



# 特記事項: Sun GigaSwift Ethernet デバイスドライバ

---

Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

Part No. 816-4664-10  
2002 年 5 月, Revision A

コメントの宛先: [docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054 U.S.A. All rights reserved.

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。

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、AnswerBook2、OpenBoot、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。

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。

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。

OPENLOOK 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。

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

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本書には、技術的な誤りまたは誤植の可能性があります。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:	<i>Platform Notes: Sun GigaSwift Ethernet Device Driver</i>
	Part No: 816-2351-10
	Revision A



# 目次

---

1.	ドライバパラメタの設定	1
	GigaSwift Ethernet デバイスドライバのパラメタ	1
	ドライバのパラメタ値と定義	2
	動作モードのパラメタ	3
	フロー制御パラメタ	4
	ギガビット強制モードのパラメタ	5
	パケット間隔 (IPG) パラメタ	6
	割り込みパラメタ	7
	ランダム早期ドロップのパラメタ	8
	PCI バスインタフェースのパラメタ	9
ce	ドライバパラメタの設定	9
	nndd ユーティリティを使用したパラメタの設定	10
	▼ nndd ユーティリティにデバイスインスタンスを指定する	10
	非対話モードと対話モード	10
	自動ネゴシエーションモードの設定	12
	▼ 自動ネゴシエーションモードを無効にする	13
	ce.conf ファイルを使用したパラメタの設定	13
	▼ ce.conf ファイルを使用して、ドライバパラメタを設定する	14
	GigaSwift Ethernet ドライバの動作の統計情報	16
	接続相手の機能の報告	17

- ▼ 接続相手の設定を確認する 20
- ネットワークホストファイルの設定 20

# はじめに

---

このマニュアルでは、Sun™ GigaSwift™ Ethernet デバイスドライバで使用されるソフトウェアを設定する方法について説明します。また、ネットワークを設定する情報についても取り上げます。

---

## 関連マニュアル

タイトル	Part No.
『Solaris 9 Sun ハードウェアマニュアル』	816-4622-10

---

## 書体と記号について

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。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を実行します。 <code>% You have mail.</code>
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% <code>su</code> 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。	<code>rm filename</code> と入力します。 <code>rm ファイル名</code> 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。	% <code>grep `^#define` \</code> <code>XV_VERSION_STRING'</code>

---

## 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)	#

---

## コメントをお寄せください

弊社では、マニュアルの改善に努力しており、お客様からのコメントおよびご忠告をお受けしております。コメントは下記宛に電子メールでお送りください。

[docfeedback@sun.com](mailto:docfeedback@sun.com)

電子メールの表題にはマニュアルの Part No. (816-4664-10) を記載してください。

なお、現在日本語によるコメントには対応できませんので、英語で記述してください。





# ドライバパラメタの設定

---

この章では、Sun GigaSwift Ethernet アダプタで使用されているドライバパラメタの設定方法について説明します。

このマニュアルの内容は次のとおりです。

- GigaSwift Ethernet デバイスドライバのパラメタ
- ce ドライバパラメタの設定
- GigaSwift Ethernet ドライバの動作の統計情報
- ネットワークホストファイルの設定

---

注 – GigaSwift Ethernet ドライバは、Virtual Local Area Networks (VLAN) をサポートしています。VLAN パッケージは、Solaris ソフトウェアの一部として自動的にインストールされます。ただし、VLAN を使用するには、VLAN を設定をする必要があります。詳細は、『Sun ハードウェアマニュアル』を参照してください。

---

---

## GigaSwift Ethernet デバイスドライバのパラメタ

ce デバイスドライバは、GigaSwift Ethernet デバイスを制御します。ce ドライバは、Sun GigaSwift Ethernet アダプタの UNIX pci 名前プロパティ pci108e, abba に接続されます (108e はベンダー ID で、abba は PCI デバイス ID です)。

ce デバイスドライバのパラメタを手動で設定して、システムの各 Sun GigaSwift Ethernet アダプタデバイスをカスタマイズできます。ここでは、アダプタで使用される GigaSwift Ethernet デバイスの機能の概要、利用できる ce デバイスドライバのパラメタ、およびこのパラメタの設定方法について説明します。

Sun GigaSwift Ethernet UTP PCI アダプタは、12 ページの「自動ネゴシエーションモードの設定」で示すすべての動作速度と動作モードで機能します。ce デバイスは、接続の遠隔地側 (接続相手) と自動ネゴシエーションを行い、共通の動作モードを選択します。また、ce デバイスは強制動作モードにも対応しています。

## ドライバのパラメタ値と定義

次の表は、ce デバイスドライバのパラメタおよび設定の一覧です。

表 1 ce ドライバパラメタの状態と説明

パラメタ	状態	説明
instance	読み取り・書き込み可	デバイスインスタンス
adv_autoneg_cap	読み取り・書き込み可	動作モードのパラメタ
adv_1000fdx_cap	読み取り・書き込み可	動作モードのパラメタ
adv_1000hdx_cap	読み取り・書き込み可	動作モードのパラメタ
adv_100T4_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
adv_asmpause_cap	読み取り・書き込み可	フロー制御パラメタ
adv_pause_cap	読み取り・書き込み可	フロー制御パラメタ
link_master	読み取り・書き込み可	1 ギガビット強制モードのパラメタ
use_int_xcvr	読み取り・書き込み可	
enable_ipg0	読み取り・書き込み可	パケット送信前の追加遅延を有効にする
ipg0	読み取り・書き込み可	パケット送信前の追加遅延
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
rx_intr_pkts	読み取り・書き込み可	割り込みブランキング値の受信
rx_intr_time	読み取り・書き込み可	割り込みブランキング値の受信
red_dv4to6k	読み取り・書き込み可	ランダム早期検出とパケットドロップベクトル

表 1 ce ドライバパラメタの状態と説明 (続き)

パラメタ	状態	説明
red_dv6to8k	読み取り・書き込み可	ランダム早期検出とパケットドロップベクトル
red_dv8to10k	読み取り・書き込み可	ランダム早期検出とパケットドロップベクトル
red_dv10to12k	読み取り・書き込み可	ランダム早期検出とパケットドロップベクトル
tx_dma_weight	読み取り・書き込み可	PCI インタフェースのパラメタ
rx_dma_weight	読み取り・書き込み可	PCI インタフェースのパラメタ
infinite_burst	読み取り・書き込み可	PCI インタフェースのパラメタ
disable_64bit	読み取り・書き込み可	PCI インタフェースのパラメタ

## 動作モードのパラメタ

次のパラメタでは、送信速度、受信速度、および二重方式の種類を指定します。次の表は、動作モードのパラメタとデフォルト値の一覧です。

表 2 動作モードのパラメタ

パラメタ	値と説明
adv_autoneg_cap	ハードウェアが通知するローカルインタフェースの機能 0 = 強制モード 1 = 自動ネゴシエーション (デフォルト)
adv_1000fdx_cap	ハードウェアが通知するローカルインタフェースの機能 0 = 1000 Mbps、全二重不可 1 = 1000 Mbps、全二重可 (デフォルト)
adv_1000hdx_cap	ハードウェアが通知するローカルインタフェースの機能 0 = 1000 Mbps、半二重不可 1 = 1000 Mbps、半二重可 (デフォルト)
adv_100fdx_cap	ハードウェアが通知するローカルインタフェースの機能 0 = 100 Mbps、全二重不可 1 = 100 Mbps、全二重可 (デフォルト)

表 2 動作モードのパラメタ (続き)

パラメタ	値と説明
adv_100hdx_cap	ハードウェアが通知するローカルインタフェースの機能 0 = 100 Mbps、半二重不可 1 = 100 Mbps、半二重可 (デフォルト)
adv_10fdx_cap	ハードウェアが通知するローカルインタフェースの機能 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インタフェースの機能 0 = 10 Mbps、半二重不可 1 = 10 Mbps、半二重可 (デフォルト)

注 - 初期設定が 0 のパラメタは変更できません。変更しても 0 に戻ります。

これらのパラメタがすべて 1 に設定されている場合、自動ネゴシエーションでは使用可能な最大の速度が使用されます。これらのパラメタがすべて 0 に設定されている場合、次のエラーメッセージが表示されます。

```
NOTICE: Last setting will leave ce1 with no link capabilities.
WARNING: ce1: Restoring previous setting.
```

## フロー制御パラメタ

ce デバイスは、IEEE 802.3x フレームのリンクレベルフロー制御プロトコル準拠の一時停止 (PAUSE) フレームの送信 (ソーシング) と受信 (ターミネーティング) に対応しています。フロー制御フレームを受信した ce デバイスは、転送レートを落すことができます。一方、ce デバイスは、フロー制御フレームを接続相手に送信して、接続相手がこの機能をサポートしている場合は、転送レートを落すように依頼します。デフォルトでは、自動ネゴシエーションの間のみ、送信と受信の両方で一時停止機能が有効となるようにドライバを設定してあります。

次の表は、フロー制御キーワードおよびその機能の一覧です。

表 3 読み取り・書き込み可能なフロー制御キーワードの説明

キーワード	説明
adv_asmpause_cap	アダプタは、一方向だけを一時停止させる非同期一時停止をサポートします。 0 = オフ (デフォルト) 1 = オン
adv_pause_cap	このパラメタには、adv_asmpause_cap の値に応じて 2 つの意味があります (デフォルト = 0)。 adv_asmpause_cap = 1 で、dv_pause_cap = 1 の場合、一時停止を受信します。 adv_asmpause_cap = 1 で、dv_pause_cap = 0 の場合、一時停止を送信します。 adv_asmpause_cap = 0 で、dv_pause_cap = 1 の場合、一時停止は送信も受信もされません。 adv_asmpause_cap = 0 の場合、dv_pause_cap は、一時停止機能を有効または無効のいずれかに指定します。

## ギガビット強制モードのパラメタ

10/100 モードでは、自動ネゴシエーションを無効にし、接続を希望する速度に強制的に設定できます。ギガビット銅線インタフェースでもこの機能は利用できますが、接続のマスタとスレーブのいずれにするかを指定する必要があります。これは、link\_master パラメタで指定します。通常スイッチをマスタとして有効にする場合、このパラメタはそのままにします (スレーブ操作がデフォルトのため)。これ以外の場合は、link\_master を使用すると、ce をマスタとして有効にできます。

表 4 強制モードのパラメタ

パラメタ	説明
link_master	1 に設定するとマスタ動作が有効になります。ただし、接続相手がスレーブであることを前提としています。 0 に設定するとスレーブ動作が有効になります。ただし、接続相手がスレーブであることを前提としています (デフォルト)。

## パケット間隔 (IPG) パラメタ

ce デバイスは、enable\_ipg0 というプログラム可能なモードをサポートしています。

enable\_ipg0 が設定された (デフォルト) パケットをドライバが受信すると、そのパケットを送信する前に追加の遅延時間を付加します。ipg0 パラメタで設定されるこの遅延時間は、ipg1 パラメタと ipg2 パラメタで設定される遅延に追加されます。追加の ipg0 遅延によって、衝突を減少させることができます。

enable\_ipg0 が無効の場合、ipg0 の値は無視され、追加の遅延は設定されません。ipg1 と ipg2 によって設定された遅延だけが使用されます。他のシステムが大量の連続パケットを送信し続ける場合、enable\_ipg0 を無効にします。enable\_ipg0 が設定されているシステムでは、ネットワーク上で時間が不足する可能性があります。

メディアバイト時間遅延である ipg0 パラメタを 0 ~ 255 の間で設定すると、追加遅延を設定できます。

次の表は、enable\_ipg0 パラメタと ipg0 パラメタの一覧です。

表 5 enable\_ipg0 と ipg0 パラメタの定義

パラメタ	値	説明
enable_ipg0	0	enable_ipg0 をリセット
	1	enable_ipg0 を設定 (デフォルト = 8)
ipg0	0 ~ 255	パケットを受信してから送信するまでの遅延時間 (間隔) (デフォルト = 8)

ce デバイスは、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。接続速度が 1,000 Mbps の場合、IPG の合計値は 0.096 マイクロ秒になります。

次の表は、IPG パラメタのデフォルト値と許容値の一覧です。

表 6 読み取り・書き込み可能なパケット間隔 (IPG) パラメタの値と説明

パラメタ	値 (バイト時間)	説明
ipg1	0 ~ 255	パケット間隔 1 (デフォルト = 8)
ipg2	0 ~ 255	パケット間隔 2 (デフォルト = 4)

デフォルトでは、ipg1 が 8 バイト時間、ipg2 が 4 バイト時間に設定されます。いずれも標準値です (バイト時間とは、接続速度が 1000 Mbps の場合に 1 バイトを送信するのにかかる時間です)。

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、ネットワークへのアクセスが遅く感じられる場合、他のマシンの長い IPG に一致するように ipg1 と ipg2 の値を大きくします。

## 割り込みパラメタ

次の表は、受信割り込みブランキング値です。

表 7 エイリアス読み取り用 RX ブランキングレジスタ

フィールド名	値	説明
rx_intr_pkts	0 ~ 511	最後のパケットの処理後、指定した数のパケットを受信したら割り込みます。0 パケットによるブランキングが行われないことを示します (デフォルト = 3)。
rx_intr_time	0 ~ 524287	最後のパケットの処理後、4.5 マイクロ秒のパルス刻みが経過してから割り込みます。0 という値は時間によるブランキングが行われないことを示します (デフォルト = 1250)。

## ランダム早期ドロップのパラメタ

次の表は、RX ランダム早期検出 8 ビットベクトルの説明です。これを使用すると、ランダム早期ドロップ (RED) のしきい値を有効にできます。受信パケットが RED 範囲に達すると、あらかじめ設定された頻度に従ってドロップされます。FIFO のレベルが上がったときは、設定頻度も増やす必要があります。制御パケットはドロップされず、統計にもカウントされません。

表 8 RX ランダム早期検出 8 ビットベクトル

フィールド名	値	説明
red_dv4to6k	0 ~ 255	FIFO のしきい値が 4,096 バイトより大きく 6,144 バイトより小さい場合のランダム早期検出とパケットドロップベクトルです。ドロップの頻度は、12.5% の粒度 (granularity) でプログラムできます。たとえば、ビット 0 を設定すると、8 パケットのうち最初のパケットがこの範囲でドロップされます (デフォルト = 0)。
red_dv6to8k	0 ~ 255	FIFO のしきい値が 6,144 バイトより大きく 8,192 バイトより小さい場合のランダム早期検出とパケットドロップベクトルです。ドロップの頻度は、12.5% の粒度 (granularity) でプログラムできます。たとえば、ビット 8 を設定すると、8 パケットのうち最初のパケットがこの範囲でドロップされます (デフォルト = 0)。
red_dv8to10k	0 ~ 255	FIFO のしきい値が 8,192 バイトより大きく 10,240 バイトより小さい場合のランダム早期検出とパケットドロップベクトルです。ドロップの頻度は、12.5% の粒度 (granularity) でプログラムできます。たとえば、ビット 16 を設定すると、8 パケットのうち最初のパケットがこの範囲でドロップされます (デフォルト = 0)。
red_dv10to12k	0 ~ 255	FIFO のしきい値が 10,240 バイトより大きく 12,288 バイトより小さい場合のランダム早期検出とパケットドロップベクトルです。ドロップの頻度は、12.5% の粒度 (granularity) でプログラムできます。たとえば、ビット 24 を設定すると、8 パケットのうち最初のパケットがこの範囲でドロップされます (デフォルト = 0)。



## PCI バスインタフェースのパラメタ

このパラメタを使用すると、PCI インタフェース機能を変更して、特定のアプリケーションに対する PCI インタフェースのパフォーマンスを向上できます。

表 9 PCI バスインタフェースのパラメタ

パラメタ	説明
tx_dma_weight	加重ラウンドロビン調停のときに TX 側に対して考慮する乗数を指定します。値は 0 ~ 3 です (デフォルト = 0)。0 は加重しないことを示しています。他の値を指定すると、このトラフィックで 2 のべき乗の割り増し加重となります。たとえば、tx_dma_weight = 0 で、rx_dma_weight = 3 の場合、RX トラフィックが連続して到着している限り、PCI にアクセスする優先順位は TX の 8 倍以上となります。
rx_dma_weight	加重ラウンドロビン調停のときに RX 側に考慮するための乗数を指定します。値は 0 ~ 3 です (デフォルト = 0)。
infinite_burst	無限バースト機能が利用できます。このパラメタが有効の場合、システムは無限バーストをサポートします。全部の packets がバス上で転送終了されるまでアダプタはバスを解放しません。値は 0 または 1 です (デフォルト = 0)。
disable_64bit	アダプタの 64 ビット機能を無効にします。場合によっては、この機能を無効にする方が便利です。値は 0 または 1 です (デフォルト = 0、64 ビット機能は有効です)。

## ce ドライバパラメタの設定

ce デバイスドライバのパラメタを設定するには、次の 2 通りの方法があります。

- ndd ユーティリティを使用する
- ce.conf ファイルを使用する

ndd ユーティリティを使用する場合、システムを再起動するまでパラメタは有効です。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パラメタが有効であるように設定するには、`/platform/sun4u/kernel/drv/ce.conf` ファイルを作成して、システムのデバイスに設定する必要がある特定のパラメタ値をそのファイルに追加します。

## ndd ユーティリティーを使用したパラメタの設定

ndd ユーティリティーを使用して、システムを再起動するまで有効なパラメタを設定します。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。

ここでは、ce ドライバと ndd ユーティリティーを使用して、各 ce デバイスのパラメタを変更 (-set オプションを使用) および表示 (-set オプションを使用しない) する方法について説明します。

### ▼ 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

ndd ユーティリティーで、ce デバイスのパラメタの取得または設定をする前に、ユーティリティーにデバイスインスタンスを指定してください。

1. /etc/path\_to\_inst ファイルを確認して、特定のデバイスに関連付けられたインスタンスを特定します。

```
# grep ge /etc/path_to_inst
"/pci@1f,2000/pci@1/network@0" 2 "ce"
"/pci@1f,2000/pci@2/network@0" 1 "ce"
"/pci@1f,2000/pci@4/network@0" 0 "ce"
```

この例では、3つの GigaSwift Ethernet インスタンスは、取り付けられているアダプタのものです。わかりやすくするために、インスタンスの番号は斜体 (AnswerBook 形式では太字) になっています。

2. このインスタンス番号を使用して、デバイスを選択します。

```
# ndd -set /dev/ce instance インスタンス番号
```

選択を変更するまでデバイスの選択は有効です。

### 非対話モードと対話モード

ndd ユーティリティーは、次の 2 通りのモードで使用できます。

- 非対話モード
- 対話モード

非対話モードでは、ユーティリティーを起動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。非対話モードでは、ユーティリティーを使用して複数のパラメタ値の取得と設定ができます (詳細については、ndd(1M) のマニュアルページを参照してください)。

## 非対話モードでの ndd ユーティリティーの使用

ここでは、パラメタ値の変更および表示の方法について説明します。

- 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起動すると、ユーティリティーは *value* で指定された値 (必須) をドライバに引き渡し、パラメタに割り当てます。値には `/dev/ce` 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/ce パラメタ名 値
```

adv パラメタを変更すると、次のようなメッセージが表示されます。

```
xcvr addr:0x00 - link up 1000 Mbps half duplex
```

- 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、照会操作と見なされ、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に関連づけられた値が取得されて表示されます。

```
# ndd /dev/ce パラメタ名
```

## 対話モードでの ndd ユーティリティーの使用

- 対話モードでパラメタ値を変更するには、次のように `ndd /dev/ce` と入力します。

ndd ユーティリティーから、パラメタ名の入力を求められます。

```
# ndd /dev/ce
name to get/set? (パラメタ名を入力するか、あるいはすべてのパラメタ名を表示するには ? を入力します)
```

パラメタ名を入力すると、ndd ユーティリティーから、パラメタ値の値の入力を求められます (表 1 から 表 11 までを参照してください)。

- ce ドライバでサポートされているパラメタをすべて表示するには、`ndd /dev/ce` と入力します。

```
# ndd /dev/ce
name to get/set ??
?
instance (read only)
adv_autoneg_cap (read and write)
adv_1000fdx_cap (read and write)
adv_1000hdx_cap (read and write)
adv_100T4_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)
adv_asmpause_cap (read and write)
adv_pause_cap (read and write)
link_master (read and write)
use_int_xcvr (read and write)
enable_ipg0 (read and write)
ipg0 (read and write)
ipg1 (read and write)
ipg2 (read and write)
rx_intr_pkts (read and write)
rx_intr_time (read and write)
red_dv4to6k (read and write)
red_dv6to8k (read and write)
red_dv8to10k (read and write)
red_dv10to12k (read and write)
tx_dma_weight (read and write)
rx_dma_weight (read and write)
infinite_burst (read and write)
disable_64bit (read and write)
name to get/set ?
#
```

## 自動ネゴシエーションモードの設定

デフォルトでは、自動ネゴシエーションは on に設定されています。つまり、アダプタは接続相手と通信して、互換性のあるネットワーク速度、全二重 / 半二重、およびフロー制御機能を決定します。

## ▼ 自動ネゴシエーションモードを無効にする

ネットワーク装置が自動ネゴシエーションをサポートしていない、または独自にネットワーク速度を指定する場合は、ce デバイスで自動ネゴシエーションをoff に設定します。

1. 次のドライバパラメタを、接続相手 (スイッチなど) に付属のマニュアルに記載されている値に設定します。

- adv\_1000fdx\_cap
- adv\_1000hdx\_cap
- adv\_100fdx\_cap
- adv\_100hdx\_cap
- adv\_10fdx\_cap
- adv\_10hdx\_cap
- adv\_asmpause\_cap
- adv\_pause\_cap

パラメタの値と説明の詳細は、表 2 を参照してください。

2. adv\_autoneg\_cap パラメタを0 に設定します。

```
# ndd -set /dev/ce adv_autoneg_cap 0
```

ndd リンクパラメタを変更すると、次のようなメッセージが表示されます。

```
xcvr addr:0x00 - link up 1000 Mbps half duplex
```

## ce.conf ファイルを使用したパラメタの設定

/platform/sun4u/kernel/drv ディレクトリに ce.conf ファイルを作成すると、デバイスごとにドライバパラメタのプロパティを設定できます。システムの特定のデバイスに特定のパラメタを設定する場合は、ce.conf ファイルを使用します。設定するパラメタは、2 ページの「ドライバのパラメタ値と定義」に記載されているパラメタの読み取り・書き込み可能なパラメタです。

詳細は、prtconf(1M) と driver.conf(4) のマニュアルページを参照してください。次の手順は、ce.conf ファイルでパラメタを設定する例です。

## ▼ ce.conf ファイルを使用して、ドライバパラメタを設定する

### 1. デバイスツリーで、ce デバイスのハードウェアパス名を取得します。

通常、パス名と関連するインスタンス番号は、`/etc/path_to_inst` ファイルにあります。

```
# grep ce /etc/path_to_inst
"/pci108e;abba;/pci@4,4000/network@0" 2 "ce"
"/pci108e;abba;/pci@6,2000/network@0" 1 "ce"
"/pci108e;abba;/pci@4,2000/network@0" 0 "ce"
```

- この例では次のようになります。
  - 最初の二重引用符内の部分は、デバイスツリーで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の部分はドライバ名です。
- デバイスパス名で、最後の / 文字と @ 文字に囲まれた最後の構成要素はデバイス名です。
- 最後の構成要素の前のパス名は親の名前を表します。
- @ 文字に続くカンマで区切られた数値は、デバイス番号と機能番号を表し、まとめて装置アドレスと呼ばれます。

ce.conf ファイルで明確に PCI デバイスを特定するには、デバイス名、親の名前、および装置アドレスを使用します。PCI デバイスの仕様の詳細は、pci(4) マニュアルページを参照してください。

例の 1 行目

- 名前 = pci108e,abba
- 親の名前 = /pci@4,4000
- 装置アドレス = 4,4

例の 2 行目

- 名前 = pci108e,abba
- 親の名前 = /pci@6,2000
- 装置アドレス = 6,2

例の 3 行目

- 名前 = pci108e,abba
- 親の名前 = /pci@4,2000
- 装置アドレス = 4,2

2. 上記のデバイスのパラメタを /platform/sun4u/kernel/drv/ce.conf ファイル内に設定します。

次の例では、adv\_autoneg\_cap パラメタと adv\_1000fdx\_cap パラメタは、すべての Sun GigaSwift Ethernet デバイスに設定されます (詳細は、driver.conf (4) のマニュアルページを参照してください)。

```
adv_autoneg_cap=0 adv_1000fdx_cap=0
```

次の例では、adv\_autoneg\_cap パラメタと adv\_1000fdx\_cap パラメタは、Sun GigaSwift Ethernet デバイスの単一のインスタンスに設定されます。

```
name=pci108e,abba parent=pci@4,4000 unit address+4 adv_autoneg_cap=0  
adv_1000fdx_cap=0;
```

3. ce.conf ファイルを保存します。
4. すべてのファイルを保存し、プログラムとともにすべて閉じて、ウィンドウシステムを終了します。
5. システムをシャットダウンし、再起動します。

---

## GigaSwift Ethernet ドライバの動作の統計情報

netstat -k コマンドを使用すると、統計情報を表示できます。

次の表は、MII (Media Independent Interface) 機能の読み取り専用パラメタの一覧です。このパラメタで、ハードウェアの機能を定義しています。GMII (Gigabit Media Independent Interface) は、次のすべての機能をサポートしています。

表 10 読み取り専用 ce デバイス機能

パラメタ	説明 (ローカルインタフェースの機能)
cap_autoneg	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
cap_1000fdx	ローカルインタフェースの全二重機能 0 = 1000 Mbps 全二重不可 1 = 1000 Mbps 全二重可
cap_1000hdx	ローカルインタフェースの半二重機能 0 = 1000 Mbps 半二重不可 1 = 1000 Mbps 半二重可
cap_100fdx	ローカルインタフェースの全二重機能 0 = 100 Mbps 全二重不可 1 = 100 Mbps 全二重可
cap_100hdx	ローカルインタフェースの半二重機能 0 = 100 Mbps 半二重不可 1 = 100 Mbps 半二重可
cap_10fdx	ローカルインタフェースの全二重機能 0 = 10 Mbps 全二重不可 1 = 10 Mbps 全二重可



表 10 読み取り専用 ce デバイス機能 (続き)

パラメタ	説明 (ローカルインタフェースの機能)
cap_10hdx	ローカルインタフェースの半二重機能 0 = 10 Mbps 半二重不可 1 = 10 Mbps 半二重可
cap_asm_pause	ローカルインタフェースのフロー制御機能 0 = 非同期一時停止不可 1 = ローカルデバイスからの非同期一時停止可
cap_pause	ローカルインタフェースのフロー制御機能 0 = 同期一時停止不可 1 = 同期一時停止可

## 接続相手の機能の報告

次の表は、接続相手の機能の読み取り専用パラメタの一覧です。

表 11 接続相手の機能の読み取り専用パラメタ

パラメタ	値と説明
lp_cap_autoneg	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_cap_1000fdx	0 = 1000 Mbps、全二重送信不可 1 = 1000 Mbps、全二重
lp_cap_1000hdx	0 = 1000 Mbps、半二重送信不可 1 = 1000 Mbps、半二重
lp_cap_100fdx	0 = 100 Mbps、全二重送信不可 1 = 100 Mbps、全二重
lp_cap_100hdx	0 = 100 Mbps、半二重送信不可 1 = 1000 Mbps、半二重
lp_cap_10fdx	0 = 10 Mbps、全二重送信不可 1 = 10 Mbps、全二重
lp_cap_10hdx	0 = 10 Mbps、半二重送信不可 1 = 10 Mbps、半二重
lp_cap_asm_pause	0 = 非同期一時停止不可 1 = 接続相手方向への非同期一時停止機能
lp_cap_pause	0 = 同期一時停止不可 1 = 同期一時停止可

接続相手が自動ネゴシエーションを実行できない場合 (lp\_cap\_autoneg が 0 の場合)、前の表に記載されている残りの情報は適用されず、パラメタ値 = 0 となります。

接続相手が自動ネゴシエーションを実行できる場合 (lp\_cap\_autoneg が 1 の場合)、自動ネゴシエーションと接続相手の機能を使用すると、動作速度および動作モード情報が表示されます。

次の表は、netstat -k 送信パラメタと受信パラメタの一覧です。

表 12 送信パラメタと受信パラメタ

パラメタ	説明
xcvr_inits	物理層の再初期化の回数。NDD を使用してリンクパラメタを変更するたびに 1 ずつ増えます。
rev_id	GigaSwift Ethernet デバイスのバージョン ID。現場で使用されているデバイスの識別に便利です。
xcvr_addr	インタフェース管理用の GMII/MII 物理層デバイスアドレス。
xcvr_id	GMII/MII 物理層デバイスの、MII レジスタ 2 および 3 の 10 進数 ID。
lb_mode	デバイスのループバックモードのコピー (利用できる場合)。
qos_mode	0 の場合、TX キューは、TCP/UDP 受信先ポート番号を基に単純なラウンドロビンキューイング方式で動作します。設定されている場合、TX キューは、VLAN 優先順位を指定する方式で動作します。
tx_starts	ドライバがパケットを送信しようとした回数。
tx_dma_bind_fail	ページテーブルのエントリが利用できなかったためにドライバがデバイスのアクセス可能なメモリにカーネルメモリを割り当てられなかった回数。
tx_queue0	ハードウェアの 1 番目の送信キューで送信待ちとなっているパケットの個数。
tx_queue1	ハードウェアの 2 番目の送信キューで送信待ちとなっているパケットの個数。
tx_queue2	ハードウェアの 3 番目の送信キューで送信待ちとなっているパケットの個数。
tx_queue3	ハードウェアの 4 番目の送信キューで送信待ちとなっているパケットの個数。
tx_max_pend	4 つのキューのうちで保留中の最大伝送数。
rx_hdr_pkts	256 バイトより小さい受信パケット数。
rx_mtu_pkts	256 バイトより大きく 1,514 バイトより小さい受信パケット数。
rx_split_pkts	2 ページに分割されたパケット数。

表 12 送信パラメタと受信パラメタ (続き)

パラメタ	説明
rx_no_comp_wb	ハードウェアが受信データの入力完了を送信できなかった回数。
rx_no_buf	受信バッファの領域不足のためハードウェアがデータを受信できなかった回数。
rx_new_pages	受信時に置き換えられたページ数。
rx_new_hdr_pgs	256 バイト未満の packets で埋められていて、受信時に置き換えられたページ数。
rx_new_mtu_pgs	256 バイトより大きく 1,514 バイトより小さい packets で埋められていて、受信時に置き換えられたページ数。
rx_new_nxt_pgs	複数のページに分割された packets を含んでいて、受信時に置き換えられたページ数。
rx_hdr_drops	ドライバがページを置き換えるために新しいページを割り当てられないことから、256 バイトより小さい packets のページ全体が削除された回数。
rx_mtu_drops	ドライバがページを置き換えるために新しいページを割り当てられないことから、256 バイトより大きく 1,514 バイトより小さく packets のページ全体が削除された回数。
rx_nxt_drops	ページを置き換えるためにドライバが新しいページを割り当てられなかったことから、分割 packets を持っているページが削除された回数。
rx_rel_flow	ドライバがフロー解放の指示を受けた回数。

## ▼ 接続相手の設定を確認する

- スーパーユーザーとして、`netstat -k` コマンドを入力します。

```
# netstat -k ce0
ce0:
ipackets 0 ipackets64 0 ierrors 0 opackets 0 opackets64 0
oerrors 0 collisions 0 rbytes 0 rbytes64 0 obytes 0 obytes64 0
multircv 0 multixmt 0 brdcstrcv 0 brdcstxmt 0 norcvbuf 0
noxmtbuf 0 first_collision 0 excessive_collisions 0 late_collisions 0
peak_attempts 0 length_err 0 alignment_err 0 crc_err 0 code_violations 0
ifspeed 0 rev_id 1 xcvr_inits 1 xcvr_inuse 3 xcvr_addr 0
xcvr_id 0 cap_autoneg 1 cap_1000fdx 1 cap_1000hdx 0 cap_100T4 0
cap_100fdx 0 cap_100hdx 0 cap_10fdx 0 cap_10hdx 0 cap_asmpause 0
cap_pause 1 lp_cap_autoneg 0 lp_cap_1000fdx 0 lp_cap_1000hdx 0
lp_cap_100T4 0 lp_cap_100fdx 0 lp_cap_100hdx 0 lp_cap_10fdx 0
lp_cap_10hdx 0 lp_cap_asmpause 0 lp_cap_pause 0 link_T4 0
link_speed 0 link_duplex 0 link_asmpause 0 link_pause 0
link_up 0 lb_mode 0 qos_mode 0 tx_inits 0 tx_starts 0 tx_nocanput 0
tx_msgdup_fail 0 tx_allocb_fail 0 tx_no_desc 0 tx_dma_bind_fail 0
tx_uflo 0 tx_queue0 0 tx_queue1 0 tx_queue2 0 tx_queue3 0
tx_max_pend 0 rx_inits 0 rx_hdr_pkts 0 rx_mtu_pkts 0 rx_split_pkts 0
rx_no_buf 0 rx_no_comp_wb 0 rx_ov_flow 0 rx_len_mm 0 rx_bad_descs 0
rx_nocanput 0 rx_msgdup_fail 0 rx_allocb_fail 0 rx_new_pages 0
rx_new_hdr_pgs 0 rx_new_mtu_pgs 0 rx_new_nxt_pgs 0 rx_hdr_drops 0
rx_mtu_drops 0 rx_nxt_drops 0 rx_rel_flow 0 rx_pkts_dropped 0
pci_err 0 pci_rta_err 0 pci_rma_err 0 pci_parity_err 0 pci_bad_ack_err 0
pci_drto_err 0 ipackets_cpu00 0 ipackets_cpu01 0 ipackets_cpu02 0
ipackets_cpu03 0
```

---

## ネットワークホストファイルの設定

ドライバソフトウェアをインストールした後、アダプタの Ethernet インタフェースに `hostname.cenumber` ファイルを作成してください。また、`/etc/hosts` ファイルに、IP アドレスと、Ethernet インタフェースのホスト名も作成してください。

1. コマンド行で、grep コマンドを使用して、/etc/path\_to\_inst ファイルを検索し、ce インタフェースを探します。

```
# grep ce /etc/path_to_inst
"/pci@1f,4000/pci@1/network@4" 0 "ce"
```

この例では、デバイスのインスタンスは、スロット 1 に取り付けられている Sun GigaSwift Ethernet アダプタのもので、わかりやすくするために、インスタンス番号は斜体 (AnswerBook 形式では太字) になっています。

2. ifconfig コマンドを使用して、アダプタの ce インタフェースを設定します。

ifconfig コマンドを使用して、ネットワークインタフェースに IP アドレスを割り当てます。コマンド行で、次のように入力します。IP アドレスはアダプタの IP アドレスに置き換えてください。

```
# ifconfig ce0 plumb IP アドレス up
```

詳細は、ifconfig(1M) マニュアルページおよび Solaris のマニュアルを参照してください。

- 再起動後も同じ設定を使用する場合は、/etc/hostname.cenumber ファイルを作成します。number は、使用する予定の ce インタフェースのインスタンス番号です。

手順 1 の例で取り上げたアダプタの ce インタフェースを使用して、/etc/hostname.ce0 ファイルを作成します。0 は ce インタフェースの番号です。このインスタンス番号が 1 の場合、ファイル名は /etc/hostname.ce1 となります。

- 使用する予定のない Sun GigaSwift Ethernet アダプタインタフェースの /etc/hostname.cenumber ファイルは作成しないでください。
- /etc/hostname.cenumber ファイルには、該当する ce インタフェースのホスト名を指定してください。
- 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 ファイルに記述してください。
- ホスト名は、他のどのインタフェースの他のホスト名とも異なるものにしてください。たとえば、/etc/hostname.ce0 と /etc/hostname.ce1 は同じホスト名を共有することはできません。

次の例は、Sun GigaSwift Ethernet アダプタ (zardoz-11) を持つ zardoz と呼ばれるシステムに必要な、`/etc/hostname.cenumber` ファイルを示しています。

```
# cat /etc/hostname.hme0
zardoz
# cat /etc/hostname.ce0
zardoz-11
```

3. `/etc/hosts` ファイルの中で、有効となっている ce インタフェースごとに適切なエントリを作成します。

たとえば、次のように指定します。

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