



特記事項 : eri FastEthernet デバイスドライバ

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900
U.S.A. 650-960-1300

Part No. 816-1531-10
Revision A, 2001 年 8 月

本製品およびそれに関連する文書は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社の書面による事前の許可なく、本製品および関連する文書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品の一部は、カリフォルニア大学からライセンスされている **Berkeley BSD** システムに基づいていることがあります。UNIX は、**X/Open Company Limited** が独占的にライセンスしている米国ならびに他の国における登録商標です。本製品のフォント技術を含む第三者のソフトウェアは、著作権法により保護されており、提供者からライセンスを受けているものです。

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

本製品は、株式会社モリサワからライセンス供与されたリュウミン L-KL (Ryumin-Light) および中ゴシック BBB (GothicBBB-Medium) のフォント・データを含んでいます。

本製品に含まれる **HG 明朝 L** と **HG ゴシック B** は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。平成明朝体 **W3** は、株式会社リコーが財団法人 日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェイスマスタをもとに作成されたものです。また、**HG 明朝 L** と **HG ゴシック B** の補助漢字部分は、平成明朝体 **W3** の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, Solaris のロゴ、**AnswerBook2, docs.sun.com** は、米国およびその他の国における米国 **Sun Microsystems, Inc.** (以下、米国 **Sun Microsystems** 社とします) の商標もしくは登録商標です。

サン・のロゴマークおよび **Solaris** は、米国 **Sun Microsystems** 社の登録商標です。

すべての **SPARC** 商標は、米国 **SPARC International, Inc.** のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。**SPARC** 商標が付いた製品は、米国 **Sun Microsystems** 社が開発したアーキテクチャーに基づくものです。

Java およびその他の **Java** を含む商標は、米国 **Sun Microsystems** 社の商標であり、同社の **Java** ブランドの技術を使用した製品を指します。

OPENLOOK, OpenBoot, JLE は、サン・マイクロシステムズ株式会社の登録商標です。

ATOK は、株式会社ジャストシステムの登録商標です。**ATOK8** は、株式会社ジャストシステムの著作物であり、**ATOK8** にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。**ATOK Server/ATOK12** は、株式会社ジャストシステムの著作物であり、**ATOK Server/ATOK12** にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

Netscape, Navigator は、米国 **Netscape Communications Corporation** の商標です。**Netscape Communicator** については、以下をご覧ください。Copyright 1995 Netscape Communications Corporation. All rights reserved.

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および **Sun Graphical User Interface** は、米国 **Sun Microsystems** 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 **Sun Microsystems** 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザーインタフェースの概念の研究開発における米国 **Xerox** 社の先駆者としての成果を認めるものです。米国 **Sun Microsystems** 社は米国 **Xerox** 社から **Xerox Graphical User Interface** の非独占的ライセンスを取得しており、このライセンスは米国 **Sun Microsystems** 社のライセンス実施権者にも適用されます。

本書には、技術的な誤りまたは誤植のある可能性があります。また、本書に記載された情報には、定期的に変更が行われ、かかる変更は本書の最新版に反映されます。さらに、米国サンまたは日本サンは、本書に記載された製品またはプログラムを、予告なく改良または変更することがあります。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: *Platform Notes: The eri FastEthernet Device Driver*
Part No: 816-1160-10
Revision A

© 2001 by Sun Microsystems, Inc. 901 SAN ANTONIO ROAD, PALO ALTO CA 94303-4900. All rights reserved.



目次

- はじめに vii
- 1. eri デバイスドライバ 1
 - ハードウェア概要 1
 - 動作速度と動作モード 2
 - 自動ネゴシエーション 2
 - 内部 (ローカル) トランシーバ 3
- 2. Sun eri FastEthernet デバイスドライバに対するドライバソフトウェアの構成 5
 - ホストファイルの構成 5
 - ▼ ホストファイルを構成する 6
 - ネットワークからの起動 7
 - ▼ ネットワークから起動する 8
 - インストール後の作業 (オプション) 8
 - ドライバパラメタの設定 8
- 3. パラメタの定義 11
 - ドライバのパラメタ値と定義 11
 - 現在の状態の定義 13

	パケット間隔 (IPG) パラメタ	13
	lance_modeと ipg0 を使用したパケット送信前の追加遅延の定義	14
	動作モードのパラメタ	15
	送信可能な連続パケット数の定義	16
	トランシーバの機能の報告	16
	接続相手の機能の報告	17

4. パラメタの設定 19

	パラメタのオプション	19
	ndd を使用したパラメタの設定	20
	デバイスインスタンスの特定	20
	非対話モードと対話モード	21
	/etc/system ファイル内のパラメタの設定	24
	▼ 再起動時に ipg1 を 10 に、ipg2 を 5 に設定する	25
	eri.conf ファイルを使用したパラメタの設定	25
	▼ eri.conf ファイルを使用してドライバのパラメタを設定する	26

表目次

表 3-1	eri ドライバのパラメタの状態と説明	11
表 3-2	現在の状態を定義する読み取り専用パラメタ	13
表 3-3	読み取り・書き込み可能なパケット間隔 (IPG) パラメタ	13
表 3-4	lance_mode と ipg0 パラメタの定義	14
表 3-5	動作モードのパラメタ	15
表 3-6	連続パケットの送信制限	16
表 3-7	トランシーバの機能を示す読み取り専用パラメタ	16
表 3-8	接続相手の機能を示す読み取り専用パラメタ	17
表 4-1	/etc/system ファイルに設定する変数	24

はじめに

このマニュアルでは、RIO ASIC(Application Specific Integrated Chip) の Ethernet 機能を使用してサンのシステムの eri ドライバを設定する方法について説明します。

このマニュアルの構成

第 1 章「eri デバイスドライバ」では、eri デバイスドライバの動作速度や動作モード、自動ネゴシエーションについて説明します。

第 2 章「Sun eri FastEthernet デバイスドライバに対するドライバソフトウェアの構成」では、eri デバイスドライバの設定方法について説明します。

第 3 章「パラメタの定義」では、eri デバイスドライバのパラメタについて説明します。

第 4 章「パラメタの設定」では、ndd ユーティリティを使用して eri デバイスドライバのパラメタ値を設定する方法と、`/etc/system` ファイルや `/kernel/drv/eri.conf` ファイルにパラメタを設定する方法について説明します。

UNIX コマンドの使い方

このマニュアルには、基本的な UNIX® のコマンドについての説明も、システムの停止や起動、デバイスの設定などの手順についての説明も含まれていません。

これらの詳細は、以下のマニュアルを参照してください。

- Solaris™ ソフトウェア環境についてのオンラインマニュアル AnswerBook2™
- システムに付属しているその他のソフトウェアマニュアル

書体と記号について

このマニュアルで使用している書体と記号について説明します。

書体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例。	.login ファイルを編集します。 ls -a を実行します。 % You have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して表します。	マシン名% su Password:
AaBbCc123 またはゴシック	コマンド行の可変部分。実際の名前や値と置き換えてください。	rm <i>filename</i> と入力します。 rm ファイル名 と入力します。
『』	参照する書名を示します。	『Solaris ユーザーマニュアル』
「」	参照する章、節、または、強調する語を示します。	第 6 章「データの管理」を参照。 この操作ができるのは「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅をこえる場合に、継続を示します。	% grep `^#define \ XV_VERSION_STRING`

シェルプロンプトについて

シェル	プロンプト
UNIX の C シェル	マシン名%
UNIX の Bourne シェルと Korn シェル	\$
スーパーユーザー (シェルの種類を問わない)	#

第1章

eri デバイスドライバ

eri デバイスドライバは、RIO ASIC を使用してサンのシステム上のSUNW,eriデバイスを制御します。

この章には、以下の節が含まれています。

- 1 ページの「ハードウェア概要」
- 2 ページの「動作速度と動作モード」
- 2 ページの「自動ネゴシエーション」

ハードウェア概要

SUNW,eri デバイスは、RIO ASIC の Ethernet 機能を使用して、100BASE-TX ネットワークインタフェースを提供します。SUNW,eri デバイスドライバは、100BASE-T IEEE 802.3u Ethernet 規格に準拠し、接続の速度を 10 Mbps または 100 Mbps に自動的に設定します。RIO ASIC は、PCI インタフェースと MAC (Media Access Control) 機能を提供します。RJ-45 コネクタに接続する内部トランシーバは、物理層機能を提供します。

RIO ASIC は、1 つの I/O チップと単一チップ Ethernet トランシーバから構成されるチップセットになっており、eri デバイスドライバは、この ASIC の Ethernet 機能を使用します。

動作速度と動作モード

SUNW,eri デバイスを使用して、接続の速度とモードは、以下のどの組み合わせでも選択することができます。

- 100 Mbps、全二重
- 100 Mbps、半二重
- 10 Mbps、全二重
- 10 Mbps、半二重

上記の速度とモードは、100BASE-T IEEE 802.3u Ethernet 規格で定義されています。

自動ネゴシエーション

Sun eri FastEthernet ドライバの主な特長は、自動ネゴシエーション機能です。100BASE-T IEEE 802.3u Ethernet 規格で定義されている自動ネゴシエーションプロトコルでは、自動的に動作モード (半二重または全二重) が選択されます。また自動検出プロトコルは、アダプタの速度 (10 Mbpsまたは100 Mbps) の選択も行います。

自動ネゴシエーションプロトコルは、システムの起動時に以下を行います。

- 接続相手に対応しているすべての動作モードを識別する。
- 接続相手に自分の機能を通知する。
- 以下の優先順位に従って(降順)、最適な共通動作モードを選択する。
 - 100 Mbps、全二重
 - 100 Mbps、半二重
 - 10 Mbps、全二重
 - 10 Mbps、半二重

接続相手とは、接続またはケーブルの反対側に存在するネットワークデバイス (システム、Ethernet ハブ、または Ethernet スイッチ) のことです。

SUNW,eriデバイスが自動ネゴシエーション機能を持たない遠隔システムまたはインタフェースに接続した場合は、システムが自動的に適切な速度と半二重モードを選択します。

Sun eri FastEthernet と、接続相手との間で、自動ネゴシエーションプロトコルの処理が失敗した場合は、このプロトコルを使用しないようにデバイスを設定することができます。この設定を行うと、ドライバはユーザーが選択したモードと速度で接続を確立します。

内部 (ローカル) トランシーバ

Sun eri FastEthernet ドライバは、内部トランシーバに対応しています。内部トランシーバは、この章ですでに述べた「動作速度と動作モード」で示されているすべての速度とモードでの動作が可能です。ドライバは、**100BASE-T IEEE 802.3u Ethernet** 規格に準拠し、接続の速度を**10 Mbps** または **100 Mbps** に自動的に設定します。

また、内部トランシーバは強制動作モードにも対応しています。強制動作モードでは、`ndd` ユーティリティーを使用するか、`/etc/system` ファイルを編集するか、または `/kernel/drv/` ディレクトリに `eri.conf` ファイルを作成することによって、速度とモードを選択します。`ndd` ユーティリティーは `eri` ドライバを呼び出し、速度とモードを選択します。

第2章

Sun eri FastEthernet デバイスドライバに対するドライバソフトウェアの構成

この章では、Sun eri FastEthernet PCI アダプタが使用するドライバソフトウェアの構成に関する情報と手順について説明します。

この章には、以下の節が含まれています。

- 5 ページの「ホストファイルの構成」
- 7 ページの「ネットワークからの起動」
- 8 ページの「インストール後の作業 (オプション)」

ホストファイルの構成

Solaris CD には 64 ビット対応のドライバが含まれています。

ネットワークインタフェースとして eri を使用する前に、以下の説明に従って、システムのホストファイルを作成・編集する必要があります。

▼ ホストファイルを構成する

1. コマンド行で、`grep` コマンドを使用して `eri` デバイスの `/etc/path_to_inst` ファイルを検索します。たとえば、以下ようになります。

```
# grep eri /etc/path_to_inst
"/pci@8,700000/network@5,1" 0 "eri"
```

2. `/etc/hostname.erinum` ファイルを作成します。ここで、`num` は使用する各インタフェースに対応するインスタンス番号です。

手順 1 の例の場合、ネットワークインタフェースを使用するには、以下のファイルを作成する必要があります。

ファイル名	インスタンス番号
<code>/etc/hostname.eri0</code>	0

- 未使用のまま残したい **Sun eri FastEthernet** ネットワークインタフェースに対して、`/etc/hostname.erinum` ファイルは作成しないでください。
- `/etc/hostname.erinum` ファイルには、該当するネットワークインタフェースに対するホスト名を入れる必要があります。
- ホスト名には **IP** アドレスを割り当てる必要があります。これは `/etc/hosts` ファイルに入力する必要があります。
- ホスト名は他のどのインタフェースの他のどのホスト名とも異なる名前にしてください。たとえば、`/etc/hostname.hme0` と `/etc/hostname.eri0` では同じホスト名を共有できません。

以下に例として、手順 1 のインスタンスの例を使用した 2 つの

`/etc/hostname.erinum` ファイル (`zardos`、`zardos-11`) を示します。これらのファイルは、**Sun eri FastEthernet** が実装された `zardoz` という名前のシステムに必要です。

```
# cat /etc/hostname.hme0
zardoz
# cat /etc/hostname.eri0
zardoz-11
```


- 有効となっているそれぞれの eri ネットワークインタフェースについて、`/etc/hosts` ファイルに適切なエントリを作成します。

上記の例を使用すると、以下のようになります。

```
# cat /etc/hosts
#
# Internet host table
#
127.0.0.1    localhost
129.144.10.57 zardoz    loghost
129.144.11.83 zardoz-11
```

注 - IPv6 (インターネットプロトコルバージョン 6) は、現在使用されている IPv4 の機能を拡張したものです。Solaris オペレーティング環境の今回のリリースに含まれる Sun eri FastEthernet デバイスドライバは、IPv4 と IPv6 の両方をサポートしています。IPv4 では、`/etc/hosts` 構成ファイルを使用しますが、IPv6 では異なる構成ファイルを使用します。IPv6 への移行、管理、実装については、『Solaris 8 のシステム管理 (第 3 巻)』を参照してください。

- システムを再起動します。

ネットワークからの起動

Sun eri インタフェースを起動デバイスとして使用するには、以下の作業を実行します。

▼ ネットワークから起動する

1. ok プロンプトで以下のように入力します。

```
ok show-nets
```

show-nets コマンドはシステムデバイスの一覧を示します。以下の例のように、eri デバイスのフルパス名が表示されます。

```
/pci@8,700000/network@5,1
```

注 - これらの eri デバイスから起動用のデバイスを1つだけ選択する必要があります。

2. ok プロンプトで以下のように入力します。

```
ok boot full_path_name_of_the_eri_device
```

インストール後の作業 (オプション)

Sun eri FastEthernet ドライバの性能をカスタマイズするには、以下の作業を実行します。

ドライバパラメタの設定

eri デバイスドライバは、SUNW,eri Ethernet デバイスを制御します。デバイスドライバは、接続相手との間で自動ネゴシエーションプロトコルを使用して、接続の速度を選択します (2 ページの「自動ネゴシエーション」を参照)。

以下の 3 通りの方法のうちのいずれかを使用して、eri デバイスドライバのパラメタを手動で設定し、システムの各 SUNW,eri デバイスをカスタマイズすることができます。

- /kernel/drv ディレクトリに `eri.conf` ファイルを作成することによって、パラメタをデバイスごとに設定します。
- `ndd` ユーティリティーを使用して、パラメタを一時的に変更します。この変更はシステムを再起動すると失われます。
- /etc/system ファイルにパラメタ変数を入力することによって、システムのすべての SUNW,eri デバイスに対して、一括して eri デバイスドライバのパラメタを設定します。

詳細は、第 4 章「パラメタの設定」を参照してください。

▼ ネットワーク速度を 10 Mbps または 100 Mbps にする

1. `ok` プロンプトで `show-nets` コマンドを入力し、システムデバイスの一覧を表示します。

以下の例のように、eri デバイスのフルパス名が表示されます。

```
/pci@8,700000/network@5,1
```

2. 以下のように入力します。

```
ok nvedit
```

3. 0: 行の終わりでリターンキーを押し、以下のように入力します。

```
0: probe-all install-console banner
1: apply transfer-speed=10 full_path_name_of_a_eri_device
```

注 - すでに NVRAM のコマンドが使える場合は、ファイルの最後にこれらの行を追加してください。

4. `full_path_name_of_a_eri_device` を入力してから、Control-C を押します。

手順 2 から 4 の手順を行って、それぞれの eri ネットワークインタフェースのネットワーク速度を設定します。

注 – 前の例では、速度は**10 Mbps** に設定しています。速度を **100 Mbps** に設定する場合は、**10** を **100** に置き換えてください。

5. ok プロンプトで、以下のように入力します。

```
ok nvstore
ok setenv use-nvramrc? true
```

6. システムを再起動します。

ネットワーク速度の設定についての詳細は、**23** ページの「強制モードの設定」を参照してください。

第3章

パラメタの定義

この章では、eri デバイスドライバのパラメタとその設定方法について説明します。

ドライバのパラメタ値と定義

eri ドライバのパラメタの一覧を表 3-1 に示します。

表 3-1 eri ドライバのパラメタの状態と説明

パラメタ	状態	説明
transceiver_inuse	読み取り専用	現在の状態
link_status	読み取り専用	現在の状態
link_speed	読み取り専用	現在の状態
link_mode	読み取り専用	現在の状態
ipg1	読み取り・書き込み可	パケット間の間隔 (IPG) パラメタ
ipg2	読み取り・書き込み可	パケット間の間隔 (IPG) パラメタ
pace_size	読み取り・書き込み可	動作モードのパラメタ
adv_autoneg_cap	読み取り・書き込み可	動作モードのパラメタ
adv_100fdx_cap	読み取り・書き込み可	動作モードのパラメタ
adv_100hdx_cap	読み取り・書き込み可	動作モードのパラメタ
adv_10fdx_cap	読み取り・書き込み可	動作モードのパラメタ
adv_10hdx_cap	読み取り・書き込み可	動作モードのパラメタ

表 3-1 eri ドライバのパラメタの状態と説明 (続き)

パラメタ	状態	説明
autoneg_cap	読み取り専用	ローカルトランシーバの 自動ネゴシエーション機能
100fdx_cap	読み取り専用	ハードウェアの ローカルトランシーバ機能
100hdx_cap	読み取り専用	ハードウェアの ローカルトランシーバ機能
10fdx_cap	読み取り専用	ハードウェアの ローカルトランシーバ機能
10hdx_cap	読み取り専用	ハードウェアの ローカルトランシーバ機能
lp_autoneg_cap	読み取り専用	接続相手の 自動ネゴシエーション機能
lp_100fdx_cap	読み取り専用	接続相手の機能
lp_100hdx_cap	読み取り専用	接続相手の機能
lp_10fdx_cap	読み取り専用	接続相手の機能
lp_10hdx_cap	読み取り専用	接続相手の機能
instance	読み取り・書き込み可	デバイスインスタンス
lance_mode	読み取り・書き込み可	パケット送信前の追加遅延
ipg0	読み取り・書き込み可	パケット送信前の追加遅延

現在の状態の定義

読み取り専用のパラメタのインタフェースの動作モードと現在の状態を表 3-2 に示します。これらのパラメタには、現在の状態が定義されています。

表 3-2 現在の状態を定義する読み取り専用パラメタ

パラメタ	説明	値
link_status	現在の接続状態	0 = 切断 1 = 接続
link_speed	接続状態のときのみ有効	0 = 10 Mbps 1 = 100 Mbps
link_mode	接続状態のときのみ有効	0 = 半二重 1 = 全二重

パケット間隔 (IPG) パラメタ

RIO ASIC の Ethernet 機能ユニットは、プログラム可能なパケット間の間隔 (IPG:Inter-Packet Gap) パラメタである ipg1 と ipg2 に対応しています。IPG の合計値は、ipg1 と ipg2 の合計値になります。自動ネゴシエーションプロトコルによって接続速度が 10 Mbps に設定されている場合、IPG の合計は 9.6 マイクロ秒になります。接続速度が 100 Mbps の場合は、0.96 マイクロ秒になります。

IPG パラメタの ipg1 および ipg2 のデフォルト値と許容値を表 3-3 に示します。

表 3-3 読み取り・書き込み可能なパケット間隔 (IPG) パラメタ

パラメタ	値 (バイト時間)	説明
ipg1	0~255	ipg1 = 8 (初期化時のデフォルト値)
ipg2	0~255	ipg2 = 4 (初期化時のデフォルト値)

デフォルトの設定では、ipg1 が 8 バイト時間に設定され、ipg2 が 4 バイト時間に設定されます。これらはともに標準値です。バイト時間は、100 Mbps または 10 Mbps の接続速度で 1 バイトを送信するのに要する時間です。

これより長い IPG (ipg1 と ipg2 の合計) を使用するシステムがネットワークに存在し、それらのマシンによるネットワークへのアクセスが遅い場合は、長い IPG に一致するように他のマシンの ipg1 と ipg2 の値を大きくしてください。

lance_modeと ipg0 を使用したパケット送信前の追加遅延の定義

RIO ASIC の Ethernet 機能ユニットは、lance_mode という、プログラム可能なモードに対応しています。ipg0 は、lance_mode に関するパラメタです。

lance_mode が有効(デフォルトの設定)な場合は、ipg0 パラメタを設定することによって、パケットを受信してから送信するまでの間に遅延が追加されます。ipg0 パラメタによって設定されるこの遅延は、ipg1 と ipg2 パラメタによって設定される遅延に追加されます。IpG0 で追加遅延を設定することによって、衝突を減少させることができます。ただし、lance_mode を有効にしたシステムでは、ネットワーク上で時間が不足する可能性があります。

lance_mode が無効な場合は、ips0 の値は無視され、追加遅延は設定されません。この場合は、ipg1 と ipg2 によって設定された遅延だけが使用されます。他のシステムから大量の連続したパケットが送信され続けている場合は、lance_mode を無効にしてください。

追加遅延は、ipg0 パラメタに 0 ~ 31 の値を設定することによって設定することができます。これは、ニブル時間で表した遅延です。ニブル時間は、接続上で 4 ビットを送信するために必要な時間です。接続速度が 10 Mbps の場合、ニブル時間は 400 ナノ秒となります。接続速度が 100 Mbps の場合は、40 ナノ秒となります。

たとえば、接続の速度が 10 Mbps で、ipg0 を 20 ニブル時間に設定した場合、追加遅延は 20×400 ナノ秒で 8000 ナノ秒になります。接続速度が 100 Mbps で、ipg0 を 30 ニブル時間に設定した場合は、 30×40 ナノ秒で 1200 ナノ秒になります。

lance_mode と ipg0 のパラメタの定義を表 3-4 に示します。

表 3-4 lance_mode と ipg0 パラメタの定義

パラメタ	値	説明
lance_mode	0	lance_mode 無効
	1	lance_mode 有効 (デフォルト)
ipg0	0~31*	パケット送信前に IPG を追加 (1 パケット受信後)

* デフォルト値は 16 ニブル時間です。10 Mbps の場合は 6.4 マイクロ秒、100 Mbps の場合は 0.64 マイクロ秒になります。

動作モードのパラメタ

動作モードのパラメタとそのデフォルト値を表 3-5 に示します。

表 3-5 動作モードのパラメタ

パラメタ	説明	値
adv_autoneg_cap	ハードウェアが通知するローカルトランシーバの機能	0 = 強制モード 1 = 自動ネゴシエーション (デフォルト)
adv_100fdx_cap*	ハードウェアが通知するローカルトランシーバの機能 (読み取り・書き込みパラメタ)	0 = 100 Mbps、全二重不可 1 = 100 Mbps、全二重可 (デフォルト)
adv_100hdx_cap*	ハードウェアが通知するローカルトランシーバの機能 (読み取り・書き込みパラメタ)	0 = 100 Mbps、半二重不可 1 = 100 Mbps、全二重可 (デフォルト)
adv_10fdx_cap*	ハードウェアが通知するローカルトランシーバの機能 (読み取り・書き込みパラメタ)	0 = 10 Mbps、全二重不可 1 = 10 Mbps、全二重可 (デフォルト)
adv_10hdx_cap*	ハードウェアが通知するローカルトランシーバの機能 (読み取り・書き込みパラメタ)	0 = 10 Mbps、半二重不可 1 = 10 Mbps、半二重可 (デフォルト)

* これらのパラメタの優先順位は、値の高いものから順に adv_100fdx_cap、adv_100hdx_cap、adv_10fdx_cap、adv_10hdx_cap となります。

送信可能な連続パケット数の定義

pace_size パラメタ (表 3-6 を参照) は、一度に送信することができる連続パケットの最大数を定義します。pace_size の値が 0 の場合は、送信することができる連続パケットの数に制限はありません。

表 3-6 連続パケットの送信制限

パラメタ	値	説明
pace_size	1~255	一度に送信できる連続パケット数
	0	一度に送信できる連続パケット数の制限なし (デフォルト)

トランシーバの機能の報告

読み取り専用トランシーバの機能を表 3-7 に示します。これらのパラメタは、ハードウェアの機能を定義します。ローカルトランシーバはこれらの機能のすべてに対応します。

表 3-7 トランシーバの機能を示す読み取り専用パラメタ

パラメタ	説明	値
autoneg_cap	ハードウェアのローカルトランシーバ機能	0 = 自動ネゴシエーション不可 1 = 自動ネゴシエーション可
100fdx_cap	ハードウェアのローカルトランシーバ機能 (起動時に初期化)	0 = 100 Mbps、全二重不可 1 = 100 Mbps、全二重可
100hdx_cap	ハードウェアのローカルトランシーバ機能 (起動時に初期化)	0 = 100 Mbps、半二重不可 1 = 100 Mbps、半二重可
10fdx_cap	ハードウェアのローカルトランシーバ機能 (起動時に初期化)	0 = 10 Mbps、全二重不可 1 = 10 Mbps、全二重可
10hdx_cap	ハードウェアのローカルトランシーバ機能 (起動時に初期化)	0 = 10 Mbps、半二重不可 1 = 10 Mbps、半二重可

接続相手の機能の報告

接続相手の機能を示す読み取り専用パラメタを表 3-8 に示します。

表 3-8 接続相手の機能を示す読み取り専用パラメタ

パラメタ	値	説明
lp_autoneg_cap	0=	自動ネゴシエーション不可
	1=	自動ネゴシエーション可
lp_100fdx_cap	0=	100 Mbps、全二重送信不可
	1=	100 Mbps、全二重送信可
lp_100hdx_cap	0=	100 Mbps、半二重送信不可
	1=	100 Mbps、半二重送信可
lp_10fdx_cap	0=	10 Mbps、全二重送信不可
	1=	10 Mbps、全二重送信可
lp_10hdx_cap	0=	10 Mbps、半二重送信不可
	1=	10 Mbps、半二重送信可

接続相手が自動ネゴシエーションを行うことができない場合 (lp_autoneg_cap が 0 の場合) は、パラメタの値は 0 となり、表 3-8 の情報は適用されません。

接続相手が自動ネゴシエーションを行うことができる場合 (lp_autoneg_cap が 1 の場合) は、自動ネゴシエーションを使用して接続相手の機能を調べることによって、動作速度と動作モードの情報を表示することができます。

第4章

パラメタの設定

この章では、eri デバイスドライバのパラメタを構成する方法について説明します。ndd ユーティリティを使用して構成したパラメタは、システムを再起動すると無効になります。

再起動した後もパラメタ値を常に有効にし、システム内のすべてのデバイスに eri ドライバのパラメタを構成するには、`/etc/system` ファイルにパラメタ値を入力します。これによりシステムの再起動時に `/etc/system` ファイルがシステムに読み込まれ、このファイル内のパラメタ値が設定されます。

システム内の特定のデバイスに対してパラメタを設定するには、`/kernel/drv` ディレクトリの `eri.conf` ファイルにパラメタを設定します。eri.conf ファイルで設定したパラメタは、`/etc/system` ファイルで設定したパラメタよりも優先され、`/etc/system` ファイルで設定したパラメタを無効にします。eri.conf ファイルで設定したパラメタ値は、再起動しても常に有効となります。

パラメタのオプション

eri デバイスドライバのパラメタを設定する方法は 3 通りあり (ndd、`/etc/system`、eri.conf)、必要に応じて使い分けることができます。システムを再起動するまで有効となるパラメタを設定するには、ndd ユーティリティを使用します。この方法は、パラメタの設定を試してみる際に有用です。

システムを再起動した後も有効となるパラメタ値を設定する方法を以下に示します。

- システム内のすべてのデバイスに対するパラメタ値を設定するには、`/etc/system` ファイルにパラメタ値を追加します。

- システム内の特定のデバイスに対するパラメタ値を設定するには、`eri.conf`ファイルを作成し、このファイルにパラメタ値を追加します。

パラメタの設定を確認するには、`ndd` ユーティリティーを使用します（「`ndd`を使用したパラメタの設定」を参照）。`ndd` ユーティリティーを使用して設定したパラメタは、システムを再起動すると無効になります。パラメタ値を、再起動をした後でも有効にするには、`/etc/system` ファイルまたは `eri.conf` ファイルに値を設定します。設定方法についてはこの章で説明します。

ndd を使用したパラメタの設定

システムを再起動するまで有効となるパラメタを設定するには、`ndd` ユーティリティーを使用します。`ndd` ユーティリティーは、DLPI (Data Link Provider Interface) が実装されているすべてのネットワークドライバに対応しています。

以下の節では、`eri` ドライバと `ndd` ユーティリティーを使用して、それぞれの `SUNW,eri` デバイスのパラメタを変更 (`-set` オプションを使用) または表示 (`-set` オプションを使用しない) する方法を説明します。

デバイスインスタンスの特定

`SUNW,eri` デバイスが複数ある場合は、`ndd` ユーティリティーを使用して `eri` デバイスのパラメタを取得または設定する際に、このユーティリティーのデバイスインスタンスを指定する必要があります。

注 - `SUNW,eri` デバイスが 1 つのみの場合は、`ndd` ユーティリティーによって自動的にこのデバイスが選択されます。

▼ `ndd` ユーティリティーにデバイスインスタンスを指定する

1. `/etc/path_to_inst` ファイルを調べて、目的のデバイスのインスタンスを特定します。

2. インスタンス番号を使用して、目的のデバイスを選択します。

```
% ndd -set /dev/eri instance インスタンス番号
```

選択したデバイスは、別のデバイスを選択するまで有効となります。

非対話モードと対話モード

ndd ユーティリティーは、以下の 2 種類のモードで使用することができます。

- 非対話モード
- 対話モード

非対話モードでは、ndd ユーティリティーを呼び出して特定のコマンドを実行します。コマンドが実行されると、ユーティリティーは終了します。対話モードでは、ndd ユーティリティーを使用して複数のパラメタ値を取得または設定することができます。詳細は、ndd(1M) のマニュアルのページを参照してください。

非対話モードで ndd ユーティリティーを使用する

- パラメタ値を変更するには、-set オプションを使用します。

-set オプションを指定して ndd ユーティリティーを呼び出すと、ユーティリティーは指定された値をドライバに引き渡し、この値をパラメタに割り当てます。値には、/dev/eri ドライバインスタンスを、名称まで含めて指定します。

```
% ndd -set /dev/eri パラメタ値
```

- パラメタ値を表示するには、パラメタ名だけを指定し、値を省略します。

-set オプションを省略すると、照会とみなされます。ndd ユーティリティーは指定されたドライバインスタンスを照会し、指定されたパラメタの値を取り出し、表示します。

```
% ndd /dev/eri パラメタ名
```

対話モードで ndd ユーティリティーを使用する

- 対話モードでパラメタ値を変更するには、以下のように ndd eri を指定します。
ndd ユーティリティーは、パラメタ名の入力を要求するプロンプトを表示します。

```
% ndd /dev/eri
name to get/set? (パラメタ名を入力します。
                  ? を入力するとパラメタの一覧が表示されます。)
```

パラメタ名を入力すると、ndd ユーティリティーはパラメタ値の入力を要求するプロンプトを表示します。表 3-1 から表 3-8 を参照してください。

- eri ドライバで使用することができるすべてのパラメタを表示するには、以下のように入力します。

```
% ndd /dev/eri \?
```

(パラメタの詳細は、表 3-1 ~ 表 3-8 を参照してください。)

コード例 4-1 eri ドライバが対応しているすべてのパラメタの一覧表示

```
example# ndd /dev/eri \?
? (read only)
transceiver_inuse (read only)
link_status (read only)
link_speed (read only)
link_mode (read only)
ipg1 (read and write)
ipg2 (read and write)
use_int_xcvr (read and write)
pace_size (read and write)
adv_autoneg_cap (read and write)
adv_100fdx_cap (read and write)
adv_100hdx_cap (read and write)
adv_10fdx_cap (read and write)
adv_10hdx_cap (read and write)
autoneg_cap (read only)
100T4_cap (read only)
100fdx_cap (read only)
100hdx_cap (read only)
10fdx_cap (read only)
10hdx_cap (read only)
lp_autoneg_cap (read only)
```


コード例 4-1 eri ドライバが対応しているすべてのパラメタの一覧表示 (続き)

lp_100fdx_cap	(read only)
lp_100hdx_cap	(read only)
lp_10fdx_cap	(read only)
lp_10hdx_cap	(read only)
instance	(read and write)
lance_mode	(read and write)
ipg0	(read and write)
example#	

強制モードの設定

強制モード (自動ネゴシエーション不可) を設定する方法を説明します。

▼ ローカルトランシーバの機能から 1つを選択して、強制モードに設定する。

1. 4 つある機能 (adv_100fdx_cap、adv_100hdx_cap、adv_10fdx_cap、adv_10hdx_cap) から 1 つを選択し、その値を 1 に設定します。

ローカルトランシーバの機能を複数選択した場合は、優先順位の最も高い機能が選択されます。

2. ハードウェアが通知するローカルトランシーバの機能として、自動ネゴシエーション不可を示す強制モード=0を設定します (adv_autoneg_cap 0)。

この章の「対話モードで ndd ユーティリティを使用する」の説明に従って、nnd ユーティリティを使用してください。

自動ネゴシエーションモード

▼ 自動ネゴシエーションモードに設定する

1. 遠隔システムに通知する機能として、4 つある機能 (adv_100fdx_cap、adv_100hdx_cap、adv_10fdx_cap、adv_10hdx_cap) から 1 つ以上を選択し、その値を 1 に設定します。

- ハードウェアが通知するローカルトランシーバの機能に対して、自動ネゴシエーション可を示す 1 を設定します (adv_autoneg_cap 1)。

この章の 22 ページの「対話モードで ndd ユーティリティーを使用する」の説明に従って、nnd ユーティリティーを使用してください。

/etc/system ファイル内のパラメタの設定

システムのすべての SUNW,eri デバイスに対して eri ドライバパラメタを設定し、かつシステムの再起動後もパラメタ変数が常に有効になるようにするには、
/etc/system ファイルにパラメタ変数を入力します。システムを再起動すると、
/etc/system ファイルが読み取られ、オペレーティングシステムのカーネルの eri モジュールにパラメタ変数が設定されます。

/etc/system ファイルに設定する変数を表 4-1 に示します。

表 4-1 /etc/system ファイルに設定する変数

パラメタ	変数
ipg1	ipg1
ipg2	ipg2
pace_size	pace_size
adv_autoneg_cap	adv_autoneg_cap
adv_100fdx_cap	adv_100fdx_cap
adv_100hdx_cap	adv_100hdx_cap
adv_10fdx_cap	adv_10fdx_cap
adv_10hdx_cap	adv_10hdx_cap
lance_mode	lance_mode
ipg0	ipg0

第 3 章で説明したこれらのパラメタ値は、システム上のすべての SUNW,eri デバイスに適用することができます。パラメタの説明については、表 3-2 ~ 表 3-8 を参照してください。以下に /etc/system ファイルに設定する例を示します。

▼ 再起動時に ipg1 を 10 に、ipg2 を 5 に設定する

1. スーパーユーザーになります。
2. テキストエディタを使用し、`/etc/system` ファイルに以下の行を追加します。

```
set eri:ipg1 = 10
set eri:ipg2 = 5
```

3. `/etc/system` ファイルを保存します。
4. すべてのファイルを保存し、すべてのプログラムを終了します。そして、ウィンドウシステムを終了します。
5. # プロンプトに対して `init 6` と入力し、システムを再起動します。
システムが停止して再起動されます。

eri.conf ファイルを使用したパラメタの設定

この章の「`/etc/system` ファイル内のパラメタの設定」で説明しているパラメタは、デバイスごとに指定することもできます。それらのパラメタをデバイスごとに設定するには、`/kernel/drv` ディレクトリに `eri.conf` ファイルを作成します。`eri.conf` ファイルのパラメタの設定は、`/etc/system` ファイルの設定に優先します。システムの特定のデバイスに対して特定のパラメタを設定する必要がある場合は、`eri.conf` ファイルを使用してください。`eri.conf` ファイルには、第 3 章に示す読み取り・書き込み可能なパラメタを設定することができます。

詳細は、`prtconf (1M)`、`system (4)`、`driver.conf (4)` のマニュアルのページを参照してください。以下に `eri.conf` ファイルを設定する例を示します。

▼ eri.conf ファイルを使用してドライバのパラメータを設定する

1. デバイスツリーでデバイスのハードウェアパス名を取得します。

通常、デバイスのパス名と関連するインスタンス番号は、`/etc/path_to_inst` ファイルに含まれています。たとえば、**Sun Blade 1000 PCI** システムでは、`/etc/path_to_inst` ファイル内に以下のエントリが存在します。

```
"/pci@8,700000/network@5,1" 0 "eri"
```

- このエントリの詳細は以下のとおりです。
 - 最初の二重引用符内は、デバイスツリー内のハードウェアノード名を表します。
 - 2番目の番号は、インスタンス番号です。
 - 最後の二重引用符内はドライバ名です。
- デバイスパス名の、最後の `/` 文字と `@` 文字の間の最後の要素は、デバイス名を表します。
- 最後の要素の前のパス名は、親の名前を表します。
- 最後の `@` 文字の後のコンマで区切られた番号は、デバイス番号と機能番号を表し、この2つで装置アドレスと呼ばれます。

`eri.conf` ファイルで PCI デバイスを明確に指定するには、デバイス名と親、装置アドレスを使用します。PCI デバイス指定の詳細については、`pci (4)` のマニュアルページを参照してください。

上記の例の 1 行目では、以下ようになります。

- 名前 = `eri`
- 親 = `/pci@8,700000`
- 装置アドレス = `5,1`

2. `/kernel/drv/eri.conf` ファイル内の上記のデバイスに、`ipg1` および `ipg2` のパラメータを設定します。

```
name = "eri" parent = "/pci@8,700000" unit-address = "5,1" ipg1=10 ipg2=5;
```