DMA Register 0 (BMR)

このレジスタは、DMA の動作モードを設定します。

BMR (Bus Mode Register) Address 1000h

bit	31	30	29	28	27	26	25	24
Field	予約				TXPR	MB	AAL	8xPBL
属性	R	R	R	R	R/W	R/W	R/W	R/W
初期値	-	-	-	-	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	USP	RPBL						FB
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	1	0

bit	15	14	13	12	11	10	9	8
Field	PR		PBL[5:0]					
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	1

bit	7	6	5	4	3	2	1	0
Field	ATDS	DSL[4:0]					DA	SWR
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/WS_SC
初期値	0	0	0	0	0	0	0	1

[bit27] TXPR (Transmit Priority)

このビットをセットすると、AHB マスタ・インタフェースでのバス・アービトレーションで、送信 DMA が受信 DMA よりもプライオリティが高くなります。

[bit26] MB (Mixed Burst)

このビットを 1, FB ビットを 1 にしたとき、AHB マスタ・インタフェースはバースト長が 16 を超えるすべてのバーストを INCR (不定長バースト)で開始し、バースト長が 16 以下の場合は固定長バースト転送(INCRx および SINGLE)に戻ります。

[bit25] AAL (Address-Aligned Beats)

このビットは Reserved です。常に 0 をライトして使用します。

[bit24] 8xPBL (8xPBL Mode)

このビットを 1 にセットしたとき、プログラムされている PBL 値 (bit[22:17]および bit[13:8])を 8 倍にします。したがって、DMA は、PBL 値に従ってデータを最大 8, 16, 32, 64, および 128 ビットにデータを転送します。

[bit23] USP (Use Separate PBL)

このビットを 1 にセットしたとき、受信 DMA が bit[22:17]で設定されている値を PBL として使用するように設定します。bit[13:8]の PBL 値は送信 DMA 動作にのみ適用されます。このビットを 0 にリセットすると、bit[13:8]の PBL 値が両方の DMA エンジンに適用されます。

[bit22:17] RPBL (RxDMA PBL)

この値は、1 回の RxDMA トランザクションで実施される不定長バースト (INCR) の最大数を指定します。受信 DMA は、ホストバス上でバースト転送を開始するとき、常に RPBL によって指定されたとおりバーストを試みます。RPBL はプログラム可能で、許容値は 1, 2, 4, 8, 16, 32 です。ほかの値を指定したときの動作は未定義です。これらのビットは USP が 1 にセットされているときのみ有効で、適用可能です。

[bit16] FB (Fixed Burst)

このビットは、AHB マスタ・インタフェースが固定長バースト転送を実行するか否かをコントロールします。このビットをセットすると、AHB は、通常のバースト転送の開始時に、SINGLE, INCR4, INCR8, または INCR16 のみを使用します。このビットをリセットすると、AHB は、SINGLE および INCR バースト転送動作を使用します。

[bit15, 14] PR (Rx:Tx priority ratio)

このビットは、受信 DMA と送信 DMA の間の重み付きラウンドロビンアービトレーションでのプライオリティ比をコントロールします。このビットは、DMA Arbitration Scheme (DA: bit1)=0 時のみ有効です。以下のプライオリティ比記載は、Transmit Priority (TXPR: bit27)=0 時、受信:送信, bit27=1 時、送信:受信です。

00 : 1:1
01 : 2:1
10 : 3:1
11 : 4:1

[bit13:8] PBL (Programmable Burst Length)

この値は、1 回の DMA トランザクションで実施される不定長バースト (INCR) の最大数を指定します。DMA は、ホストバス上でバースト転送を開始するとき、常に PBL によって指定されたとおりバーストを試みます。PBL はプログラム可能で、許容値は 1, 2, 4, 8, 16, 32 です。ほかの値を指定したときの動作は未定義です。USP が 1 にセットされているとき、この PBL 値は送信 DMA トランザクションにのみ適用されます。

8xPBL がセットされているとき、PBL の最大値は 16 です。

転送するビットの数が 33 以上である場合、以下の手順を実行します。

1. 8xPBL モードをセットします。
2. PBL をセットします。

例えば、転送するビットの最大数が 64 である場合、最初に 8xPBL を 1 にセットし、次に PBL を 8 にセットします。

[bit7] ATDS(Alternate Descriptor Size)

このビットを 1 にセットすると、ディスクリプタのサイズが 32 バイト(8 word)になります。これはタイムスタンプ機能がイネーブルされるときに必要となります。このビットを 0 にリセットすると、ディスクリプタのサイズが 16 バイト (4 word) となり、TDES4-7, RDES4-7 に対する処理が行われなくなります。

[bit6:2] DSL [4:0](Descriptor Skip Length)

このビットは、2 つの Chain されていないディスクリプタの間でスキップする Word 数を指定します。現在のディスクリプタの終わりから次のディスクリプタの開始までアドレススキッピングが行われます。Ring モードで、DSL 値が 0 のとき、DMA はディスクリプタテーブルが隣接していると見なします。

[bit1] DA (DMA Arbitration scheme)

0: Rx:Tx または Tx:Rx を指定した重み付きラウンドロビンが選択されます。
パス間のプライオリティは、PR と TXPR で指定されたプライオリティの重みに従います。

1: 固定プライオリティ。
TXPR=1 時、送信 DMA が受信 DMA に優先します。TXPR=0 時、受信 DMA が送信 DMA に優先します。

[bit0] SWR (Software Reset)

このビットを 1 にセットすると、DMA Controller はすべての内部レジスタおよびロジックをリセットします。このビットは、すべてのクロックドメインでリセット動作が完了した後、自動的にクリアされます。いずれかのレジスタを再プログラムするときは、その前にこのビットから 0 が読み出せる状態である必要があります。

＜注意事項＞

- *リセット動作は、すべてのアクティブクロック内のすべてのリセットがデアサートされているときにのみ完了します。したがって、ソフトリセットが完了するためには、(選択した PHY インタフェースに適用される)すべての PHY 入力クロックが存在することが不可欠です。*

4.34 DMA Register 1 (TPDR)

Transmit Poll Demand レジスタは、送信 DMA に対して、現在のディスクリプタが DMA によって所有されているかどうかをチェックすることを指示します。

TPDR (Transmit Poll Demand Register) Address 1004h

bit	31	30	29	28	27	26	25	24
Field	TPD[31:24]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0
bit	23	22	21	20	19	18	17	16
Field	TPD[23:16]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0
bit	15	14	13	12	11	10	9	8
Field	TPD[15:8]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0
bit	7	6	5	4	3	2	1	0
Field	TPD[7:0]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0

Transmit Poll Demand コマンドは、送信 DMA が Suspend モードの場合、ウェイクアップするために発行します。送信 DMA は、送信フレームの Underflow エラーや、所有されているディスクリプタが利用できないために Suspend モードになることがあります。このコマンドはいつでも発行でき、送信 DMA はホストメモリから現在のディスクリプタの再取込みを開始した後でこのコマンドをリセットします。

[bit31:0] TPD (Transmit Poll Demand)

このビットに何らかの値が書き込まれたとき、DMA は DMA Register 18 (CHTDR)によって指定されている現在のディスクリプタを読み出します。そのディスクリプタが利用できない場合(ホストによって所有されている場合)、送信は Suspend ステートに戻り、DMA Register 5[2]がアサートされます。ディスクリプタが利用可能である場合、送信が再開されます。

4.35 DMA Register 2 (RPDR)

Receive Poll Demand レジスタは、受信 DMA が新しいディスクリプタをチェックすることを指示します。

RPDR (Receive Poll Demand Register) Address 1008h

bit	31	30	29	28	27	26	25	24
Field	RPD[31:24]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	RPD[23:16]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	RPD[15:8]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	RPD[7:0]							
属性	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT	R/WT
初期値	0	0	0	0	0	0	0	0

受信 DMA を Suspend ステートからウェイクアップするためにこのコマンドを発行します。受信 DMA が Suspend ステートになるのは、受信 DMA によって所有されているディスクリプタが存在しない場合だけです。

[bit31:0] RPD[31:0] (Receive Poll Demand)

これらのビットに何らかの値が書き込まれたとき、DMA は DMA Register 19 によって指定されている現在のディスクリプタを読み出します。そのディスクリプタが利用できない場合(ホストによって所有されている場合)、受信は Suspended ステートに戻り、DMA Register 5[7]がアサートされます。ディスクリプタが利用可能である場合、受信 DMA はアクティブステートに戻ります。

4.36 DMA Register 3 (RDLAR)

Receive Descriptor List Address レジスタは、Receive Descriptor List の先頭を指定します。

RDLAR (Receive Descriptor List Address Register) Address 100Ch

bit	31	30	29	28	27	26	25	24
Field	SRL[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	SRL[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	SRL[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	SRL[7:2]						Must be 0	
属性	R/W	R/W	R/W	R/W	R/W	R/W	R	R
初期値	0	0	0	0	0	0	0	0

ディスクリプタリストはホストの物理メモリスペースにあり、ワードアラインされていなければなりません。DMA は、対応する LS ビットを 0 にすることによって、それをバス幅にアライメントされたアドレスに内部的に変換します。DMA Register 3 への書込みは、受信が停止しているときのみ行えます。受信停止しているとき、受信 Start コマンドを発行する前に DMA Register 3 への書込みを行わなければなりません。

[bit31:2] SRL[31:2] (Start of Receive List)

このフィールドは、Receive Descriptor リスト内の最初のディスクリプタのベースアドレスを含みます。

[bit1:0] Must be 0

これらのビットには、必ず 0 を書き込んでください。

4.37 DMA Register 4 (TDLAR)

Transmit Descriptor List Address レジスタは、Transmit Descriptor List の先頭を指定します。

TDLAR (Transmit Descriptor List Address Register) Address 1010h

bit	31	30	29	28	27	26	25	24
Field	STL[31:24]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	STL[23:16]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	STL[15:8]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	STL[7:2]						Must be 0	
属性	R/W	R/W	R/W	R/W	R/W	R/W	R	R
初期値	0	0	0	0	0	0	0	0

ディスクリプタリストはホストの物理メモリスペースにあり、ワードアラインされていなければなりません。DMA は、対応する LS ビットを 0 にすることによってそれをバス幅アライメントされたアドレスに内部的に変換します。DMA Register 4 への書込みは、送信が停止しているときのみ許可されます。送信停止しているとき、送信 Start コマンドを発行する前に DMA Register 4 への書込みを行うことができます。

[bit31:2] STL (Start of Transmit List)

このフィールドは、Transmit Descriptor リスト内の最初のディスクリプタのベースアドレスを含みます。

[bit1:0] Must be 0

これらのビットには、必ず 0 を書き込んでください。

4.38 DMA Register 5 (SR)

ステータスレジスタは、DMA がホストにレポートするすべてのステータスビットを含みます。

SR (Status Register) Address 1014h

bit	31	30	29	28	27	26	25	24
Field	予約	GLPII	TTI	GPI	GMI	GLI	EB[2:1]	
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	EB[0]	TS[2:0]			RS[2:0]			NIS
属性	R	R	R	R	R	R	R	R_SS_WC
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	AIS	ERI	FBI	予約		ETI	RWT	RPS
属性	R_SS_WC	R_SS_WC	R_SS_WC	R	R	R_SS_WC	R_SS_WC	R_SS_WC
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	RU	RI	UNF	OVF	TJT	TU	TPS	TI
属性	R_SS_WC	R_SS_WC	R_SS_WC	R_SS_WC	R_SS_WC	R_SS_WC	R_SS_WC	R_SS_WC
初期値	0	0	0	0	0	0	0	0

DMA Register 5 は通常、割込みサービスルーチンまたはポーリング中にソフトウェアドライバによって読み込まれます。このレジスタのほとんどのフィールドは、ホストへの割込みを発生させます。DMA Register 5 のビットは、読出しによってクリアされません。DMA Register 5[16:0]の Reserved 以外のビットに 1 を書き込むとそれらのビットがクリアされます。0 の書込みは無効です。各フィールド(bit[16:0])は、DMA Register 7 で該当するビットをマスクすることによってマスクできます。

[bit30] GLPII (GMAC LPI Interrupt)

このビットは、GMAC の LPI ロジック内の割込みイベントを示します。このビットを 0 にリセットするには、ソフトウェアは対応するレジスタを読み出し、割込みの正確な原因を突き止め、その原因を除去しなければなりません。このビットが 1 のとき、割込み信号(INT_SBD および INT_LPI)がアサートされています。

[bit29] TTI (Time-Stamp Trigger Interrupt)

このビットは、GMAC の Time Stamp Generator ブロック内の割込みイベントを示します。このビットを 0 にリセットするには、ソフトウェアは対応するレジスタを読み出し、割込みの正確な原因を突き止め、その原因を除去しなければなりません。このビットが 1 のとき、割込み信号(INT_SBD)がアサートされています。

[bit28] GPI (GMAC PMT Interrupt)

このビットは、Wake-up Frame または Magic Packet の割込みイベントを示します。このビットを 0 にリセットするには、ソフトウェアは対応するレジスタを読み出し、割込みの正確な原因を突き止め、その原因を除去しなければなりません。このビットが 1 のとき、割込み信号(INT_SBD および INT_PMT)がアサートされています。

<注意事項>

- *INT_PMT は RX_CLK ドメインで生成されるため、PMT ステータスレジスタへのリード時のクリアは即時には行われません。生成されるクリア信号が RX_CLK ドメインへ送られ、そのときに割込みソースがクリアされます。この遅延は、RX_CLK の 4 クロックサイクル以上であり、Ethernet-MAC が 10 Mbps モードで動作しているときには重大な影響を及ぼすことがあります。*

[bit27] GMI (GMAC MMC Interrupt)
このビットは、MMC レジスタ内の割込みイベントを反映します。このビットを 0 にリセットするには、ソフトウェアは対応するレジスタを読み出し、割込みの正確な原因を突き止め、その原因を除去しなければなりません。このビットが 1 のとき、割込み信号(INT_SBD)がアサートされています。

[bit26] GLI(GMAC Line interface Interrupt)
このビットは、予約ビットです。

[bit25:23] EB (Error Bits)
これらのビットは､AHB バスアクセス時のバスエラーの原因となったエラーのタイプを示します｡これは Fatal Bus Error ビット(DMA Register 5[13])がセットされているときのみ有効です。このフィールドは割込みを生成しません。

bit23=1 :送信 DMA のデータ転送中のエラー
bit23=0 :受信 DMA のデータ転送中のエラー
bit24=1 :リード転送中のエラー
bit24=0 :ライト転送中のエラー
bit25=1 :ディスクリプタアクセス中のエラー
bit25=0 :データバッファアクセス中のエラー

[bit22:20] TS (Transmit Process State)
これらのビットは、送信 DMA の状態を示します。このフィールドは割込みを生成しません。

000 : Stopped - Reset または Stop Transmit コマンドが発行された

001 : Running - 送信転送ディスクリプタ(Transmit Transfer Descriptor)の取込み中

010 : Running - ステータス待ち

011 : Running - メモリバッファからデータを読み出し、データを送信バッファのキューに入れている

100 : タイムスタンプ書込みステート

101 : 予約

110 : Suspended - 送信バッファアンダフロー、または送信ディスクリプタが利用不能

111 : Running - 送信ディスクリプタの終了中

[bit19:17] RS (Receive Process State)
これらのビットは、受信 DMA の状態を示します。このフィールドは割込みを生成しません。

000: Stopped - Reset または Stop Receive コマンドが発行された

001: Running - 受信転送ディスクリプタ(Receive Transfer Descriptor)の取込み中

010: 予約

011: Running - 受信パケット待ち

100: Suspended - 受信ディスクリプタが利用不能

101: Running - 受信ディスクリプタの終了中

110: タイムスタンプ書込みステート

111: Running - 受信バッファからホストメモリへ受信パケットデータを転送中

[bit16] NIS (Normal Interrupt Summary)

Normal Interrupt Summary ビットの値は、下記のビットのうち DMA Register 7 で対応する割込みビットがイネーブルにされているビットの論理 OR です。

DMA Register 5[0]: Transmit Interrupt
DMA Register 5[2]: Transmit Buffer Unavailable
DMA Register 5[6]: Receive Interrupt
DMA Register 5[14]: Early Receive Interrupt

マスクされていないビットだけが Normal Interrupt Summary ビットに影響します。
これは sticky ビットであり、NIS がセットされる原因となった対応するビットをクリアするたびに、 (このビットに 1 を書き込んで) クリアしなければなりません。

[bit15] AIS (Abnormal Interrupt Summary)

Abnormal Interrupt Summary ビットの値は、下記のビットのうち DMA Register 7 で対応する割込みビットがイネーブルにされているビットの論理 OR です。

DMA Register 5[1]: Transmit Process Stopped
DMA Register 5[3]: Transmit Jabber Timeout
DMA Register 5[4]: Receive Overflow
DMA Register 5[5]: Transmit Underflow
DMA Register 5[7]: Receive Buffer Unavailable
DMA Register 5[8]: Receive Process Stopped
DMA Register 5[9]: Receive Watchdog Timeout
DMA Register 5[10]: Early Transmit Interrupt
DMA Register 5[13]: Fatal Bus Error

マスクされていないビットだけが Abnormal Interrupt Summary ビットに影響します。
これは、sticky ビットであり、AIS がセットされる原因となった対応するビットをクリアするたびにクリアしなければなりません。

[bit14] ERI (Early Receive Interrupt)

このビットは、DMA がパケットの最初のデータバッファを充填したことを示します。Receive Interrupt DMA Register 5[6]は、このビットを自動的にクリアします。

[bit13] FBI (Fatal Bus Error Interrupt)

このビットは、EB[2:0]で詳しく記述するエラーが発生したことを示します。このビットがセットされているとき、対応する DMA エンジンは、そのすべてのバスアクセスをディセーブルします。

[bit10] ETI (Early Transmit Interrupt)

このビットは、送信するフレームを完全に MTL Transmit FIFO に転送したことを示します。

[bit9] RWT (Receive Watchdog Timeout)

このビットは、2048 バイト(Jumbo Frame モードがイネーブルされているときは 10240 バイト)を超える長さのフレームが受信されたときにアサートされます。

[bit8] RPS (Receive process Stopped)

このビットは、受信プロセスが Stopped ステートに入ったときにアサートされます。

[bit7] RU (Receive Buffer Unavailable)
このビットは、受信リスト内の次のディスクリプタがホストによって所有されていて、DMA がそれを取得できないことを示します。受信プロセスは Suspend されます。受信ディスクリプタの処理を再開するには、ホストがディスクリプタの所有権を変更し、Receive Poll Demand コマンドを発行しなければなりません。Receive Poll Demand コマンドが発行されない場合、受信フレームの転送処理は、次の受信フレームが認識されたときに再開します。DMA Register 5[7]は、前の受信ディスクリプタが DMA によって所有されている場合にのみセットされます。

[bit6] RI (Receive Interrupt)
このビットは、フレーム受信の完了を示します。個別のフレームのステータス情報はディスクリプタに書き込まれます。受信は Running ステートのままです。

[bit5] UNF (Transmit underflow)
このビットは、フレーム送信中に送信バッファがアンダフローになったことを示します。送信は中断され、Underflow Error TDES0[1]がセットされます。

[bit4] OVF (Receive Overflow)
このビットは、フレーム受信中に受信バッファがオーバフローになったことを示します。部分的なフレームがアプリケーションに転送された場合、オーバフローステータスが RDES0[11]にセットされます。

[bit3] TJT (Transmit Jabber Timeout)
このビットは、Transmit Jabber Timer が時間切れになったことを示します。これはトランスミッタが過度にアクティブであったことを示します。送信プロセスがアボートされ、Stopped ステートになります。それによって Transmit Jabber Timeout TDES0[14]フラグがアサートされます。

[bit2] TU (Transmit Buffer Unavailable)
このビットは、送信リスト内の次のディスクリプタがホストによって所有されていて、DMA がそれを取得できないことを示します。送信が中断されます。bit[22:20]が送信プロセスの状態遷移を示します。送信ディスクリプタの処理を再開するには、ホストがディスクリプタの所有権を変更し、Transmit Poll Demand コマンドを発行しなければなりません。

[bit1] TPS (Transmit Process Stopped)
このビットは、送信が停止されたときにセットされます。

[bit0] TI (Transmit Interrupt)
このビットは、フレーム送信が完了し、最初のディスクリプタの TDES1[31]がセットされたことを示します。

4.39 DMA Register 6 (OMR)

Operation Mode レジスタは、送受信オペレーティングモードおよびコマンドを指定します。このレジスタは、DMA 初期化時の最後に書き込みます。

OMR (Operation Mode Register) Address 1018h

bit	31	30	29	28	27	26	25	24
Field	予約					DT	RSF	DFF
属性	R	R	R	R	R	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約		TSF	FTF	予約			TTC[2]
属性	R	R	R/W	R/WS_SC	R	R	R	R/W
初期値	-	-	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	TTC[1:0]		ST	予約				
属性	R/W	R/W	R/W	R	R	R	R	R
初期値	0	0	0	-	-	-	-	-

bit	7	6	5	4	3	2	1	0
Field	FEF	FUF	予約	RTC[1:0]		OSF	SR	予約
属性	R/W	R/W	R	R/W	R/W	R/W	R/W	R
初期値	0	0	0	0	0	0	0	0

[bit26] DT (Disable Dropping of TCP/IP Checksum Error Frames)

このビットをセットすると、GMAC は、Receive Checksum Offload エンジンによるエラー検出のみのフレームを廃棄しません。このようなフレームは、GMAC が受信した Ethernet フレーム内にはいかなるエラーも含まず(FCS エラーを含めて)、カプセル化されたペイロードにのみエラーが含まれています。このビットをリセットすると、FEF ビットがリセットされている場合にすべてのエラーフレームが廃棄されます。

[bit25] RSF (Receive Store and Forward)

このビットをセットすると、MTL 受信 FIFO にすべての受信フレームデータが書き込まれた後に、ホストメモリへの転送を開始し、RTC ビットを無視します。このビットをリセットすると、受信 FIFO は Cut-Through モードで動作し、RTC ビットによって指定されたしきい値に従って、ホストメモリへの転送を開始します。

[bit24] DFF (Disable Flushing of Received Frames)

このビットをセットすると、受信 DMA は、受信ディスクリプタ/バッファが利用不能であるために、通常(このビットがリセットされているとき)のようなフレームのフラッシュを行いません。

[bit21] TSF (Transmit Store Forward)

このビットをセットすると、MTL 送信 FIFO にすべての送信フレームが転送された後に、PHY インタフェースへの送信を開始します。このビットがセットされているとき、DMA Register 6[16:14]で指定された TTC 値は無視されます。このビットは送信が停止しているときにのみ変更できます。

[bit20] FTF (Flush Transmit FIFO)

このビットをセットすると、送信 FIFO コントローラロジックがそのデフォルト値にリセットされ、それによって送信 FIFO 内のすべてのデータが失われます(Flush されます)。このビットは、Flush 動作が完全に完了したときに内部的にクリアされます。このビットがクリアされるまで Operation Mode レジスタへの書込みを行ってはいけません。GMAC トランスミッタによって既に受領されたデータは Flush されません。この送信スケジュールにより、アンダフローが起こり、runt フレームが送信されます。

＜注意事項＞

- *フラッシュ動作は、送信 FIFO の内容が空になり、送信フレームのすべての保留中の送信ステータスがホストによって受領された後にのみ完了します。このフラッシュ動作を完了するためには、PHY 送信クロック (TX_CLK)がアクティブになっている必要があります。*

[bit16:14] TTC[2:0] (Transmit Threshold Control)

この 3 ビットは、MTL 送信 FIFO のしきい値レベルをコントロールします。MTL 送信 FIFO 内のフレームのサイズがしきい値よりも大きいとき、PHY インタフェースへの送信が開始されます。全体の長さがしきい値より小さいフレームは送信されます。これらのビットは、TSF ビット(bit 21)がリセットされているときにのみ使用されます。

000 :64 バイト
001 :128 バイト
010 :192 バイト
011 :256 バイト
100 :40 バイト
101 :32 バイト
110 :24 バイト
111 :16 バイト

[bit13] ST (Start/Stop Transmission Command)

このビットをセットすると、送信 DMA は Running ステートに入り、送信リストの現在の位置で、送信するフレームをチェックします。リスト内の現在の位置(DMA Register 4 によって設定された送信リストベースアドレス)から、または前に送信が停止したときに保持された位置からディスクリプタの取得が試みられます。現在のディスクリプタが DMA によって所有されていない場合、送信は Suspended ステートに入り、Transmit Buffer Unavailable (DMA Register 5[2])がセットされます。Start Transmission コマンドは送信が停止しているときのみ有効です。DMA Register 4 がセットされる前にこのコマンドが発行された場合の DMA の動作は予測不能です。
このビットをリセットすると、現在のフレームの送信完了後、送信プロセスは Stopped ステートに入ります。送信リスト内の次のディスクリプタの位置が保存され、送信再開時に現在の位置になります。送信停止コマンドは、現在のフレームの送信が完了したか、または送信が Suspended ステートにあるときにのみ有効です。

[bit7] FEF (Forward Error Frames)

このビットをリセットすると、受信 FIFO はエラーステータス(CRC エラー, コリジョンエラー, GMII_ER, Giant フレーム, ウォッチドッグタイムアウト, オーバフロー)が発生しているフレームを廃棄します。しかし、フレームの開始バイト(ライト)ポインタが(Threshold モードで)既にリードコントローラ側に転送されている場合、そのフレームは廃棄されません。
FEF をセットすると、Runt エラーフレームを除くすべてのフレームが DMA によって転送されます。しかし、FEF がセットされている場合でも、フレームの一部分の受信により受信 FIFO がオーバフローしたときは、そのフレームは廃棄されます。

[bit6] FUF (Forward Undersized Good Frames)

このビットをセットすると、受信 FIFO はサイズが小さいフレーム(エラーを含まず、パッドバイトと CRC を含む長さが 64 バイト未満のフレーム)を転送します。このビットをリセットすると、64 バイト未満のフレームは、Receive Threshold の値が小さいために既に転送済みである場合を除き、すべて廃棄されます。

[bit4, 3] RTC (Receive Threshold Control)

この 2 ビットは、MTL 受信 FIFO のしきい値レベルをコントロールします。DMA への転送(リクエスト)は、MTL 受信 FIFO 内のフレームのサイズがしきい値よりも大きいときに開始します。また、全体のサイズがしきい値より小さいフレームは自動的に転送されます。このビットは RSF ビットが 0 のときのみ有効で、RSF ビットが 1 にセットされているときは無視されます。

00 : 64
01 : 32
10 : 96
11 : 128

[bit2] OSF (Operate on Second Frame)

このビットをセットすると、最初のフレームの送信ステータスがまだ取得されていない場合でも、送信データの 2 番目のフレームの処理を開始するよう DMA に指示します。

* このビットがセットされているときに送信動作を停止するには、次のフロー1 または 2 を実行します。

1. 送信 DMA をディセーブルし、前のフレーム転送が完了する(送信 FIFO が空になる)のを待ちます。
   (この指定は OSF モードに関係なく適用されます)。
2. GMAC Configuration レジスタの当該ビットをクリアし、GMAC トランスミッタをディセーブルします。
   (このほかに FIFO が空かどうかを知る方法はありません。そのため STOP ステータスを待ってからトランスミッタをディセーブルすることができます)。

[bit1] SR (Start/Stop Receive)

このビットをセットすると、DMA 受信プロセスは Running ステートに入ります。受信リストからディスクリプタを取得し、受信フレームの処理を試みます。リスト内の現在の位置(DMA Register 3 によって設定されたアドレス)から、または前に受信プロセスが停止したときに保持されている位置からディスクリプタの取得が試みられます。DMA がディスクリプタを所有していない場合、受信は中断され、Receive Buffer Unavailable (DMA Register 5[7])がセットされます。Start Receive コマンドは、受信が停止しているときのみ有効です。DMA Register 3 をセットする前にこのコマンドが発行された場合の DMA の動作は予測不能です。

このビットがクリアされると、受信 DMA 動作は、現在のフレームの転送後に停止されます。送信リスト内の次のディスクリプタの位置が保存され、受信プロセス再開後に現在の位置になります。Stop Receive コマンドは、受信プロセスが Running (受信パケットを待機)ステートまたは Suspended ステートの場合のみ有効です。

4.40 DMA Register 7 (IER)

このレジスタは、DMA からの割込みを許可します。

IER (Interrupt Enable Register) Address 101Ch

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							NIE
属性	R	R	R	R	R	R	R	R/W
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	AIE	ERE	FBE	予約		ETE	RWE	RSE
属性	R/W	R/W	R/W	R	R	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	RUE	RIE	UNE	OVE	TJE	TUE	TSE	TIE
属性	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit16] NIE (Normal Interrupt Summary Enable)

このビットをセットすると、正常動作系割込みがイネーブルされます。このビットがリセットされると、正常動作系割込みがディセーブルされます。このビットは以下のビットをイネーブルにします。

DMA Register 5[0]: Transmit Interrupt
DMA Register 5[2]: Transmit Buffer Unavailable
DMA Register 5[6]: Receive Interrupt
DMA Register 5[14]: Early Receive Interrupt

[bit15] AIE (Abnormal Interrupt Summary Enable)

このビットをセットすると、異常動作系割込みがイネーブルされます。このビットがリセットされると、異常動作系割込みがディセーブルされます。このビットは以下のビットをイネーブルにします。

DMA Register 5[1]: Transmit Process Stopped
DMA Register 5[3]: Transmit Jabber Timeout
DMA Register 5[4]: Receive Overflow
DMA Register 5[5]: Transmit Underflow
DMA Register 5[7]: Receive Buffer Unavailable
DMA Register 5[8]: Receive Process Stopped
DMA Register 5[9]: Receive Watchdog Timeout
DMA Register 5[10]: Early Transmit Interrupt
DMA Register 5[13]: Fatal Bus Error

[bit14] ERE (Early Receive Interrupt Enable)

このビットを Normal Interrupt Summary Enable (DMA Register 7[16])と共にセットすると、Early Receive Interrupt がイネーブルされます。このビットをリセットすると、Early Receive Interrupt がディセーブルされます。

[bit13] FBE (Fatal Bus Error Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Fatal Bus Error Interrupt がイネーブルされます。このビットをリセットすると、Fatal Bus Error Enable Interrupt がディセーブルされます。

[bit10] ETE (Early Transmit Interrupt Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Early Transmit Interrupt がイネーブルされます。このビットをリセットすると、Early Transmit Interrupt がディセーブルされます。

[bit9] RWE (Receive Watchdog Timeout Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Receive Watchdog Timeout Interrupt がイネーブルされます。このビットをリセットすると、Receive Watchdog Timeout Interrupt がディセーブルされます。

[bit8] RSE (Receive Process Stopped Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Receive Stopped Interrupt がイネーブルされます。このビットをリセットすると、Receive Stopped Interrupt がディセーブルされます。

[bit7] RUE (Receive Buffer Unavailable Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Receive Buffer Unavailable Interrupt がイネーブルされます。このビットをリセットすると、Receive Buffer Unavailable Interrupt がディセーブルされます。

[bit6] RIE (Receive Interrupt Enable)
このビットを Normal Interrupt Summary Enable (DMA Register 7[16])と共にセットすると、Receive Interrupt がイネーブルされます。このビットをリセットすると、Receive Interrupt がディセーブルされます。

[bit5] UNE (Transmit underflow Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Transmit Underflow Interrupt がイネーブルされます。このビットをリセットすると、Underflow Interrupt がディセーブルされます。

[bit4] OVE (Receive Overflow Enable)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Receive Overflow Interrupt がイネーブルされます。このビットをリセットすると、Overflow Interrupt がディセーブルされます。

[bit3] TJE (Transmit Jabber Timeout)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Transmit Jabber Timeout Interrupt がイネーブルされます。このビットをリセットすると、Transmit Jabber Timeout Interrupt がディセーブルされます。

[bit2] TUE (Transmit Buffer Unavailable)
このビットを Normal Interrupt Summary Enable (DMA Register 7[16])と共にセットすると、Transmit Buffer Unavailable Interrupt がイネーブルされます。このビットをリセットすると、Transmit Buffer Unavailable Interrupt がディセーブルされます。

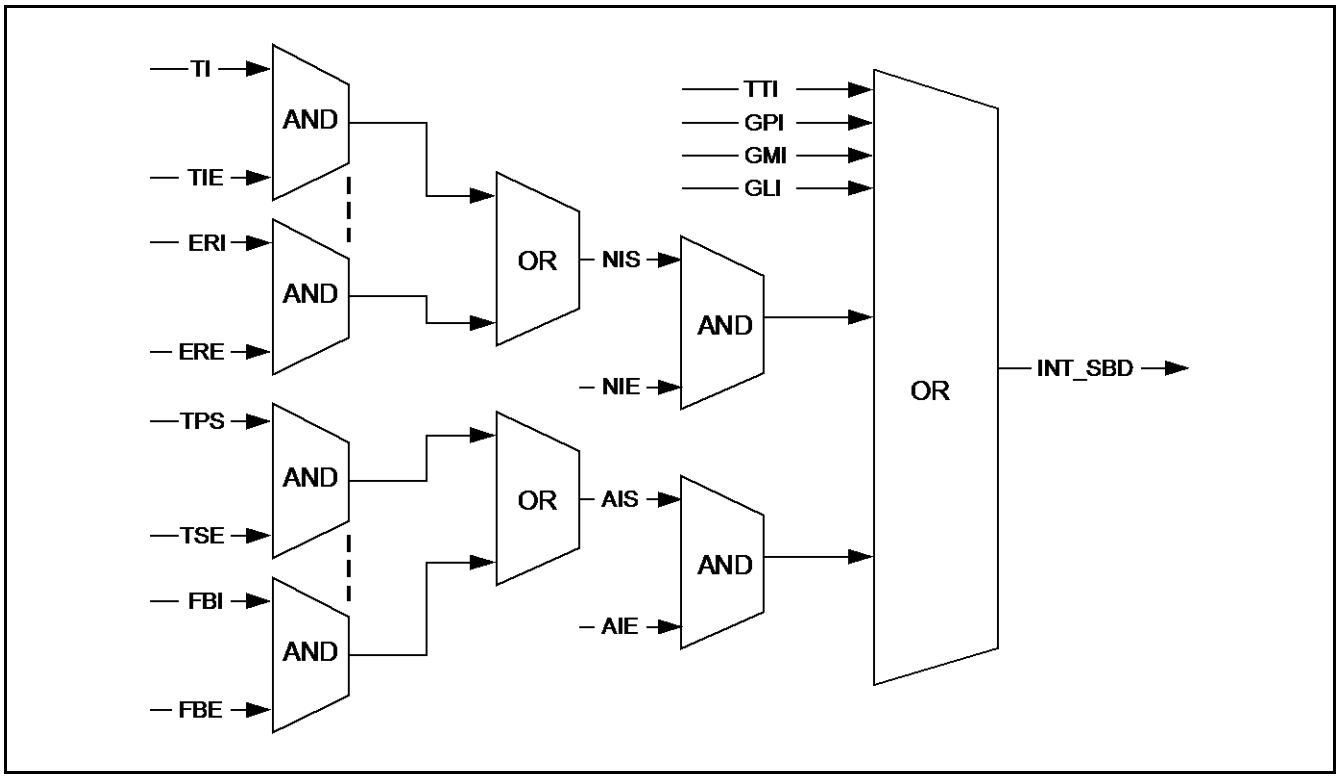
[bit1] TSE (Transmit Process Stopped)
このビットを Abnormal Interrupt Summary Enable (DMA Register 7[15])と共にセットすると、Transmission Stopped Interrupt がイネーブルされます。このビットをリセットすると、Transmission Stopped Interrupt がディセーブルされます。

[bit0] TIE (Transmit Interrupt)

このビットを Normal Interrupt Summary Enable (DMA Register 7[16])と共にセットすると、Transmit Interrupt がイネーブルされます。このビットをリセットすると、Transmit Interrupt がディセーブルされます。

Figure 4-2 に示すように、INT_SBD 割込みが生成されます。この割込みは、DMA ステータスレジスタの TTI, GPI, GMI, または GLI ビットがアサートされたとき、または NIS/AIS ステータスビットがアサートされ、対応する Interrupt Enable ビット (NIE/AIE)がイネーブルされたときのみアサートされます。

Figure 4-2 INT_SBD の生成



4.41 DMA Register 8 (MFBOCR)

DMA には、受信中に失われたフレームの数を追跡する 2 つのカウンタがあります。このレジスタはカウンタの現在の値を報告します。このカウンタは診断のために使用されます。

MFBOCR (Missed Frame and Buffer Overflow Counter Register) Address 1020h

bit	31	30	29	28	27	26	25	24
Field	予約			ONMFF	NMFF[10:7]			
属性	R	R	R	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	NMFF[6:0]							ONMFH
属性	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	NMFH[15:8]							
属性	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	NMFH[7:0]							
属性	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC	R_SS_RC
初期値	0	0	0	0	0	0	0	0

bit[15:0]は、ホストバッファが有効でないため失われたフレームの数を示します。bit[27:17]は、オーバフロー条件により失われたフレームおよび MTL によって廃棄された runt フレーム(64 バイト未満の有効なフレーム)の数を示します。

[bit28] ONMFF (Overflow NMFF)

このビットは、NMFF ビットがオーバフローしたことを示します。

[bit27:17] NMFF(Number of Missed frame by Ethernet-MAC)

このビットは、アプリケーションで失われたフレームの数を示します。このレジスタがリードされるとカウンタはクリアされます。

[bit16] ONMFH (Overflow NMFH)

このビットは、NMFH ビットがオーバフローしたことを示します。

[bit15:0] NMFH[15:0] (Number of Missed frame by HOST)

このビットは、Host Receive Buffer が有効でないため、コントローラで失われたフレームの数を示します。このレジスタがリードされるとカウンタはクリアされます。

4.42 DMA Register 9 (RIWTR)

このレジスタに 0 以外の値が書き込まれたとき、RI (DMA Register 5[6])の Watchdog Timer がイネーブルされます。

RIWTR (Receive Interrupt Watchdog Timer Register) Address 1024h

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	RIWT[7:0]							
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初期値	0	0	0	0	0	0	0	0

[bit7:0] RIWT (RI Watchdog Timer count)

このビットは、Watchdog Timer 設定値としてシステムサイクルを 256 倍した数値を設定します。Watchdog Timer は、フレームを受信 DMA が転送完了してからプログラムされた時間が経過したときにトリガされます。 (対応する受信ディスクリプタ RDES1[31]の設定により、RI ステータスビットがセットされない場合) Watchdog Timer が期限切れになると、RI ビットがセットされ、タイマは停止します。受信フレームの RDES1[31]の設定により、RI ステータスビットが 1 にセットされたときは、Watchdog Timer はリセットされます。

4.43 DMA Register 11 (AHBSR)

このレジスタは、AHB マスタ・インタフェースの状態を表示します。

AHBSR (AHB Status Register) Address 102Ch

bit	31	30	29	28	27	26	25	24
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	予約							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	予約							AHBS
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit0] AHBS(AHB Status)

このビットが 1 のとき、AHB マスタ・インタフェースがアイドル以外の状態であることを示します。

4.44 DMA Register 18 (CHTDR)

Current Host Transmit Descriptor レジスタは、DMA によって読み込まれた現在の送信ディスクリプタの開始アドレスを示します。

CHTDR (Current Host Transmit Descriptor Register) Address 1048h

bit	31	30	29	28	27	26	25	24
Field	HTDAP[31:24]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	HTDAP [23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	HTDAP [15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	HTDAP [7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit31:0] HTDAP[31:0] (Host Transmit Descriptor Address Pointer)

このビットはリセット時にクリアされます。ポインタは動作中に DMA によって更新されます。

4.45 DMA Register 19 (CHRDR)

Current Host Receive Descriptor レジスタは、DMA によって読み込まれた現在の受信ディスクリプタの開始アドレスを示します。

CHRDR (Current Host Receive Descriptor Register) Address 104Ch

bit	31	30	29	28	27	26	25	24
Field	HRDAP[31:24]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	HRDAP [23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	HRDAP [15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	HRDAP [7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit31:0] HRDAP[31:0] (Host Receive Descriptor Address Pointer)

このビットは、リセット時にクリアされます。ポインタは動作中に DMA によって更新されます。

4.46 DMA Register 20 (CHTBAR)

Current Host Transmit Buffer Address レジスタは、DMA によって読み込まれる現在の送信バッファアドレスを示します。

CHTBAR (Current Host Transmit Buffer Address Register) Address 1050h

bit	31	30	29	28	27	26	25	24
Field	HTBAR[31:24]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	HTBAR[23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	HTBAR[15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	HTBAR[7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit31:0] HTBAR[31:0] (Host Transmit Buffer Address Register)

このビットは、リセット時にクリアされます。ポインタは、動作中に DMA によって更新されます。

4.47 DMA Register 21 (CHRBAR)

Host Receive Buffer Address レジスタは、DMA によって読み込まれる現在の受信バッファアドレスを示します。

CHRBAR (Current Host Receive Buffer Address Register) Address 1054h

bit	31	30	29	28	27	26	25	24
Field	HRBAR[31:24]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	HRBAR[23:16]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	HRBAR[15:8]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	HRBAR[7:0]							
属性	R	R	R	R	R	R	R	R
初期値	0	0	0	0	0	0	0	0

[bit31:0] HRBAR[31:0] (Host Receive Buffer Address Register)

このビットは、リセット時にクリアされます。ポインタは、動作中に DMA によって更新されます。

4.48 MMC Register List

Table 4-5 に、MMC モジュールのレジスタマップと、各カウンタの統計数値の内容を記載します。

Table 4-5 MMC レジスタマップ

アドレス	レジスタ番号	レジスタ名	説明
0100	GMAC Reg. 64	mmc_cntl	MMC カウンタのオペレーションモード設定。 (4.49 参照)
0104	GMAC Reg. 65	mmc_intr_rx	すべての受信統計カウンタからの割込み。 (4.50 参照)
0108	GMAC Reg. 66	mmc_intr_tx	すべての送信統計カウンタからの割込み。 (4.51 参照)
010C	GMAC Reg. 67	mmc_intr_mask_rx	すべての受信統計カウンタからの割込みマスク (4.52 参照)
0110	GMAC Reg. 68	mmc_intr_mask_tx	すべての送信統計カウンタからの割込みマスク (4.53 参照)
0114	GMAC Reg. 69	txoctetcount_gb	送信した正常および不良なフレームのバイト数。 プリアンブル, およびリトライしたバイトを除く。
0118	GMAC Reg. 70	txframecount_gb	送信した正常および不良なフレーム数。 リトライしたフレームは除く。
011C	GMAC Reg. 71	txbroadcastframes_g	送信した正常なブロードキャストフレーム数。
0120	GMAC Reg. 72	txmulticastframes_g	送信した正常なマルチキャストフレーム数。
0124	GMAC Reg. 73	tx64octets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 64 バイトのフレーム数(リトライしたフレームは除く)。
0128	GMAC Reg. 74	tx65to127octets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 65～127 バイトのフレーム数(リトライしたフレームは除く)。
012C	GMAC Reg. 75	tx128to255octets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 128～255 バイトのフレーム数(リトライしたフレームは除く)。
0130	GMAC Reg. 76	tx256to511octets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 256～511 バイトのフレーム数(リトライしたフレームは除く)。
0134	GMAC Reg. 77	tx512to1023octets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 512～1023 バイトのフレーム数(リトライしたフレームは除く)。
0138	GMAC Reg. 78	tx1024tomaxoctets_gb	送信した正常および不良なフレームで、プリアンブルを除いた送信バイト数が 1024～Maxsize バイトのフレーム数(リトライしたフレームは除く)。
013C	GMAC Reg. 79	txunicastframes_gb	送信した正常および不良なユニキャストフレーム数。
0140	GMAC Reg. 80	txmulticastframes_gb	送信した正常および不良なマルチキャストフレーム数。
0144	GMAC Reg. 81	txbroadcastframes_gb	送信した正常および不良なブロードキャストフレーム数。
0148	GMAC Reg. 82	txunderflowerror	フレームアンダフローエラーによりアボートされたフレーム数。
014C	GMAC Reg. 83	txsinglecol_g	半二重モードで、1 回のコリジョンの後に正常送信したフレーム数。
0150	GMAC Reg. 84	txmulticol_g	半二重モードで、2 回以上のコリジョンの後に正常送信したフレーム数。
0154	GMAC Reg. 85	txdeferred	半二重モードで、一定時間保留後に正常送信したフレーム数。
0158	GMAC Reg. 86	txlatecol	レイトコリジョンエラーによりアボートされたフレーム数。
015C	GMAC Reg. 87	txexesscol	過剰コリジョン(16 回)のエラーによりアボートされたフレーム数。
0160	GMAC Reg. 88	txcarriererrror	キャリアセンスエラー(キャリアなしまたはキャリアロスト)によりアボートされたフレーム数。
0164	GMAC Reg. 89	txoctetcount_g	プリアンブルを除いた正常送信バイト数。
0168	GMAC Reg. 90	txframecount_g	正常に送信したフレーム数。
016C	GMAC Reg. 91	txexecessdef	過剰 Deferral Error (最大フレームの 2 倍の時間以上遅延した場合)によりアボートされたフレーム数。
0170	GMAC Reg. 92	txpauseframes	正常に送信した PAUSE フレーム数
0174	GMAC Reg. 93	txvlanframes_g	正常に送信した VLAN フレーム数(リトライしたフレームは除く)。

アドレス	レジスタ番号	レジスタ名	説明
0178, 017C	予約	予約	-
0180	GMAC Reg. 96	rxframecount_gb	受信した正常および不良なフレーム数。
0184	GMAC Reg. 97	rxoctetcount_gb	正常および不良なフレームで、プリアンブルを除いた受信バイト数。
0188	GMAC Reg. 98	rxoctetcount_g	正常に受信した、プリアンブルを除いた受信バイト数。
018C	GMAC Reg. 99	rxbroadcastframes_g	正常に受信したブロードキャストフレーム数。
0190	GMAC Reg. 100	rxmulticastframes_g	正常に受信したマルチキャストフレーム数。
0194	GMAC Reg. 101	rxcrcerror	受信したフレームで CRC Error が発生したフレーム数。
0198	GMAC Reg. 102	rxallignmenterror	受信したフレームで Alignment (Dribble)Error が発生したフレーム数(10/100 モードのときのみ有効)。
019C	GMAC Reg. 103	rxrunterror	受信したフレームで Runt Error (64 バイト以下で CRC Error)が発生したフレーム数。
01A0	GMAC Reg. 104	rxjabbererror	受信したフレームでフレーム長が 1518 バイト以上で CRC Error が発生したフレーム数。
01A4	GMAC Reg. 105	rxundersize_g	エラーなしで正常に受信した、フレーム長が 64 バイト以下のフレーム数。
01A8	GMAC Reg. 106	rxoversize_g	エラーなしで正常に受信した、フレーム長が Maxsize 以上のフレーム数。
01AC	GMAC Reg. 107	rx64octets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 64 バイトのフレーム数。
01B0	GMAC Reg. 108	rx65to127octets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 65～127 バイトのフレーム数。
01B4	GMAC Reg. 109	rx128to255octets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 128～255 バイトのフレーム数。
01B8	GMAC Reg. 110	rx256to511octets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 256～511 バイトのフレーム数。
01BC	GMAC Reg. 111	rx512to1023octets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 512～1023 バイトのフレーム数。
01C0	GMAC Reg. 112	rx1024tomaxoctets_gb	受信した、正常および不良なフレームで、プリアンブルを除いた受信バイト数が 1024～Maxsize バイトのフレーム数。
01C4	GMAC Reg. 113	rxunicastframes_g	正常に受信したユニキャストフレーム数。
01C8	GMAC Reg. 114	rxlengtherror	受信した有効な Length Field のすべてのフレームで Length Error (Length type field = フレームサイズでない)が発生したフレーム数。
01CC	GMAC Reg. 115	rxoutofrangetype	受信したフレームで、有効なフレームサイズでない(1500 バイト以上)の Length/Type フィールドを持つフレーム数。
01D0	GMAC Reg. 116	rxpauseframes	正常に受信した、有効な PAUSE フレーム数。
01D4	GMAC Reg. 117	rxfifooverflow	FIFO オーバフローにより受信できなかったフレーム数。
01D8	GMAC Reg. 118	rxvlanframes_gb	受信した、正常および不良な VLAN フレーム数。
01DC	GMAC Reg. 119	rxwatchdogerror	受信したフレームで、Watchdog Timeout Error により (2048 バイト以上のデータ負荷を持つフレーム)エラーが発生したフレーム数。
01E0 | 01FC	予約	予約	-
0200	GMAC Reg. 128	mmc_ipc_intr_mask_rx	受信 IPC 統計カウンタからの割込みのマスク。 (4.54 参照)
0204	予約	予約	-
0208	GMAC Reg. 130	mmc_ipc_intr_rx	受信 IPC 統計カウンタからの割込み。 (4.55 参照)
020C	予約	予約	-

アドレス	レジスタ番号	レジスタ名	説明
0210	GMAC Reg. 132	rxipv4_gd_frms	TCP, UDP, または ICMP ペイロードで受信した正常な IPv4 データグラムの数。
0214	GMAC Reg. 133	rxipv4_hdrerr_frms	受信した、Header Error (チェックサム, 長さ, またバージョンの不一致)が発生した IPv4 データグラムの数。
0218	GMAC Reg. 134	rxipv4_nopay_frms	受信した、Checksum エンジンによって処理された TCP, UDP または ICMP ペイロードを持っていなかった IPv4 データグラムフレーム数。
021C	GMAC Reg. 135	rxipv4_frag_frms	フラグメンテーションされた正常な IPv4 データグラムの数。
0220	GMAC Reg. 136	rxipv4_udsbl_frms	受信した、チェックサムがディセーブルされている UDP ペイロードを持つ正常な IPv4 データグラムの数。
0224	GMAC Reg. 137	rxipv6_gd_frms	受信した、TCP, UDP, または ICMP ペイロードを持つ正常な IPv6 データグラムの数。
0228	GMAC Reg. 138	rxipv6_hdrerr_frms	受信した、Header Error (長さまたはバージョンの不一致)が発生した IPv6 データグラムの数。
022C	GMAC Reg. 139	rxipv6_nopay_frms	受信した、TCP, UDP または ICMP ペイロードを持っていなかった IPv6 データグラムフレーム数。これには、フラグメンテーションされたまたはセキュリティ拡張ヘッダを持つすべての IPv6 データグラムが含まれます。
0230	GMAC Reg. 140	rxudp_gd_frms	正常な UDP ペイロードを持つ正常な IP データグラムの数。このカウンタは、rxipv4_udsbl_frms カウンタが増分されたとき、更新されません。
0234	GMAC Reg. 141	rxudp_err_frms	UDP ペイロードで Checksum Error が発生した正常な IP データグラムの数。
0238	GMAC Reg. 142	rxtcp_gd_frms	正常な TCP ペイロードを持つ正常な IP データグラムの数。
023C	GMAC Reg. 143	rxtcp_err_frms	TCP ペイロードで Checksum Error が発生した正常な IP データグラムの数。
0240	GMAC Reg. 144	rxicmp_gd_frms	正常な ICMP ペイロードを持つ正常な IP データグラムの数。
0244	GMAC Reg. 145	rxicmp_err_frms	ICMP ペイロードで Checksum Error が発生した正常な IP データグラムの数。
0248, 024C	予約	予約	-
0250	GMAC Reg. 148	rxipv4_gd_octets	TCP, UDP または ICMP データをカプセル化する正常な IPv4 で受信したバイト数(Ethernet ヘッダ, FCS, パッドまたは IP パッドバイトはこのカウンタまたは下記にリストしているオクテットカウンタには含まれていません)。
0254	GMAC Reg. 149	rxipv4_hdrerr_octets	受信した Header Error (チェックサム, 長さ, バージョンの不一致)が発生した IPv4 データグラムのバイト数。IPv4 ヘッダの Length field の値は、このカウンタを更新するために使用されます。
0258	GMAC Reg. 150	rxipv4_nopay_octets	受信した、TCP, UDP または ICMP ペイロードを持っていなかった IPv4 データグラムのバイト数。IPv4 ヘッダの Length field の値は、このカウンタを更新するために使用されます。
025C	GMAC Reg. 151	rxipv4_frag_octets	受信した、フラグメント化された IPv4 データグラムのバイト数。IPv4 ヘッダの Length field の値は、このカウンタを更新するために使用されます。
0260	GMAC Reg. 152	rxipv4_udsbl_octets	受信した、ディセーブルした UDP チェックサムを持つ UDP セグメントのバイト数。このカウンタは IP Header バイトをカウントしません。
0264	GMAC Reg. 153	rxipv6_gd_octets	受信した、TCP, UDP または ICMPv6 データをカプセル化する正常な IPv6 データグラムのバイト数。
0268	GMAC Reg. 154	rxipv6_hdrerr_octets	受信した、Header Error (長さ, バージョンの不一致)が発生した IPv6 データグラムのバイト数。IPv6 ヘッダの Length field の値は、このカウンタを更新するために使用されます。
026C	GMAC Reg. 155	rxipv6_nopay_octets	受信した、TCP, UDP または ICMP ペイロードを持っていなかった IPv6 データグラムのバイト数。IPv6 ヘッダの Length field の値は、このカウンタを更新するために使用されます。

アドレス	レジスタ番号	レジスタ名	説明
0270	GMAC Reg. 156	rxudp_gd_octets	受信した、正常な UDP セグメントのバイト数。このカウンタ(および以下のカウンタ)は IP ヘッダバイトをカウントしません。
0274	GMAC Reg. 157	rxudp_err_octets	受信した、Checksum Error が発生した UDP セグメントのバイト数。
0278	GMAC Reg. 158	rxtcp_gd_octets	受信した、正常な TCP セグメントのバイト数。
027C	GMAC Reg. 159	rxtcp_err_octets	受信した、Checksum Error が発生した TCP セグメントのバイト数。
0280	GMAC Reg. 160	rxicmp_gd_octets	受信した、正常な ICMP セグメントのバイト数。
0284	GMAC Reg. 161	rxicmp_err_octets	受信した、Checksum Error が発生した ICMP セグメントのバイト数。
0288 | 02FC	予約	予約	-

<注意事項>

- *以降の節で説明している MMC コントロールレジスタ, MMC 受信割込みレジスタ, MMC 送信割込みレジスタ, MMC 受信チェックサムオフロード割込みレジスタ, MMC 受信割込みマスクレジスタ, MMC 送信割込みマスクレジスタ, MMC 受信チェックサムオフロード割込みマスクレジスタ以外の以下のアドレスの MMC カウンタレジスタは、読出し専用で、初期値は 0 です。*

・0x0114 to 0x0174
・0x0180 to 0x01DC
・0x0210 to 0x0244
・0x0250 to 0x0284

4.49 GMAC Register.64 (mmc_cntl)

MMC コントロールレジスタは、マネジメントカウンタのオペレーションモードを設定します。

MMC コントロールレジスタ(mmc_cntl) Address:0100h

bit	属性	初期値	説明
31:6	R	0	予約
5	R/W	0	Full-Half preset このビットを 0、bit4 をセットすると、すべての MMC カウンタはほぼ half 値にプリセットされます。すべてのオクテットカウンタは 0x7FFF_F800 (half-2KBytes)にプリセットされます。すべてのフレームカウンタは 0x7FFF_FFF0 (half-16)にプリセットされます。 このビットを 1、bit4 をセットすると、すべての MMC カウンタはほぼ full 値にリセットされます。すべてのオクテットカウンタは 0xFFFF_F800 (full-2KBytes)にプリセットされます。 すべてのフレームカウンタは 0xFFFF_FFF0 (full -16)にプリセットされます。
4	R/W_SC	0	Counters Preset このビットをセットすると、すべてのカウンタが初期化されるか、上記 bit5 に従ってほぼ full またはほぼ half にプリセットされます。1 クロックサイクル後、このビットは自動的にクリアされます。bit5 とこのビットにより、MMC カウンタが half-full または full になるため、割込みのアサートのデバッグおよびテストに使用できます。
3	R/W	0	MMC Counter Freeze このビットを 1 にセットしているとき、すべての MMC カウンタが現在の値にフリーズされます(このビットが 0 にリセットされるまで、送信または受信されたフレームによってどの MMC カウンタも更新されません。bit2 の Reset on Read がセットされていて MMC カウンタが読み込まれた場合、このモードであっても、カウンタはクリアされます。
2	R/W	0	Reset on read このビットをセットすると、MMC カウンタはリードの後、0 にリセットされます(リセット後 self-clearing)。最下位バイトレーン(bit [7:0])が読み込まれたとき、これらのカウンタはクリアされます。
1	R/W	0	Counter Stop Rollover このビットをセットしているとき、カウンタは最大値に到達した後、0 に戻りません。
0	R/W_SC	0	Counter reset このビットをセットすると、すべてのカウンタがリセットされます。このビットは、1 クロックサイクルの後、自動的にクリアされます。

4.50 GMAC Register.65 (mmc_intr_rx)

このレジスタは、各受信統計カウンタからの割込みを表示します。

MMC 受信割込みレジスタ(mmc_intr_rx) Address:0104h

bit	属性	初期値	説明
31:24	R	0	予約
23	R_SS_RC	0	rxwatchdogerror カウンタ値が規定値に到達したとき、1 にセットされます。
22	R_SS_RC	0	rxvlanframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
21	R_SS_RC	0	Rxfifooverflow_gb カウンタの値が規定値に到達したとき、1 にセットされます。
20	R_SS_RC	0	rxpauseframes カウンタの値が規定値に到達したとき、1 にセットされます。
19	R_SS_RC	0	rxoutofrangetype カウンタの値が規定値に到達したとき、1 にセットされます。
18	R_SS_RC	0	Rxlengtherror カウンタの値が規定値に到達したとき、1 にセットされます。
17	R_SS_RC	0	rxunicastframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
16	R_SS_RC	0	rx1024tomaxoctets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
15	R_SS_RC	0	rx512to1023octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
14	R_SS_RC	0	rx256to511octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
13	R_SS_RC	0	rx128to255maxoctects_gb カウンタの値が規定値に到達したとき、1 にセットされます。
12	R_SS_RC	0	rx65to127maxoctets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
11	R_SS_RC	0	rxto64octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
10	R_SS_RC	0	rxoversize_g カウンタの値が規定値に到達したとき、1 にセットされます。
9	R_SS_RC	0	rxundersize_g カウンタの値が規定値に到達したとき、1 にセットされます。
8	R_SS_RC	0	Rxjabbererror カウンタの値が規定値に到達したとき、1 にセットされます。
7	R_SS_RC	0	rxrunterror カウンタの値が規定値に到達したとき、1 にセットされます。
6	R_SS_RC	0	rxallignmenterror カウンタの値が規定値に到達したとき、1 にセットされます。
5	R_SS_RC	0	rxcrcerror カウンタの値が規定値に到達したとき、1 にセットされます。
4	R_SS_RC	0	rxmulticastframes_g カウンタの値が規定値に到達したとき、1 にセットされます。
3	R_SS_RC	0	rxbroadcastframes_g カウンタの値が規定値に到達したとき、1 にセットされます。
2	R_SS_RC	0	rxoctetcounter_g カウンタの値が規定値に到達したとき、1 にセットされます。
1	R_SS_RC	0	rxoctetcounter_gb カウンタの値が規定値に到達したとき、1 にセットされます。
0	R_SS_RC	0	Rxframecount_gb カウンタの値が規定値に到達したとき、1 にセットされます。

MMC 受信割込みレジスタの各ビットは､各受信統計カウンタの値が最大値の半分に到達したとき(32 ビットカウンタの場合 0x8000_0000, および 16 ビットカウンタの場合 0x8000)、および最大値に到達したとき (32 ビットカウンタの場合 0xFFFF_FFFF, および 16 ビットカウンタの場合 0xFFFF)に生成された割込みを保持します｡Counter Stop Rollover がセットされると、割込みがセットされますが、カウンタはすべて 1 のままです。MMC 受信割込みレジスタは、32 ビット幅レジスタです。割込みを生成する各 MMC カウンタが読み込まれたとき、割込みビットがクリアされます。割込みビットをクリアするためには、各カウンタの最下位バイトレーン(bit[7:0])を読み込まなければなりません。

＜注意事項＞

- ***これらのレジスタは内部的にセットされ、Counter レジスタが読み込まれたときクリアされます。***

4.51 GMAC Register.66 (mmc_intr_tx)

このレジスタは、各送信統計カウンタの割込みを表示します。

MMC 送信割込みレジスタ(mmc_intr_tx) Address:0108h

bit	属性	初期値	説明
31:25	R	0	予約
24	R_SS_RC	0	txvlanframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
23	R_SS_RC	0	txpauseframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
22	R_SS_RC	0	txoexcessdef カウンタの値が規定値に到達したとき、1 にセットされます。
21	R_SS_RC	0	txframecount カウンタの値が規定値に到達したとき、1 にセットされます。
20	R_SS_RC	0	txoctetcount カウンタの値が規定値に到達したとき、1 にセットされます。
19	R_SS_RC	0	txcarriererror カウンタの値が規定値に到達したとき、1 にセットされます。
18	R_SS_RC	0	txlexesscol カウンタの値が規定値に到達したとき、1 にセットされます。
17	R_SS_RC	0	txlatecol カウンタの値が規定値に到達したとき、1 にセットされます。
16	R_SS_RC	0	txdeffer カウンタの値が規定値に到達したとき、1 にセットされます。
15	R_SS_RC	0	txmulticol_g カウンタの値が規定値に到達したとき、1 にセットされます。
14	R_SS_RC	0	txsinglecol_g カウンタの値が規定値に到達したとき、1 にセットされます。
13	R_SS_RC	0	txunderflowerror カウンタの値が規定値に到達したとき、1 にセットされます。
12	R_SS_RC	0	txbroadcastframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
11	R_SS_RC	0	txmulticastframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
10	R_SS_RC	0	txunicastframes_gb カウンタの値が規定値に到達したとき、1 にセットされます。
9	R_SS_RC	0	tx1024tomaxoctets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
8	R_SS_RC	0	tx512to1023octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
7	R_SS_RC	0	tx256to511octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
6	R_SS_RC	0	tx128to255maxoctects_gb カウンタの値が規定値に到達したとき、1 にセットされます。
5	R_SS_RC	0	tx65to127maxoctets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
4	R_SS_RC	0	tx64octets_gb カウンタの値が規定値に到達したとき、1 にセットされます。
3	R_SS_RC	0	txmalticastframes_g カウンタの値が規定値に到達したとき、1 にセットされます。
2	R_SS_RC	0	txbroadcastframes_g カウンタの値が規定値に到達したとき、1 にセットされます。
1	R_SS_RC	0	txframecount_gb カウンタの値が規定値に到達したとき、1 にセットされます。
0	R_SS_RC	0	txoctetcount_gb カウンタの値が規定値に到達したとき、1 にセットされます。

MMC 送信割込みレジスタの各ビットは、各送信統計カウンタの値が最大値の半分に到達したとき(32 ビットカウンタの場合 0x8000_0000, 16 ビットカウンタの場合 0x8000)、および最大値に到達したとき(32 ビットカウンタの場合 0xFFFF_FFFF, 16 ビットのカウンタ場合 0xFFFF)に生成された割込みを保持します。Counter Stop Rollover がセットされると、割込みがセットされますが、カウンタはすべて 1 のままです。MMC 送信割込みレジスタは、32 ビット幅レジスタです。割込みを発生した各カウンタが読み込まれたとき、割込みビットがクリアされます。割込みビットをクリアするためには、各カウンタの最下位バイトレーン(bit [7:0])を読み込まなければなりません。

＜注意事項＞

- *これらのレジスタは内部的にセットされ、Counter レジスタが読み込まれたときクリアされます。*

4.52 GMAC Register.67 (mmc_intr_mask_rx)

このレジスタは、各受信統計カウンタからの割込みをマスクします。

MMC 受信割込みマスク割込みレジスタ(mmc_intr_mask_rx) Address:010Ch

bit	属性	初期値	説明
31:24	R	0	予約
23	R/W	0	1 にセットすると、rxwatchdogerror カウンタの割込みをマスクします。
22	R/W	0	1 にセットすると、rxvlanframes_gb カウンタの割込みをマスクします。
21	R/W	0	1 にセットすると、rxfifooverflow_gb カウンタの割込みをマスクします。
20	R/W	0	1 にセットすると、rxpauseframes カウンタの割込みをマスクします。
19	R/W	0	1 にセットすると、rxoutofrangetype カウンタの割込みをマスクします。
18	R/W	0	1 にセットすると、rxlengtherror カウンタの割込みをマスクします。
17	R/W	0	1 にセットすると、rxunicastframes_gb カウンタの割込みをマスクします。
16	R/W	0	1 にセットすると、rx1024tomaxoctets_gb カウンタの割込みをマスクします。
15	R/W	0	1 にセットすると、rx512to1023octets_gb カウンタの割込みをマスクします。
14	R/W	0	1 にセットすると、rx256to511octets_gb カウンタの割込みをマスクします。
13	R/W	0	1 にセットすると、rx128to255maxoctects_gb カウンタの割込みをマスクします。
12	R/W	0	1 にセットすると、rx65to127maxoctets_gb カウンタの割込みをマスクします。
11	R/W	0	1 にセットすると、rxto64octets_gb カウンタの割込みをマスクします。
10	R/W	0	1 にセットすると、rxoversize_g カウンタの割込みをマスクします。
9	R/W	0	1 にセットすると、rxundersize_g カウンタの割込みをマスクします。
8	R/W	0	1 にセットすると、rxjabbererror カウンタの割込みをマスクします。
7	R/W	0	1 にセットすると、rxrunterror カウンタの割込みをマスクします。
6	R/W	0	1 にセットすると、rxallignmenterror カウンタの割込みをマスクします。
5	R/W	0	1 にセットすると、rxcrcerror カウンタの割込みをマスクします。
4	R/W	0	1 にセットすると、rxmulticastframes_g カウンタの割込みをマスクします。
3	R/W	0	1 にセットすると、rxbroadcastframes_g カウンタの割込みをマスクします。
2	R/W	0	1 にセットすると、rxoctetcounter_g カウンタの割込みをマスクします。
1	R/W	0	1 にセットすると、rxoctetcounter_gb カウンタの割込みをマスクします。
0	R/W	0	1 にセットすると、rxframecount_gb カウンタの割込みをマスクします。

MMC 受信割込みマスクレジスタの各ビットは、各受信統計カウンタの値が最大値の半分に到達したとき、および最大値に到達したときに生成される割込みをマスクします。このレジスタは 32 ビット幅です。

4.53 GMAC Register.68 (mmc_intr_mask_tx)

このレジスタは、各送信統計カウンタからの割込みをマスクします。

MMC 送信割込みマスクレジスタ(mmc_intr_mask_tx)Address:0110h

bit	属性	初期値	説明
31:25	R	0	予約
24	R/W	0	1 にセットすると、txvlanframes_gb カウンタの割込みをマスクします。
23	R/W	0	1 にセットすると、txpauseframes_gb カウンタの割込みをマスクします。
22	R/W	0	1 にセットすると、txoexcessdef カウンタの割込みをマスクします。
21	R/W	0	1 にセットすると、txframecount カウンタの割込みをマスクします。
20	R/W	0	1 にセットすると、txoctetcount カウンタの割込みをマスクします。
19	R/W	0	1 にセットすると、txcarriererror カウンタの割込みをマスクします。
18	R/W	0	1 にセットすると、txlexesscol カウンタの割込みをマスクします。
17	R/W	0	1 にセットすると、txlatecol カウンタの割込みをマスクします。
16	R/W	0	1 にセットすると、txdeffer カウンタの割込みをマスクします。
15	R/W	0	1 にセットすると、txmulticol_g カウンタの割込みをマスクします。
14	R/W	0	1 にセットすると、txsinglecol_g カウンタの割込みをマスクします。
13	R/W	0	1 にセットすると、txunderflowerror カウンタの割込みをマスクします。
12	R/W	0	1 にセットすると、txbroadcastframes_gb カウンタの割込みをマスクします。
11	R/W	0	1 にセットすると、txmulticastframes_gb カウンタの割込みをマスクします。
10	R/W	0	1 にセットすると、txunicastframes_gb カウンタの割込みをマスクします。
9	R/W	0	1 にセットすると、tx1024tomaxoctets_gb カウンタの割込みをマスクします。
8	R/W	0	1 にセットすると、tx512to1023octets_gb カウンタの割込みをマスクします。
7	R/W	0	1 にセットすると、tx256to511octets_gb カウンタの割込みをマスクします。
6	R/W	0	1 にセットすると、tx128to255maxoctects_gb カウンタの割込みをマスクします。
5	R/W	0	1 にセットすると、tx65to127maxoctets_gb カウンタの割込みをマスクします。
4	R/W	0	1 にセットすると、txto64octets_gb カウンタの割込みをマスクします。
3	R/W	0	1 にセットすると、txmalticastframes_g カウンタの割込みをマスクします。
2	R/W	0	1 にセットすると、txbroadcastframes_g カウンタの割込みをマスクします。
1	R/W	0	1 にセットすると、txframecount_gb カウンタの割込みをマスクします。
0	R/W	0	1 にセットすると、txoctetcount_gb カウンタの割込みをマスクします。

MMC 送信割込みマスクレジスタの各ビットは、各送信統計カウンタの値が最大値の半分に到達したとき、および最大値に到達したときに生成される割込みをマスクします。このレジスタは 32 ビット幅です。

4.54 GMAC Register.128 (mmc_ipc_intr_mask_rx)

このレジスタは、各受信 IPC(チェックサムオフロード)統計カウンタからの割込みをマスクします。

MMC 受信チェックサムオフロード割込みマスクレジスタ(mmc_ipc_intr_mask_rx) Address:0200h

bit	属性	初期値	説明
31, 30	R	0	予約
29	R/W	0	1 にセットすると、rxicmp_err_octets カウンタの割込みをマスクします。
28	R/W	0	1 にセットすると、rxicmp_gd_octets カウンタの割込みをマスクします。
27	R/W	0	1 にセットすると、rxtcp_err_octets カウンタの割込みをマスクします。
26	R/W	0	1 にセットすると、rxtcp_gd_octets カウンタの割込みをマスクします。
25	R/W	0	1 にセットすると、rxudp_err_octets カウンタの割込みをマスクします。
24	R/W	0	1 にセットすると、rxudp_gd_octets カウンタの割込みをマスクします。
23	R/W	0	1 にセットすると、rxipv6_nopay_octets カウンタの割込みをマスクします。
22	R/W	0	1 にセットすると、rxipv6_hdrerr_octets カウンタの割込みをマスクします。
21	R/W	0	1 にセットすると、rxipv6_gd_octets カウンタの割込みをマスクします。
20	R/W	0	1 にセットすると、rxipv4_udsbl_octets カウンタの割込みをマスクします。
19	R/W	0	1 にセットすると、rxipv4_frag_octets カウンタの割込みをマスクします。
18	R/W	0	1 にセットすると、rxipv4_nopay_octets カウンタの割込みをマスクします。
17	R/W	0	1 にセットすると、rxipv4_hdrerr_octets カウンタの割込みをマスクします。
16	R/W	0	1 にセットすると、rxipv4_gd_octets カウンタの割込みをマスクします。
15, 14	R	0	予約
13	R/W	0	1 にセットすると、rxicmp_err_frms カウンタの割込みをマスクします。
12	R/W	0	1 にセットすると、rxicmp_gd_frms カウンタの割込みをマスクします。
11	R/W	0	1 にセットすると、rxtcp_err_frms カウンタの割込みをマスクします。
10	R/W	0	1 にセットすると、rxtcp_gd_frms カウンタの割込みをマスクします。
9	R/W	0	1 にセットすると、rxudp_err_frms カウンタの割込みをマスクします。
8	R/W	0	1 にセットすると、rxudp_gd_frms カウンタの割込みをマスクします。
7	R/W	0	1 にセットすると、rxipv6_nopay_frms カウンタの割込みをマスクします。
6	R/W	0	1 にセットすると、rxipv6_hdrerr_frms カウンタの割込みをマスクします。
5	R/W	0	1 にセットすると、rxipv6_gd_frms カウンタの割込みをマスクします。
4	R/W	0	1 にセットすると、rxipv4_udsbl_frms カウンタの割込みをマスクします。
3	R/W	0	1 にセットすると、rxipv4_frag_frms カウンタの割込みをマスクします。
2	R/W	0	1 にセットすると、rxipv4_nopay_frms カウンタの割込みをマスクします。
1	R/W	0	1 にセットすると、rxipv4_hdrerr_frms カウンタの割込みをマスクします。
0	R/W	0	1 にセットすると、rxipv4_gd_frms カウンタの割込みをマスクします。

MMC 受信チェックサムオフロード割込みマスクレジスタの各ビットは、各受信 IPC(チェックサムオフロード)統計カウンタの値が最大値の半分に到達したとき、および最大値に到達したときに生成される割込みをマスクします。このレジスタは、32 ビット幅です。

4.55 GMAC Register.130 (mmc_ipc_intr_rx)

このレジスタは、各受信 IPC(チェックサムオフロード)統計カウンタからの割込みを表示します。

MMC 受信チェックサムオフロード割込みレジスタ(mmc_ipc_intr_rx) Address:0208h

bit	属性	初期値	説明
31, 30	R	0	予約
29	R_SS_RC	0	rxicmp_err_octets カウンタの値が規定値に到達したとき、1 にセットされます。
28	R_SS_RC	0	rxicmp_gd_octets カウンタの値が規定値に到達したとき、1 にセットされます。
27	R_SS_RC	0	rxtcp_err_octets カウンタの値が規定値に到達したとき、1 にセットされます。
26	R_SS_RC	0	rxtcp_gd_octets カウンタの値が規定値に到達したとき、1 にセットされます。
25	R_SS_RC	0	rxudp_err_octets カウンタの値が規定値に到達したとき、1 にセットされます。
24	R_SS_RC	0	rxudp_gd_octets カウンタの値が規定値に到達したとき、1 にセットされます。
23	R_SS_RC	0	rxipv6_nopay_octets カウンタの値が規定値に到達したとき、1 にセットされます。
22	R_SS_RC	0	rxipv6_hdrerr_octets カウンタの値が規定値に到達したとき、1 にセットされます。
21	R_SS_RC	0	rxipv6_gd_octets カウンタの値が規定値に到達したとき、1 にセットされます。
20	R_SS_RC	0	rxipv4_udsbl_octets カウンタの値が規定値に到達したとき、1 にセットされます。
19	R_SS_RC	0	rxipv4_frag_octets カウンタの値が規定値に到達したとき、1 にセットされます。
18	R_SS_RC	0	rxipv4_nopay_octets カウンタの値が規定値に到達したとき、1 にセットされます。
17	R_SS_RC	0	rxipv4_hdrerr_octets カウンタの値が規定値に到達したとき、1 にセットされます。
16	R_SS_RC	0	rxipv4_gd_octets カウンタの値が規定値に到達したとき、1 にセットされます。
15, 14	R	0	予約
13	R_SS_RC	0	rxicmp_err_frms カウンタの値が規定値に到達したとき、1 にセットされます。
12	R_SS_RC	0	rxicmp_gd_frms カウンタの値が規定値に到達したとき、1 にセットされます。
11	R_SS_RC	0	rxtcp_err_frms カウンタの値が規定値に到達したとき、1 にセットされます。
10	R_SS_RC	0	rxtcp_gd_frms カウンタの値が規定値に到達したとき、1 にセットされます。
9	R_SS_RC	0	rxudp_err_frms カウンタの値が規定値に到達したとき、1 にセットされます。
8	R_SS_RC	0	rxudp_gd_frms カウンタの値が規定値に到達したとき、1 にセットされます。
7	R_SS_RC	0	rxipv6_nopay_frms カウンタの値が規定値に到達したとき、1 にセットされます。
6	R_SS_RC	0	rxipv6_hdrerr_frms カウンタの値が規定値に到達したとき、1 にセットされます。
5	R_SS_RC	0	rxipv6_gd_frms カウンタの値が規定値に到達したとき、1 にセットされます。
4	R_SS_RC	0	rxipv4_udsbl_frms カウンタの値が規定値に到達したとき、1 にセットされます。
3	R_SS_RC	0	rxipv4_frag_frms カウンタの値が規定値に到達したとき、1 にセットされます。
2	R_SS_RC	0	rxipv4_nopay_frms カウンタの値が規定値に到達したとき、1 にセットされます。
1	R_SS_RC	0	rxipv4_hdrerr_frms カウンタの値が規定値に到達したとき、1 にセットされます。
0	R_SS_RC	0	rxipv4_gd_frms カウンタの値が規定値に到達したとき、1 にセットされます。

MMC 受信チェックサムオフロード割込みレジスタの各ビットは、各受信 IPC(チェックサムオフロード)統計カウンタの値が最大値の半分に到達したとき、および最大値に到達したときに生成される割込みを保持します。このレジスタは 32 ビット幅です。

5. ディスクリプタ

ディスクリプタについて説明します。

ディスクリプタの概要

DMA は、「3.3. DMA コントローラ」で説明しているように、ホストにより構築されたメモリ上のディスクリプタリストに基づきデータを転送します。
各ディスクリプタには、2 つのバッファ, 2 つのバイトカウントバッファ, 2 つのアドレスポインタがあります。ディスクリプタアドレスは使用するバス幅にアラインされなければなりません。

DMA は、以降に説明する拡張ディスクリプタを使用します。Time-stamping 機能, Full IPC Offload 機能を使用する際、ソフトウェアは各ディスクリプタに対して、32 バイト(8 WORD)のメモリを割り当てることを必要とします。Time Stamp 機能, Full IPC Offload 機能を使用しない場合、ディスクリプタのサイズは 4WORD (TDES0～TDES3, RDES0～REDS3)です。
DMA Register 0 の bit7 を使って、この変更に対応できるように構成する必要があります。

5.1 送信拡張ディスクリプタ

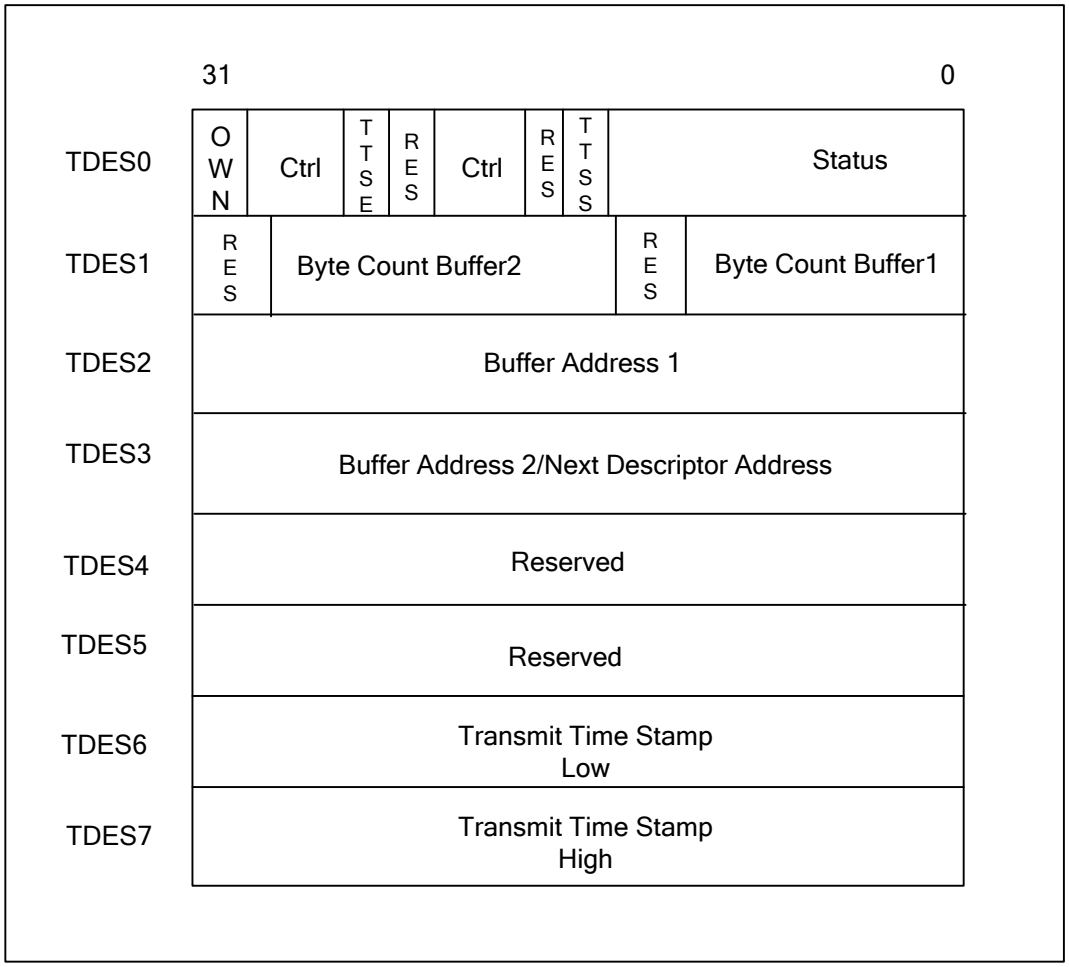
送信拡張ディスクリプタについて説明します。

送信拡張ディスクリプタの構造を Figure 5-1 に示します。アプリケーションソフトウェアは、ディスクリプタの初期化時に、コントロールビット TDES0[31:20]をプログラムしなければなりません。DMA がディスクリプタを更新するとき、OWN ビットを除くすべてのコントロールビットをライトバックし、ステータスビット[19:0]を更新します(OWN ビットはクリアされます)。

<注意事項>

- *Time Stamp 機能がイネーブルされているとき、ソフトウェアは DMA Register 0 の bit7 をセットして、DMA が拡張ディスクリプタサイズで動作するよう指示する必要があります。このコントロールビットがリセットされている場合、TDES4～TDES7 のディスクリプタスペースは無効です。*

Figure 5-1 送信拡張ディスクリプタの構成図



5.1.1 Transmit Enhanced Descriptor 0 (TDES0)

TDES0 は、送信フレームの制御ビット, ステータスビットにより構成されます。

TDES0

bit	31	30	29	28	27	26	25	24
Field	OWN	IC	LS	FS	DC	DP	TTSE	予約

bit	23	22	21	20	19	18	17	16
Field	CIC		TER	TCH	予約		TTSS	IHE

bit	15	14	13	12	11	10	9	8
Field	ES	JT	FF	IPE	LC	NC	LCO	EC

bit	7	6	5	4	3	2	1	0
Field	VF	CC[3:0]				ED	UF	DB

[bit31] OWN (OWN bit)

このビットをセットすると、ディスクリプタが DMA によって所有されるよう指示します。リセットされていると、ディスクリプタがホストによって所有されていることを示します。DMA は、フレームの送信完了時、またはディスクリプタで割り当てられているバッファが空になったときにこのビットをクリアします。フレームの最初のディスクリプタの OWN bit は、同じフレームの後続のすべてのディスクリプタがセットされた後にセットしなければなりません。これは、ディスクリプタのフェッチと、ドライバが OWN bit をセットする動作がレースコンディションを発生することを防止するためです。

[bit30] IC (Interrupt on Completion)

このビットをセットすると、現在のフレーム送信後、Transmit Interrupt (DMA Register 5[0])がセットされます。

[bit29] LS (Last Segment)

このビットをセットすると、バッファがフレームの最後の部分を含むことを指示します。このビットをセットする場合、TDES1 の TSB1 (送信バッファ 1 のサイズ) または TSB2 (送信バッファ 2 のサイズ) のいずれかが 0 以外である必要があります。

[bit28] FS (First Segment)

このビットをセットすると、バッファがフレームの最初の部分を含むことを指示します。

[bit27] DC (Disable CRC)

このビットをセットすると、Ethernet-MAC は送信フレームの終わりに巡回冗長検査(CRC)を付加しません。このビットは、FS(TDES0[28])がセットされているときのみ有効です。

[bit26] DP (Disable Pad)

このビットをセットすると、Ethernet-MAC は 64 バイト未満のフレームに対して自動 padding を行いません。このビットをリセットすると、DMA は 64 バイト未満のフレームに対して、DC (TDES0[27])にかかわらず自動 padding と CRC フィールドを付加します。このビットは、FS(TDES0[28])がセットされているときのみ有効です。

[bit25] TTSE (Transmit Time Stamp Enable)

このビットをセットすると、ディスクリプタにより参照される送信フレームの IEEE1588 ハードウェアタイムスタンプ生成を有効にします。このビットは、FS(TDES0[28])がセットされているときのみ有効です。

[bit23, 22] CIC (Checksum Insertion Control)

これらのビットは、チェックサム計算および挿入をコントロールします。ビットのエンコーディングを以下に示します。

00: チェックサム挿入がディセーブルされる。

01: IP ヘッダチェックサムの計算と挿入のみがイネーブルされる。

10: IP ヘッダチェックサムとペイロードチェックサムの計算と挿入がイネーブルされるが、擬似ヘッダチェックサムはハードウェアでは計算されない。

11: IP ヘッダチェックサムとペイロードチェックサムの計算と挿入がイネーブルされ、擬似ヘッダチェックサムがハードウェアで計算される。

＜注意事項＞

- *IPv6 認証ヘッダを使用する Ethernet 環境下では、必ず CIC は"00"または"01"にしてください。*

[bit21] TER (Transmit End of Ring)

このビットをセットすると、ディスクリプタリストの最後のディスクリプタに到達したことを指示します。DMA はリストのベースアドレスに戻り、ディスクリプタリングを形成します。

[bit20] TCH (Second Address Chained)

このビットをセットすると、TDES3 のアドレスが、バッファ 2 のアドレスではなく、次のディスクリプタアドレスであることを指示します。このビットをセットしているとき、バッファ 2 のサイズ (TBS2 ：TDES1[28:16])は、「don't care」値です。TER (TDES0[21]) をセットしている場合、このビットの値は無視されます。

[bit17] TTSS (Transmit Time Stamp Status)

このビットは、指定した送信フレームのタイムスタンプがキャプチャされたことを表すテータスビットです。このビットがセットされているとき、TDES6 および TDES7 には、送信フレームに対してキャプチャされたタイムスタンプ値が格納されています。このビットは、ディスクリプタの LS (TDES0[29])がセットされているときのみ有効です。

[bit16] IHE (IP Header Error)

このビットがセットされると、GMAC トランスミッタが IP データグラムヘッダでエラーを検出したことを表示します。トランスミッタは、IPv4 パケットのヘッダの長さをアプリケーションから受け取ったヘッダバイトの数と照合し、一致しなければエラーステータスを示します。IPv6 フレームに対しては、メインヘッダの長さが 40 バイトでない場合、ヘッダエラーが報告されます。さらに、IPv4 または IPv6 フレームの Ethernet Length/Type フィールド値は、パケットと共に受け取った IP ヘッダバージョンと一致しなければなりません。IPv4 フレームに対しては、Header Length フィールドの値が 0x5 未満である場合にもエラーステータスが示されます。

[bit15] ES (Error Summary)

下記のビットの論理 OR を表示します。

TDES0[16]:	IP Header Error
TDES0[14]:	Jabber Timeout
TDES0[13]:	Frame Flush
TDES0[12]:	IP Payload Error
TDES0[11]:	Loss of Carrier
TDES0[10]:	No Carrier
TDES0[9]:	Late Collision
TDES0[8]:	Excessive Collision
TDES0[2]:	Excessive Deferral
TDES0[1]:	Underflow Error

[bit14] JT (Jabber Timeout)
このビットがセットされると、GMAC トランスミッタでジャバータイムアウトが発生したことを表示します。このビットは、GMAC Register 0 の JD ビットがセットされていないときのみセットされます。

[bit13] FF (Frame Flushed)
このビットがセットされると、CPU が SW フラッシュコマンドを発行したため DMA/MTL がフレームをフラッシュしたことを表示します。

[bit12] IPE (IP Payload Error)
このビットがセットされると、GMAC トランスミッタが TCP, UDP, または ICMP IP データグラムペイロードでエラーを検出したことを表示します。
トランスミッタは、IPv4 or IPv6 ヘッダで受信したペイロードの長さをアプリケーションから受信した TCP, UDP, または ICMP の実際の数と照合してチェックし、一致しない場合はエラーステータスを発行します。

[bit11] LC (Loss of Carrier)
このビットがセットされると、フレーム送信中にキャリアが失われたこと(フレーム送信中に CRS 信号が 1 つ以上の送信クロック時間の間、非アクティブになっている)を表示します。このビットは、GMAC が半二重モードで動作し、送信フレームにコリジョンが発生しなかったときにセットされます。

[bit10] NC (No Carrier)
このビットがセットされると、PHY からのキャリアセンス信号が送信中にアサートされなかったことを表示します。

[bit9] LCO (Late Collision)
このビットがセットされると、コリジョンウィンドウ(MII モードでプリアンブルを含む 64 バイトタイム)以降にコリジョンが発生したためにフレーム送信が中断されたことを表示します。Underflow Error がセットされている場合は無効です。

<注意事項>

- ***フレームが送信された直後に COL 信号入力が変化した場合、これをレイトコリジョンとして認識できない場合があります。***

[bit8] EC (Excessive Collision)
このビットがセットされると、現在のフレーム送信試行中に 16 回のコリジョンが発生した後、送信が中断されたことを表示します。GMAC Register0 の DR (Disable Retry)ビットがセットされている場合、最初のコリジョンの後このビットがセットされ、フレームの送信が中断されます。

[bit7] VF (VLAN Frame)
このビットがセットされると、送信されたフレームが VLAN TAG フレームだったことを表示します。

[bit6:3] CC (Collision Count)
この 4 ビットカウンタの値は、フレーム送信完了までに発生したコリジョンの数を表示します。Excessive Collisions ビット (TDES0[8])がセットされているとき、このカウンタ値は無効です。

[bit2] ED (Excessive Deferral)
このビットがセットされると、GMAC Register0 の Deferral Check (DC)がセットされている場合に、24288 ビットタイム(Jumbo Frame イネーブルモードでは 155680 ビットタイム)を超える延期のために送信が終了したことを表示します。

[bit1] UF (Underflow Error)

このビットがセットされると、ホストメモリからのデータが遅れたため GMAC がフレームを中断したことを表示します。Underflow Error は、フレームの送信中に DMA が空の送信バッファ(Transmit Buffer)を検出したことを示します。送信プロセスは一時停止ステートに入り、Transmit Underflow (DMA Register 5[5])および Transmit Interrupt (DMA Register 5[0])の両方をセットします。

[bit0] DB (Deferred Bit)

このビットがセットされると、キャリアの存在のために GMAC が送信を延期したことを示します。このビットは半二重モード時のみ有効です。

5.1.2 Transmit Enhanced Descriptor 1 (TDES1)

TDES1 は、送信バッファ 1, 2 のサイズを指定します。

TDES1

bit	31	30	29	28	27	26	25	24
Field	予約			TBS2[12:8]				

bit	23	22	21	20	19	18	17	16
Field	TBS2[7:0]							

bit	15	14	13	12	11	10	9	8
Field	予約			TBS1[12:8]				

bit	7	6	5	4	3	2	1	0
Field	TBS1[7:0]							

[bit28:16] TBS2 (Transmit Buffer 2 Size)

このビットは、送信データバッファ 2 のサイズをバイト単位で指定します。TCH (TDES0[20]) がセットされている場合、このフィールドは無効です。

[bit12:0] TBS1 (Transmit Buffer 1 Size)

このビットは、送信データバッファ 1 のサイズをバイト単位で指定します。このフィールドが 0 である場合、DMA は送信データバッファ 1 を無視し、TCH (TDES0[20])の値に従って、送信データバッファ 2、または次のディスクリプタを使用します。

5.1.3 Transmit Enhanced Descriptor 2 (TDES2)

TDES2 は、送信バッファ 1 の物理アドレスを指定します。

TDES2								
bit	31	30	29	28	27	26	25	24
Field	B1AP[31:24]							
bit	23	22	21	20	19	18	17	16
Field	B1AP[23:16]							
bit	15	14	13	12	11	10	9	8
Field	B1AP[15:8]							
bit	7	6	5	4	3	2	1	0
Field	B1AP[7:0]							

[bit31:0] B1AP[31:0] (Buffer 1 Address Pointer)

このビットは、バッファ 1 の物理アドレスを指定します。バッファアドレスのアライメントについての制限はありません。

5.1.4 Transmit Enhanced Descriptor 3 (TDES3)

TDES3 は、送信バッファ 2 の物理アドレスまたは次のディスクリプタの物理アドレスを指定します。

TDES3

bit	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
Field	B2AP[31:24]							

bit	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
Field	B2AP[23:16]							

bit	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
Field	B2AP[15:8]							

bit	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
Field	B2AP[7:0]							

[bit31:0] B2AP[31:0] (Buffer 2 Address Pointer)

このビットは、ディスクリプタのリング構造を使用するとき、バッファ 2 の物理アドレスを指定します。Second Address Chained (TDES0[20])をセットしている場合、このアドレスは次のディスクリプタがある物理アドレスを指定します。この値は、TDES0[20]をセットしているときのみ、バス幅にアラインしなければなりません(LSB は内部的に無視されます)。

5.1.5 Transmit Enhanced Descriptor 6 (TDES6)

TDES6 には、送信時にキャプチャされたタイムスタンプ下位 32 ビットが DMA により格納されます。

TDES6

bit	31	30	29	28	27	26	25	24
Field	TTSL[31:24]							

bit	23	22	21	20	19	18	17	16
Field	TTSL[23:16]							

bit	15	14	13	12	11	10	9	8
Field	TTSL[15:8]							

bit	7	6	5	4	3	2	1	0
Field	TTSL[7:0]							

[bit31:0] TTSL (Transmit Frame Time Stamp Low)

このフィールドは、対応する送信フレームに対してキャプチャされたタイムスタンプの下位 32 ビットの値に、DMA により更新されます。このフィールドは、ディスクリプタの Last Segment bit (LS) ビットがセットされ、Time stamp status (TTSS) ビットがセットされている場合のみタイムスタンプが格納されます。

5.1.6 Transmit Enhanced Descriptor 7 (TDES7)

TDES7 には、送信時にキャプチャされたタイムスタンプ上位 32 ビットが DMA により格納されます。

TDES7

bit	31	30	29	28	27	26	25	24
Field	TTSH[31:24]							

bit	23	22	21	20	19	18	17	16
Field	TTSH[23:16]							

bit	15	14	13	12	11	10	9	8
Field	TTSH[15:8]							

bit	7	6	5	4	3	2	1	0
Field	TTSH[7:0]							

[bit31:0] TTSH (Transmit Frame Time Stamp High)

このフィールドは、対応する送信フレームに対してキャプチャされたタイムスタンプの上位 32 ビットの値に、DMA により更新されます。このフィールドは、ディスクリプタの Last Segment bit (LS)ビットがセットされ、Time stamp status (TTSS)ビットがセットされている場合にのみタイムスタンプが格納されます。

5.2 受信拡張ディスクリプタ

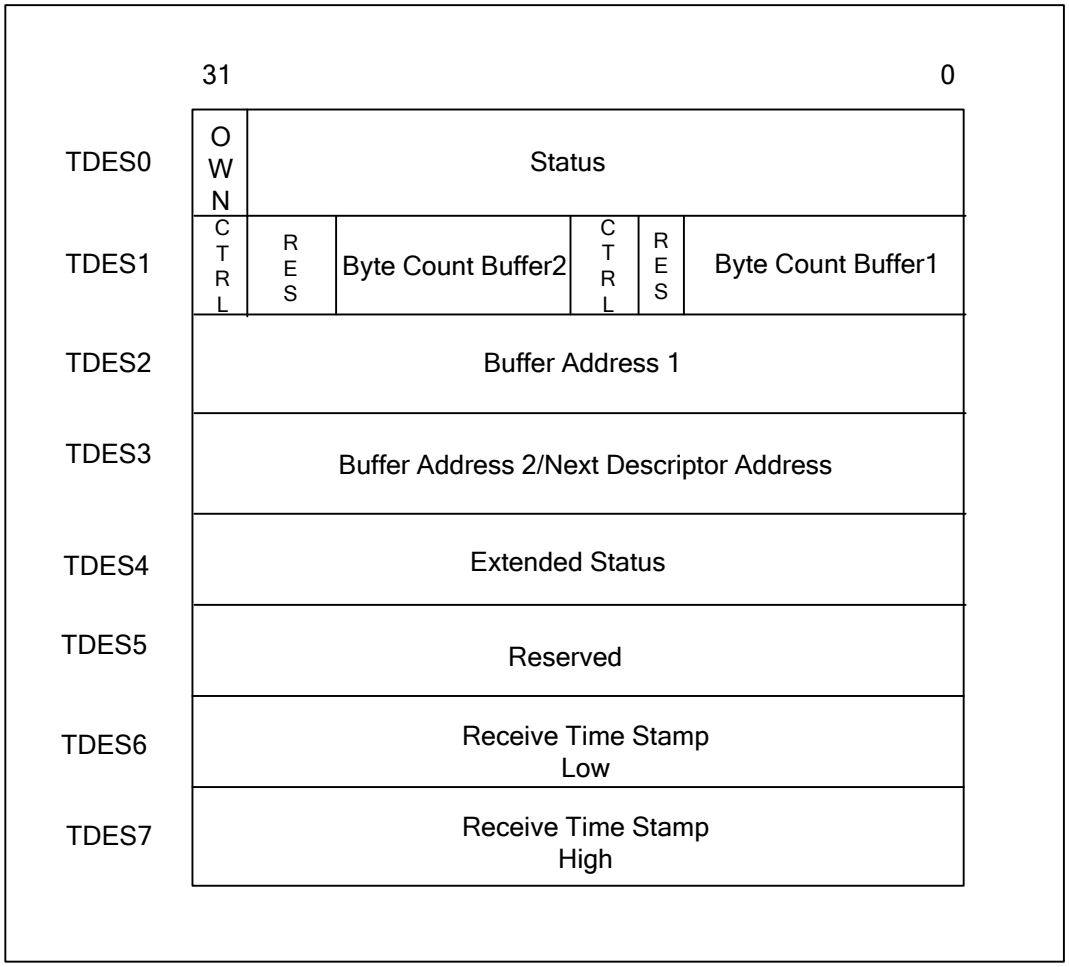
受信拡張ディスクリプタについて説明します。

受信 DMA は、フレームを受信するとき、少なくとも 2 つのディスクリプタを必要とします。DMA の受信ステートマシンは、フレームの受信を想定して、常に余分のディスクリプタを取得しようとします(受信フレームのサイズは未知です)。フレームを受信しなかった場合でも、受信 DMA がディスクリプタを閉じる前に次のディスクリプタを取得しようとします。受信拡張ディスクリプタの構造を Figure 5-2 に示します。

<注意事項>

- *Time Stamp 機能がイネーブルされているとき、ソフトウェアは DMA Register 0 の bit7 をセットして、DMA が拡張ディスクリプタサイズで動作するよう指示する必要があります。このコントロールビットがリセットされている場合、RDES4～RDES7 のディスクリプタスペースは無効です。*

Figure 5-2 受信拡張ディスクリプタ構成図



5.2.1 Receive Enhanced Descriptor 0 (RDES0)

RDES0 は、受信フレームの制御ビット, ステータスビットにより構成されます。

RDES0

bit	31	30	29	28	27	26	25	24
Field	OWN	AFM	FL[13:8]					

bit	23	22	21	20	19	18	17	16
Field	FL[7:0]							

bit	15	14	13	12	11	10	9	8
Field	ES	DE	SAF	LE	OE	VLAN	FS	LS

bit	7	6	5	4	3	2	1	0
Field	TS	LC	FT	RWT	RE	DE	CE	ESA

[bit31] OWN (OWN bit)

このビットをセットすると、ディスクリプタが DMA によって所有されるよう指示します。リセットされていると、ディスクリプタがホストによって所有されていることを表示します。DMA は、フレーム受信完了時、またはこのディスクリプタで割り当てられているバッファが一杯のときに、このビットをクリアします。

[bit30] AFM(Destination Address Filter Fail)

このビットがセットされると、フレームが GMAC の DA Filter でフェイルしたことを表示します。

[bit29:16] FL[13:0] (Frame Length)

このビットは、ホストメモリに転送された受信フレームのバイト長(CRC を含む)を表示します。このフィールドは、Last Descriptor (RDES0[8])がセットされ、Descriptor Error (RDES0[14])または Overflow Error (RDES0[11])のどちらかがリセットされているときに有効です。

このフィールドは、Last Descriptor (RDES0[8])がセットされているときに値が確定しますが、Last Descriptor ビットおよび Error Summary ビットがセットされていないとき、このフィールドは現在のフレームの転送済みバイトの累積数を表示します。

[bit15] ES (Error Summary)

下記のビットの論理 OR を表示します。

RDES0[1] : CRC Error
RDES0[3] : Receive Error
RDES0[4] : Watchdog Timeout
RDES0[6] : Late Collision
RDES0[7] : IPC Checksum
RDES0[11] : Overflow Error
RDES0[14] : Descriptor Error
RDES4[4:3] : IP Header/Payload Error

このフィールドは Last Descriptor (RDES0[8])がセットされているときに有効です。

[bit14] DE (Descriptor Error)

このビットがセットされると、フレームが現在のディスクリプタバッファ内に収まらず、DMA が次のディスクリプタを所有していないためにフレームが切り捨てられたことを表示します。フレームは切り捨てられます。このフィールドは、LS(RDES0[bit8])がセットされているときのみ有効です。

[bit13] SAF(Source Address Filter Fail)

このビットがセットされると、フレームの SA フィールドが GMAC の SA Filter でフェイルしたことを表示します。

[bit12] LE (Length Error)
このビットがセットされると、受信フレームの実際の長さと Length/Type フィールドの値が一致しないことを表示します。このビットは、Frame Type (RDES0[5])ビットがリセットされているときのみ有効です。CRC エラーがあるとき、Length エラーステータスは無効です。

[bit11] OE (Overflow Error)
このビットがセットされると、受信したフレームが MTL でのバッファオーバフローのために損傷したことを表示します。

[bit10] VLAN (VLAN tag)
このビットがセットされると、このディスクリプタで指定しているフレームが、GMAC によってタグ付けされた VLAN フレームであることを表示します。

[bit9] FS (First Descriptor)
このビットがセットされると、このディスクリプタが、フレームの最初のバッファを含んでいることを表示します。最初のバッファのサイズが 0 である場合、2 番目のバッファがフレームの先頭を含んでいます。2 番目のバッファのサイズも 0 である場合、次のディスクリプタがフレームの先頭を含んでいます。

[bit8] LS (Last Descriptor)
このビットがセットされると、このディスクリプタで指定しているバッファがフレームの最後のバッファであることを表示します。

[bit7] TS (Time Stamp)
このビットがセットされると、タイムスタンプのスナップショットがディスクリプタのワード 6 (RDES6)および 7 (RDES7)に書き込まれたことを表示します。このビットは、Last Descriptor ビット (RDES0[8])がセットされているときのみ有効です。

[bit6] LC (Late Collision)
このビットがセットされると、半二重モードでのフレーム受信中にレイトコリジョンが発生したことを表示します。

[bit5] FT (Frame Type)
このビットがセットされると、受信フレームが Ethernet タイプのフレーム(LT フィールドが 0x0600 以上)であることを表示します。このビットがリセットされると、受信フレームが IEEE802.3 フレームであることを表示します。このビットは、14 バイト未満の Runt フレームでは無効です。

[bit4] RWT (Receive Watchdog Timeout)
このビットがセットされると、現在のフレームの受信中に Receive Watchdog Timer が時間切れになったことを表示します。ウォッチドッグのタイムアウト後に現在のフレームが中途で切り捨てられます。

[bit3] RE (Receive Error)
このビットがセットされると、RX_DV 信号がアサートされているフレームの受信中に、RX_ER 信号がアサートされたことを表示します。このエラーはまた、GMII 時の半二重モードでのキャリアエクステンションエラーを含みます。エラーは、エクステンションの過小/不在、またはエクステンション中のエラー(RXD ≠ 0f)である場合があります。

[bit2] DE (Dribble Bit Error)
このビットがセットされると、受信フレームがバイトの非整数倍長であったことを表示します。

[bit1] CE (CRC Error)
このビットがセットされると、受信フレームで Cyclic Redundancy Check (CRC) Error が発生したことを表示します。このフィールドは、Last Descriptor (RDES0[8])がセットされているときのみ有効です。

[bit0] ESA (Extended Status Available)
このビットがセットされると、ディスクリプタのワード 4 (RDES4)の拡張ステータスが利用できることを表示します。このビットは Last Descriptor ビット(RDES0[8])がセットされているときのみ有効です。

5.2.2 Receive Enhanced Descriptor 1 (RDES1)

RDES2 は、受信バッファ 1, 2 のサイズと制御情報を指定します。

RDES1

bit	31	30	29	28	27	26	25	24
Field	DIC	予約		RBS2[12:8]				

bit	23	22	21	20	19	18	17	16
Field	RBS2[7:0]							

bit	15	14	13	12	11	10	9	8
Field	RER	RCH	予約	RBS1[12:8]				

bit	7	6	5	4	3	2	1	0
Field	RBS1[7:0]							

[bit31] DIC(Disable Interrupt on Completion)

このビットをセットすると、このディスクリプタで指定しているバッファ内で終了する受信フレームに対し、ステータスレジスタの RI (DMA Register 5[6])をセットしないよう指示します。このため、そのフレームに対する RI によるホストへの割込み信号アサートがディセーブルされます。

[bit28:16] RBS2 (Receive Buffer 2 Size)

このビットは、バッファ 2 のサイズをバイト単位で指示します。RDES3 (バッファ 2 のアドレスポインタ)がバス幅にアラインされていない場合でも、バッファサイズは 4 の倍数でなければなりません。バッファサイズが 4 の倍数でない場合の動作は定義されていません。RDES1[14]がセットされている場合、このフィールドは無効です。

[bit15] RER (Receive End of Ring)

このビットをセットすると、ディスクリプタリストが最後のディスクリプタに達したことを指示します。DMA はリストのベースアドレスに戻り、ディスクリプタリングを形成します。

[bit14] RCH (Second Address Chained)

このビットをセットすると、RDES3 のアドレスが、バッファ 2 のアドレスではなく、次のディスクリプタアドレスであることを指示します。このビットをセットしているとき、バッファ 2 のサイズ RBS2 (RDES1[28:16])は、「don't care」値です。RER (RDES1[15]) をセットしている場合、このビットの値は無視されます。

[bit12:0] RBS1 (Receive Buffer 1 Size)

このビットは、バッファ 1 のサイズをバイト単位で指示します。RDES2 (バッファ 1 のアドレスポインタ)がバス幅にアラインされていない場合でも、バッファサイズは、バス幅に応じて 4, 8, または 16 の倍数でなければなりません。バッファサイズが 4 の倍数でない場合の動作は定義されていません。このフィールドが 0 である場合、DMA はこのバッファを無視し、RCH (RDES1[14])の値に従ってバッファ 2 または次のディスクリプタを使用します。

5.2.3 Receive Enhanced Descriptor 2 (RDES2)

RDES2 は、受信バッファ 1 の物理アドレスを指定します。

RDES2								
bit	31	30	29	28	27	26	25	24
Field	B1AP[31:24]							
bit	23	22	21	20	19	18	17	16
Field	B1AP[23:16]							
bit	15	14	13	12	11	10	9	8
Field	B1AP[15:8]							
bit	7	6	5	4	3	2	1	0
Field	B1AP[7:0]							

[bit31:0] B1AP[31:0] (Buffer 1 Address Pointer)

このビットは、バッファ 1 の物理アドレスを指定します。以下の条件を除いて、バッファアドレスのアライメントに制限はありません：

DMA は RDES2 の値をバッファ 1 のアドレス生成のために使用します。DMA は、フレームの先頭部分の転送の際、RDES2[1:0] ビットを 00 と見なして、32 ビット幅でメモリに書き込む動作を行います。ただし、フレームデータは設定されたバッファアドレスポインタにシフトして書き込みます。バッファアドレスが 32 ビット幅にアライメントされていない場合、下位側にダミーデータを書き込みます。DMA は、フレームの中央または最後の部分の転送の際は、RDES2[1:0]を無視し、32 ビット幅でメモリに書込みを行います。

5.2.4 Receive Enhanced Descriptor 3 (RDES3)

RDES3 は、受信バッファ 2 の物理アドレスまたは次のディスクリプタの物理アドレスを指定します。

RDES3

bit	31	30	29	28	27	26	25	24
Field	B2AP[31:24]							

bit	23	22	21	20	19	18	17	16
Field	B2AP[23:16]							

bit	15	14	13	12	11	10	9	8
Field	B2AP[15:8]							

bit	7	6	5	4	3	2	1	0
Field	B2AP[7:0]							

[bit31:0] B2AP[31:0] (Buffer 2 Address Pointer)

このビットは、ディスクリプタのリング構造を使用するとき、バッファ 2 の物理アドレスを指定します。Second Address Chained (RDES1[14])をセットしているとき、このアドレスは次のディスクリプタがある物理アドレスを指定します。
RDES1[14]をセットしている場合、次のディスクリプタアドレスポインタはバス幅にアラインされていなければなりません (RDES3[1:0] = 0。LSB は内部的に無視されます)。
RDES1[14]をリセットしている場合、次の条件を除いて、RDES3 値のアライメントに制限はありません。
RDES3 の値をフレームの先頭の転送に使用しているとき、DMA はその設定値をアドレス生成のために使用し、その位置からフレームの転送を行います。RDES3 の値をフレームの中央または最後の部分の転送に使用している場合、DMA は RDES3[1:0]を無視します。

5.2.5 Receive Enhanced Descriptor 4 (RDES4)

RDES4 には、受信フレームの拡張ステータス情報が格納されます。

RDES4 bit	31	30	29	28	27	26	25	24
Field	予約							

bit	23	22	21	20	19	18	17	16
Field	予約							

bit	15	14	13	12	11	10	9	8
Field	予約	TD	PV	PFT	MT			

bit	7	6	5	4	3	2	1	0
Field	IP6R	IP4R	IPCB	IPE	IPHE	IPT		

[bit14] TD (Timestamp Dropped)

このビットがセットされていると、受信フレームのタイムスタンプがキャプチャされたが、オーバフローのために RxFIFO で廃棄されたことを表示します。

[bit13] PV (PTP Version)

このビットがセットされていると、受信した PTP メッセージが IEEE 1588 バージョン 2 フォーマットであることを表示します。リセットされていると、バージョン 1 フォーマットであることを表示します。このビットは MT (message type)が、0000 でない場合のみ有効です。

[bit12] PFT (PTP Frame Type)

このビットがセットされていると、PTP message over Ethernet を受信したことを表示します。このビットがセットされておらず、MT (message type)が 0000 でない場合、PTP message over UDP-IPv4 または、PTP message over UDP-IPv6 を受信したことを表示します。IPv4 または IPv6 に関する情報は、bit6 および bit7 から取得できます。

[bit11:8] MT (Message Type)

このビットは、受信したメッセージのタイプを表示します。

0000 : No PTP message received or PTP packet with Reserved message type (*)
0001 : SYNC (all clock types)
0010 : Follow_Up (all clock types)
0011 : Delay_Req (all clock types)
0100 : Delay_Resp (all clock types)
0101 : Pdelay_Req (in peer-to-peer transparent clock)
0110 : Pdelay_Resp (in peer-to-peer transparent clock)
0111 : Pdelay_Resp_Follow_Up (in peer-to-peer transparent clock)
1000 : Announce
1001 : Management
1010 : Signaling
1011-1110: Reserved
1111 : PTP packet with reserved message type (Control Field 0x05- 0x0f)

* PTP control field を 0xMN として、M=0x0-0x0F, N=0x0 or 0x1
かつ、受信した UDP 送信先ポートアドレスが General PTP message PTP control field を 0xMN として、M=0x0-0x0F, N=0x2, 0x3, 0x4
かつ、受信した UDP 送信先ポートアドレスが Event PTP message

[bit7] IP6R (IPv6 Packet Received)
このビットがセットされると、受信したパケットが IPv6 パケットであることを表示します。このビットは、IPC(Checksum Offload: GMAC Register0[10])をセットしているときのみ更新されます。

[bit6] IP4R (IPv4 Packet Received)
このビットがセットされると、受信したパケットが IPv4 パケットであることを表示します。このビットは、IPC(Checksum Offload: GMAC Register0[10])をセットしているときのみ更新されます。

[bit5] IPCB (IP Checksum Bypassed)
このビットがセットされると、チェックサムオフロードエンジンがバイパスされたことを表示します。

[bit4] IPE (IP Payload Error)
このビットがセットされると、GMAC が計算した 16 ビットの IP ペイロードチェックサム(すなわち、TCP, UDP, ICMP のチェックサム)値と、受信したセグメントの対応するチェックサムフィールドの値が一致しなかったことを表示します。また、TCP, UDP, ICMP のセグメント長が、IP Header フィールドのペイロード長と一致しないときも、このビットがセットされます。

[bit3] IPHE (IP Header Error)
このビットがセットされると、GMAC が計算した 16 ビットの IPv4 ヘッダチェックサムと、受信したチェックサムバイトが一致しない、または IP データグラムのバージョンが Ethernet Type 値と一致しないことを表示します。

[bit2:0] IPT (IP Payload Type)
このビットは、受信チェックサムロードエンジン (Receive Checksum Offload Engine : COE)によって処理された IP データグラムの中のカプセル化されたペイロードタイプを表示します。COE は、IP ヘッダエラー, IP 断片化のために、IP データグラムのペイロードを処理しなかった場合、このビットを 000 にします。

000 : 未知、または IP ペイロードを処理しなかった
001 : UDP
010 : TCP
011 : ICMP
1xx : 予約

5.2.6 Receive Enhanced Descriptor 6 (RDES6)

RDES6 には、受信時にキャプチャされたタイムスタンプ下位 32 ビットが DMA により格納されます。

RDES6								
bit	31	30	29	28	27	26	25	24
Field	RTSL[31:24]							
bit	23	22	21	20	19	18	17	16
Field	RTSL[23:16]							
bit	15	14	13	12	11	10	9	8
Field	RTSL[15:8]							
bit	7	6	5	4	3	2	1	0
Field	RTSL[7:0]							

[bit31:0] RTSL (Receive Frame Time Stamp Low)

このフィールドは、DMA によって、対応する受信フレームに対してキャプチャされたタイムスタンプの下位 32 ビットの値に更新されます。このフィールドは、受信フレームの最後のディスクリプタ(Last Descriptor (RDES0[8])によって示される)のみ、DMA によって更新されます。

5.2.7 Receive Enhanced Descriptor 7 (RDES7)

RDES7 には、受信時にキャプチャされたタイムスタンプ上位 32 ビットが DMA により格納されます。

RDES7								
bit	31	30	29	28	27	26	25	24
Field	RTSH[31:24]							
bit	23	22	21	20	19	18	17	16
Field	RTSH[23:16]							
bit	15	14	13	12	11	10	9	8
Field	RTSH[15:8]							
bit	7	6	5	4	3	2	1	0
Field	RTSH[7:0]							

[bit31:0] RTSH (Receive Frame Time Stamp High)

このフィールドは、DMA によって、対応する受信フレームに対してキャプチャされたタイムスタンプの上位 32 ビットの値に更新されます。このフィールドは、受信フレームの最後のディスクリプタ(Last Descriptor (RDES0[8])によって示される)のみ、DMA によって更新されます。

6. プログラミングガイド

Ethernet-MAC を使用する際のプログラミングガイドを記載します。

6.1 DMA の初期化 - ディスクリプタ

DMA の初期化では、以下の処理を行います。

1. ソフトリセットを実行します。これによって Ethernet-MAC 内部レジスタおよびロジックがすべてリセットされます(DMA Register 0: Bus Mode Register, bit0)。
2. リセット処理が完了するまで待ちます(DMA Register 0: Bus Mode Register[0]をポーリングします。このビットはリセット動作完了後にクリアされます)。
3. DMA Register 11 (AHB Status Register, bit0)をポーリングし、前に開始していた(ソフトリセットの前)、または現在進行中の AHB トランザクションが完了していることを確認します。

* アプリケーションがソフトリセット後に(性能上の理由で) AHB Status Register をポーリングできない場合、次のステップに進んでから、DMA の動作をトリガする前に、もう一度このレジスタをチェックすることを推奨します(ステップ 12 の説明を参照)。

4. PS ビット(GMAC Register 0 :MCR[15])に 1 をセットし、MII/RMII インタフェースを選択します。
   このレジスタの初期値は 0 であるため、必ず 1 の書込みが必要です。
5. 下記のフィールドをプログラムし、DMA Register0 (Bus Mode Register)の値を設定することによって Bus Mode Register を初期化します。
   a. MB (Mixed Burst), AAL, FB (Fixed Burst)
      FM3 ファミリ マイコンのシステムバスは、マルチレイヤバス構成であり、CPU と Ethernet-MAC の DMA はそれぞれ独立したバスを持っています。また、Ethernet-MAC の DMA が不定長バースト (INCR) による転送動作を行う場合、CPU のバスアクセス動作は随時実行されます。このため、MB=1, AAL=0, FB=0 のレジスタ設定を推奨します。
   b. PBL, RPBL, 8xPBL, USP (Burst length values and burst mode values)
   c. DSL (Descriptor length)
      Ring Mode を使用する場合、設定が必要です。
   d. TXPR, PR, DA (Tx Rx DMA arbitration scheme)
      Ethernet-MAC の送信 DMA と受信 DMA によるシステムメモリへのアクセスが競合集中する場合、それぞれのメモリ・アクセス動作が遅延する状態が発生します。この場合に、送信 DMA と受信 DMA 間のバス占有率を PR, DA により制御できます。
   e. ATDS (Alternate Descriptor Size)
      タイムスタンプ機能, 受信 IPC オフロード機能を使用する際は、セットします。
6. 送信および受信のための適切なディスクリプタ Chain を作成します。この際、受信ディスクリプタは、DMA の所有にしておく必要があります。(RDES0[31]をセットしておく必要があります)。OSF モードを使用するとき、少なくとも 2 つのディスクリプタが必要です。
7. いずれかのディスクリプタを再使用する前には、ソフトウェアがリング内に 3 つ以上の異なる送信または受信ディスクリプタを作成している必要があります。
8. 受信および送信ディスクリプタリストアドレスを送信および受信ディスクリプタのベースアドレスによって初期化します(DMA Register 3 - Receive Descriptor List Address Register, DMA Register 4 - Transmit Descriptor List Address Register)。
9. 下記のフィールドをプログラムし、DMA Register6 (Operation Mode Register)の値を設定することによってオペレーションモードを初期化します。
   a. RSF, TSF (Receive and Transmit Store and Forward)
   b. RTC, TTC (Receive and Transmit Threshold Control)
   c. FEF, FUF (Error Frame and undersized good frame forwarding enable)
   d. OSF (Operation on Second Frame)
10. ステータスレジスタがセットされているビット(割込みビットのみ)は、書込みにより、割込み要求をクリアしておきます。例えば、通常割込みサマリ(NIS)では、bit16 に 1 を書き込むことによってこのビット(DMA Register 5 :Status Register SR[16])をクリアします。

11. 割込みイネーブルレジスタ(DMA Register7 - Interrupt enable register)をプログラムすることによって、割込みを許可しておきます。
12. DMA Register 11 (AHB Status Register)を読み出し、前のすべての AHB トランザクションが完了していることを確認します。
13. コントロールレジスタ(DMA Register6 - operation mode register)の SR(bit1)および ST (bit13)をセットすることによって、Receive および Transmit DMA の動作を開始します。

6.2 GMAC の初期化

DMA の初期化シーケンスを完了した後、以下の GMAC 初期化処理を実行します。DMA をセットアップする前に GMAC の初期化を実行した場合、GMAC 受信部の許可指示(下記の最後のステップ)は、DMA がアクティブ化された後のみ可能となります。そうしない場合、受信フレームにより、受信 FIFO が充填され、オーバフローが発生することがあります。

1. 外部 PHY のマネジメントサイクル制御のため、GMAC Register4 - GMII Address Register をプログラムします。例えば、Physical Layer Address PA (bit15:11)など。PHY への書込みおよび PHY からの読出しのために bit0 (GMII Busy)をセットします。
2. リンク状態, オペレーション速度, オペレーションモードの PHY から読み出された 16 ビットデータを、GMAC Register5 (GMII Data Register)から読み出します。
3. GMAC Register0 (MAC Configuration register)の該当するフィールドをプログラムします。例えば、送信中のフレーム間隔, ジャバーのディセーブルなど。オートネゴシエーションの結果を元に、bit11 (全二重/半二重), bit14 (10Mpbs/100Mbps)などを設定します。その後 PHY インタフェースクロックの 2 サイクル時間待機します。
4. MAC アドレスを設定します(GMAC Register16, 17：MAR0H, MAR0L)。
5. ハッシュフィルタレジスタをプログラムします(GMAC Register2, 3:MHTRH, MHTRL)。
6. 受信フレームの適切なフィルタを下記のフィールドで設定します。 (GMAC Register1:MFFR)
   a. Receive All
   b. Promiscuous mode
   c. Hash or Perfect Filter
   d. Unicast, Multicast, broad cast and control frames filter settings etc.
7. 適切なフローコントロールを下記のフィールドで設定します。 (GMAC Register6: FCR)
   a. Pause time and other pause frame control bits
   b. Receive and Transmit Flow control bits
   c. Flow Control Busy/Backpressure Activate
8. システム構成の必要に応じ、Interrupt Mask レジスタビットをプログラムします。
9. GMAC Register0:MCR の Transmit Enable (TE:bit3)および Receive Enable (RE:bit2)をセットします。

6.3 通常の受信および送信動作

通常の動作では、以下のステップを実行します。

1. 通常の送信と受信割込みが発生した場合、割込みステータスを読み出します。次にディスクリプタをポーリングし、ホストが所有しているディスクリプタのステータス(送信または受信)を読み出します。
2. ディスクリプタに適切な値を設定します。データの送信および受信を再開するためには、DMA が送信および受信ディスクリプタを所有しているようにする必要があります。
3. DMA がディスクリプタを所有していない場合(または利用可能なディスクリプタがない場合)、DMA は Suspend ステートに入ります。送信または受信を再開するには、ディスクリプタを開放し、Tx/Rx ポーリング要求レジスタ(DMA Register1 - Transmit Poll Demand Register と DMA Register2 - Receive Poll Demand Register)に 0 を書き込むことによってポーリング要求を発行します。
4. デバッグ処理用に、現在の送信と受信のディスクリプタのアドレスポインタ値を読み出すことができます。(DMA Register18 - Current Host Transmit Descriptor Register と DMA Register19 - Current Host Receive Descriptor Register)デバッグ処理用に、現在のホストメモリ上の送信バッファアドレスポインタと受信バッファアドレスポインタを読み出すことができます(DMA Register20 - Current Host Transmit Buffer Address Register および DMA Register21 - Current Host Receive Buffer Address Register).

6.4 動作の停止および開始

データ転送を一時的に停止する必要があるとき、以下のステップを実行します。

1. 制御レジスタ(DMA Register6:OMR)の ST (bit13)をクリアすることによって送信 DMA をディセーブルします(送信 DMA が動作している場合)
2. 実施中のフレーム転送処理が完了するまで待ちます。(DMA Register 5[22:20]を読み出すことによってチェックします)
3. 送信 FIFO をフラッシュし、empty 動作を迅速化します。
4. GMAC Register0:MCR の Transmitter Enable (TE:bit3)と Receiver Enable (RE:bit2)をクリアし、GMAC トランスミッタおよび GMAC レシーバをディセーブルします。
5. リンクが確立された後、最新の構成を知るために PHY レジスタを読み出し、それに従って GMAC のレジスタをプログラムします。
6. 送信 DMA をイネーブルにし、GMAC トランスミッタと GMAC レシーバをイネーブルすることによって、動作を再開します。受信 DMA をディセーブルする必要はありません。GMAC レシーバがディセーブルされると受信 FIFO にデータは取り込まれません。

6.5 リンクの停止/確立シーケンス

PHY とのリンクがダウンしているとき、以下のステップを実行します。

1. リンクが確立するまで待ち、TX および RX クロックが停止している場合は RX および TX クロックが入力されるまで待ちます。
2. DMA Register 0 (Bus Mode Register)で Software Reset (SWR:bit0)をセットすることによって、ソフトリセットを実行し、DMA および GMAC のトランスミッタ/レシーバをクリアします。
3. 「6.1 DMA の初期化 - ディスクリプタ」の初期化の手順に従います。

6.6 IEEE タイムスタンプ生成のプログラミング・ガイドライン

システムタイム生成の初期化ガイドライン

タイムスタンプ生成機能は、Time Stamp コントロールレジスタの bit0 をセットすることによってイネーブルできます。しかし、このビットをセットした後、Time Stamp カウンタを初期化し、タイムスタンプ動作を開始することが重要です。これは Ethernet-MAC の初期化中に、以下のステップを実行することによって行います。

1. GMAC Register 15:IMR の bit9 をセットし、Time Stamp Trigger 割込みをマスクします。
2. GMAC Register 448 :Time Stamp Control register の bit0 をセットし、タイムスタンプ機能をイネーブルします。
3. GMAC Register 449: Sub-Second Increment register を PTP クロック周波数に基づいてプログラムします。
4. Fine Correction アプローチを使用する場合、GMAC Register 454 : Time Stamp Addend レジスタをプログラムし、Time Stamp Control register の Addend Register Update (TARU:bit5)をセットします。
5. TARU :bit5 がクリアされるまで、レジスタをポーリングします。
6. Fine Update メソッドを選択する場合、Time Stamp Control register の TFCU:bit1 をセットします。
7. GMAC Register 452:System Time - Second Update Register と GMAC Register 453 : System Time - Nanosecond Update Register に適切な時間値をプログラミングします。
8. Time Stamp Control register の Time Stamp Init (TSI:bit2)をセットします。
9. Time Stamp カウンタは、Time Stamp Update レジスタに書き込まれた値によって初期化され、すぐに動作を開始します。
10. 適切なタイムスタンプ生成のため、GMAC レシーバおよびトランスミッタをイネーブルします。
11. GMAC Register 15:IMR の bit9 をリセットし、Time Stamp Trigger 割込みを許可します。

<注意事項>

- *タイムスタンプ動作が、Time Stamp Control register の bit0 のクリアによってディセーブルされた場合、タイムスタンプ操作を再開するためには上記のステップを繰り返さなければなりません。*

システム時間補正

1 回の処理で、システム時間を同期化・更新する (粗補正法)には、以下のステップを実行します。

1. Time Stamp Update レジスタ(GMAC register 452, 453)にオフセット(正または負)を書き込みます。
2. Time Stamp Control レジスタ(GMAC Register 448)の TSU:bit3 をセットします。
3. TSU ビットがクリアされると、Time Stamp Update レジスタの値がシステム時間に加算されるか、またはシステム時間から減算されます。

システム時間のジッタを減らすために、システム時間を同期化・更新する(細密補正法)には、以下のステップを実行します。

1. 「System Time Register Module」で説明しているアルゴリズムを使用して、システム時間を遅くする、または速くするときの増分値を計算します。
2. Time Stamp Addend レジスタ(GMAC Register 454)を新しい値に更新し、次に Time Stamp Control register の TARU(bit5)をセットします。
3. Addend レジスタの新しい値で、システム時間が補正されるまで待ちます。以下のように、システム時間がターゲット値に達した後に Time Stamp Trigger 割込みがアクティブになるようにします。
4. GMAC Register 455 および GMAC Register 456 で必要なターゲット時間をプログラムします。GMAC Register 15 の bit9 をクリアし、割込みマスクを解除します。
5. GMAC Register 448(Time Stamp Control Register)の TITE ビット(bit4)をセットします。
6. このトリガによって割込みが発生したとき、GMAC Register 14 が読み出します。
7. GMAC Register 454(Time Stamp Addend Register)を古い値で再プログラミングし、TARU(bit5)を再びセットします。

6.7 Energy Efficient Ethernet のプログラミング・ガイドライン

GMAC の初期化中に以下のステップを実行します。

1. MDIO インタフェースを通じて PHY レジスタを読み出し、EEE 機能があるかどうかを調べ、タイマ値を折衝します。
2. MDIO インタフェースを通じて PHY レジスタをプログラムします(LPI モードで RX_CLK を停止するかどうかを PHY に指示する RX_CLK_stoppable ビットを含む)。
3. GMAC Register 13 (LPI Timers Control Register)で bit[25:16] (LIT: LPI LS TIMER)および bit[15:0] (TWT: LPI TW TIMER)をプログラムします。
4. MDIO インタフェースを使って PHY チップのリンクステータスを読み出し、それに従って GMAC Register 12 (LPI Control and Status Register)の PLS:bit17 を更新します。PHY チップのリンクステータスが変更されたときは常にこの更新を行う必要があります。
5. GMAC Register 12 (LPI Control and Status Register)の LPI Enable (LPIEN:bit16)をセットして、GMAC を LPI ステートに入れます。
   進行中の送信が完了した後、GMAC は LPI モードになり、Transmit LPI Entry (TLPIEN:bit0)をセットします。

＜注意事項＞

- *GMAC が送信 FIFO のキューに入っているすべてのフレームの送信を完了した後のみ LPI ステートに入るようにするには、GMAC Register 12 (LPI Control and Status Register)の bit19 (LPITXA: LPI TX Automate) をセットする必要があります。*
- *LPI ステート中に SYS_CLK, GMII 送信クロックまたはシステムのほかの部分をオフに切り換える場合、GMAC Register 12 (LPI Control and Status Register)の TLPIEN Interrupt が生成されるまで待機する必要があります。LPI ステートを終了するとき、ステップ 6 を実行する前にクロックを復元します。*

6. GMAC の LPI ステートを終了するために、GMAC Register 12 (LPI Control and Status Register)の bit16 (LPIEN: LPI Enable)をリセットします。
   GMAC は、bit[15:0] (TWT: LPI TW TIMER)でプログラムされている時間の間待機してから、TLPIEX 割込みステータスビットをセットし、送信を再開します。

6.8 スタンバイモードへの移行と SYS_CLK の停止について

CPU が WFI 命令を実行すると、STOP モード・タイマモードに移行し、Ethernet-MAC への SYS_CLK (マイコンの HCLK が接続されています。) 供給が停止します。このため、STOP モード, タイマモードに移行する際には、GMAC Register 11 (PMTR)の PD ビットにより、GMAC レシーバの動作を停止しておく必要があります。
この際、同時に、Wake-Up パケットの受信により、INT_PMT 割込みを発生するように設定しておくことができます。
Ethernet-MAC は、Wake-Up パケットの受信により、INT_PMT 割込みを発生させます。CPU は、INT_PMT 割込みにより、Run モードへ復帰します。

Appendixes

1. 主な変更内容

Spansion Publication Number: MN709-00017

ページ	場所	変更箇所
Revision 1.0		
-	-	Initial release
Revision 2.0		
6	本マニュアルにおける対象製品	TYPE5-M4 の対象型格を追加

注意事項: 以降の変更点に関しては、「改訂履歴」を参照してください。

改訂履歴

文書名: 32 ビット・マイクロコントローラ FM4 ファミリ Peripheral Manual Ethernet 編 文書番号: 002-04964			
版	**ECN 番号**	**変更者**	**変更内容**
**	-	AKIH	サイプレスとしてドキュメントコード 002-04964 に登録しました。 本版の内容およびフォーマットに変更はありません。 (これは英語版 002-04963 Rev.** を翻訳した日本語版です。)
*A	5217904	AKIH	これは英語版 002-04963 Rev.*A を翻訳した日本語版です。 社名変更と記述フォーマットの変換
*B	5738080	YSAT	これは英語版 002-04963 Rev.*B を翻訳した日本語版です。 Cypress の新ロゴを適用
*C	6182747	NOSU	これは英語版 002-04963 Rev.*C を翻訳した日本語版です。 はじめに 周辺機能の搭載有無についての注意事項を追加 マイコンサポート情報を追加 本マニュアルにおける対象製品 Table 1, 2 の型格を 8 桁表記に修正