



# Sun™ フレームバッファ 使用の手引き

---

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

Part No. 806-6175-10  
Revision A, 2000年10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コ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、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：	<i>Solaris Handbook for Sun Frame Buffers</i> Part No: 806-6054-10 Revision A
-----	---

© 2000 by Sun Microsystems, Inc. 901 SAN ANTONIO ROAD, PALO ALTO CA 94303-4900. All rights reserved.



# 目次

---

はじめに	xiii
1. TurboGXplus フレームバッファ	1
TurboGXplus がサポートするモニター	1
デフォルトの画面解像度	2
画面解像度の設定	3
UNIX スクリプトによるモニターの設定	5
CPROM によるモニターの設定	6
PROM による 1 台のモニターの設定	7
UNIX スクリプトによる 1 台のモニターの設定	7
UNIX スクリプトによる複数のモニターの設定	8
2. Creator グラフィックスアクセラレータ	9
デフォルトの画面解像度	10
サポートする画面解像度	11
画面解像度を変更する (-res)	13
▼ Creator がサポートする解像度と接続されているモニターを確認する	13
▼ 画面解像度を一時的に変更する	14
▼ 画面解像度を立体表示に変更する	14

画像表示形式リストを変更する	14
画像表示形式リストの順序を変更する (-linearorder、-overlayorder)	15
デフォルトの画像表示形式を変更する (-deflinear、 -defoverlay)	16
OpenGL の画像表示形式に変更する (-expvis)	17
SERVER_OVERLAY_VISUALS 属性に変更する (-sov)	17
Creator シリーズ 3 のオプション	18
ガンマ補正を設定する (-g、-gfile)	18
ffbcconfig 画像フラグの画像表示形式リストへの影響	19
デフォルトの画像表示形式とその順序への影響	19
選択されたグループ中での画像インスタンス番号に対する影響	20
8 ビットオーバーレイグループへの SERVER_OVERLAY_VISUALS 属性と透過 SOV 画像表示形式の追加	22
ステレオコネクタ	23
Creator シリーズ 1 のステレオコネクタ	23
Creator シリーズ 2 とシリーズ 3 のステレオコネクタ	24
ステレオ信号	25
3. Creator ウィンドウシステム	27
Creator の画像表示形式	28
画像表示形式のリスト	28
オーバーレイとアンダーレイの構造	30
SX アクセラレータとの比較	31
Creator と ZX アクセラレータとの比較	33
ハードウェアのカラー LUT の使用方法	34
カラーマップのフラッシュ現象の削減	35
一般ユーザーへの注意事項	35
開発者への注意事項	36
ハードウェアのウィンドウ ID	37

カーソルの使用方法	39
ハードウェアのダブルバッファリング	39
デバイスの設定	40
製品仕様について	41
Direct Xlib	41
X11perf のベンチマークテスト	42
Creator でのピクセルコピー用ハードウェアの不在	42
背景の設定が None の場合に発生する一時的なカラー障害	42
<b>4. Creator グラフィックスアクセラレータにおける XIL の高速化</b>	<b>45</b>
XIL データの種類	45
高速化された関数	46
表 4-1 に対する注意事項	47
ダブルバッファのサポート	50
<b>5. PGX グラフィックスアクセラレータ</b>	<b>53</b>
サポートする画面解像度	54
画面解像度を一時的に変更する	55
PGX のハードウェア構成の表示	56
詳細情報	57
PEX ライブラリバグの回避策	57
<b>6. PGX32 グラフィックスアクセラレータ</b>	<b>59</b>
対話形式による設定	59
非対話形式による設定	61
使用例	62
画面解像度のその他の変更方法	62
ガイドライン	63

変更方法 63

PGX32 用の EDID 自動検出機能 63

`output-device` を使用する方法 64

ビデオモードを使用する方法 64

ビデオタイミングを使用する方法 66

PGX32 をコンソールとして設定する (オプション) 68

フレームバッファに PGX32 カードだけを使用する場合 68

PGX32 カードと二次フレームバッファを併用する場合 69

デスクトップ環境の起動 71

OpenWindows 環境 71

共通デスクトップ環境 (CDE) 72

X ディスプレイマネージャ 73

`nvedit` を使用して NVRAM を修正する 74

## 7. Elite3D グラフィックスアクセラレータ 75

デフォルトの画面解像度 76

サポートする画面解像度 77

画面解像度を変更する (`-res`) 78

▼ Elite3D がサポートする解像度と接続されているモニターを確認する 78

▼ 画面解像度を一時的に変更する 79

▼ 画面解像度を立体表示に変更する 79

画像表示形式リストを変更する 79

画像表示形式リストの順序を変更する (`-linearorder`、`-overlayorder`)  
80

デフォルトの画像表示形式を変更する (`-deflinear`、  
`-defoverlay`) 81

OpenGL の画像表示形式に変更する (`-expvis`) 82

SERVER\_OVERLAY\_VISUALS 属性に変更する (`-sov`) 83

ガンマ補正を設定する (`-g`、`-gfile`) 83

拡張オーバーレイを選択する ( <code>-extovl</code> )	84
オーバーレイ中での WID 数と色数を選択する ( <code>-maxwids</code> )	85
<code>afbconfig</code> 画像フラグの画像表示形式リストへの影響	85
デフォルトの画像表示形式とその順序への影響	85
選択されたグループ中での画像インスタンス番号に対する影響	86
8 ビットオーバーレイグループへの <code>SERVER_OVERLAY_VISUALS</code> 属性と透過 SOV 画像表示形式の追加	88
ステレオコネクタ	89
ステレオ信号	90
<b>8. Expert3D グラフィックスアクセラレータ</b>	<b>91</b>
設定ユーティリティー	91
<code>fbconfig</code> ユーティリティー	92
<code>SUNWifb_config</code> ユーティリティー	92
デフォルトの画面解像度	93
サポートする画面解像度	93
画面解像度を変更する ( <code>-res</code> )	95
▼ Expert3D がサポートする解像度と接続されているモニターを確認する	95
▼ 画面解像度を一時的に変更する	95
▼ 画面解像度を立体表示に変更する	95
画像表示形式リストを変更する	96
画像表示形式リストの順序を変更する ( <code>-linearorder</code> 、 <code>-overlayorder</code> )	96
デフォルトの画像表示形式を変更する ( <code>-deflinear</code> 、 <code>-defoverlay</code> )	97
OpenGL の画像表示形式に変更する ( <code>-expvis</code> )	98
<code>SERVER_OVERLAY_VISUALS</code> 属性に変更する ( <code>-sov</code> )	99
ガンマ補正を設定する ( <code>-g</code> 、 <code>-gfile</code> )	99
▼ 値を使用してガンマ補正を設定する	99

▼	ファイルを使用してガンマ補正を設定する	100
	拡張オーバーレイを選択する ( <code>-extovl</code> )	100
▼	拡張オーバーレイモードを選択する	100
	複数モニターの設定	100
	Expert3D カードをデフォルトのコンソールとして定義する	101
	Expert3D の外部コネクタ	103
	ステレオコネクタ	104
	ビデオ同期ポート (Expert3D のみ)	105
	マルチビューポート	105
	複数の Expert3D カードの設定	106
9.	複数のモニターの接続	109
	複数のモニターを設定する	109
	Sun Ultra システム	109
	Sun SPARCstation Sbus システム	110
	デバイスファイル名	111
	利用可能なフレームバッファを確認する	112
	コンソールから OpenWindows を起動する	113
	複数のモニターで OpenWindows を実行する	113
▼	OpenWindows バージョン 3 上で複数のモニターを使用する	113
	ポーリング順序を変更する	115
	SBus アドレス	115
	ポーリングの順序	115
	<code>sbus-probe-list</code> の変更	116
	索引	119

## 表目次

---

表 P-1	このマニュアルで使用している書体と記号	xv
表 P-2	シェルプロンプト	xvi
表 1-1	TurboGXplus がサポートするモニター	1
表 1-2	TurboGXplus のモニターセンスコード	2
表 1-3	ビデオ設定の仕様	4
表 1-4	TurboGXplus の解像度コード	6
表 2-1	Creator グラフィックアクセラレータのモニターセンスコード	10
表 2-2	Creator がサポートする画面解像度	11
表 2-3	Creator モニター同期のための Solaris のパッチ	13
表 2-4	ffbconfig の画像フラグのデフォルトの設定	14
表 2-5	Creator シリーズ 2 とシリーズ 3 のステレオコネクタの信号	25
表 4-1	高速化された XIL 関数	46
表 5-1	PGX がサポートする画面解像度	54
表 5-2	PGX の画面解像度の形式	55
表 6-1	PGX32 設定ウィンドウ	60
表 6-2	非対話形式の設定で使用するパラメタ	61
表 6-3	PGX32 の画面解像度	65
表 6-4	同期値	68
表 6-5	NVRAM エディタ	74
表 7-1	Elite3D グラフィックアクセラレータのモニターセンスコード	76
表 7-2	Elite3D がサポートする画面解像度	77

表 7-3	Elite3D モニター同期のための Solaris のパッチ	78
表 7-4	afbconfig の画像フラグのデフォルトの設定	79
表 7-5	afbconfig -extovl オプションのデフォルトの設定	84
表 7-6	afbconfig -maxwids オプションのデフォルトの設定	85
表 7-7	Elite3D のステレオコネクタの信号	90
表 8-1	Expert3D グラフィックアクセラレータのモニターセンスコード	93
表 8-2	Expert3D がサポートする画面解像度	93
表 8-3	SUNwifb_config の画像フラグのデフォルトの設定	96
表 8-4	Expert3D ステレオコネクタの信号	104

## 図目次

---

- 図 2-1 Creator シリーズ 1 のステレオコネクタ 24
- 図 2-2 Creator シリーズ 2 とシリーズ 3 のステレオコネクタ 24
- 図 2-3 Creator のステレオ信号 25
- 図 3-1 Creator アクセラレータの 11 種類の画像表示形式 30
- 図 4-1 Creator のデプスキューイング 48
- 図 6-1 PGX32 設定ウィンドウ 60
- 図 7-1 Elite3D のステレオコネクタ 89
- 図 7-2 Elite3D のステレオ信号 90
- 図 8-1 Expert3D の外部コネクタ 103
- 図 8-2 Expert3D-Lite の外部入出力コネクタ 104
- 図 8-3 Expert3D のステレオコネクタ 104
- 図 8-4 Expert3D のステレオ信号 105
- 図 8-5 Expert3D のマルチビュー入力コネクタとマルチビュー出力コネクタ 106
- 図 9-1 sbus-probe-list の書式 117



## はじめに

---

このマニュアルでは、サンのフレームバッファの設定オプションについて説明しています。このマニュアルで扱っているフレームバッファのデバイスは以下のとおりです。

- TurboGXplus™ フレームバッファ
- Creator、Creator 3D グラフィックスアクセラレータ、フレームバッファ
- PGX™ グラフィックスアクセラレータ
- PGX32™ グラフィックスアクセラレータ
- Elite3D グラフィックスアクセラレータ
- Expert3D、Expert3D-Lite グラフィックスアクセラレータ

このマニュアルは、ビデオやグラフィックスの仕様に合わせてフレームバッファやグラフィックスアクセラレータを設定する方を対象としています。

---

## マニュアルの構成

このマニュアルの構成は、以下のとおりです。

第1章では、TurboGXplus カードを備えたシステムのさまざまな画面解像度を設定する方法や、複数モニターを使用する方法について説明します。

第2章では、fbconfig ユーティリティの使用方法について説明します。このユーティリティにより、Creator 3D グラフィックスアクセラレータ、接続モニター、対応する X11 ウィンドウシステムの設定オプションを選択することができます。

第3章では、Creator および Creator 3D グラフィックスアクセラレータ上での Solaris X11 ウィンドウシステムの動作について説明します。

第4章では、Creator および Creator 3D グラフィックスアクセラレータに固有の XIL 関数について説明します。

第5章では、m64config ユーティリティを使用して、PGX グラフィックスアクセラレータの画面解像度を変更する方法について説明します。

第6章では、GFXconfig コマンドを使用して PGX32 グラフィックスアクセラレータの画面解像度を変更する方法について説明します。

第7章では、afbconfig ユーティリティの使用方法について説明しています。afbconfig ユーティリティは、Elite3D グラフィックスアクセラレータ、接続モニター、対応するX11 ウィンドウシステムの設定オプションを選択することができます。

第8章では、ifbconfig ユーティリティの使用方法について説明しています。ifbconfig ユーティリティは、Expert3D および Expert3D-Lite グラフィックスアクセラレータ、接続モニター、対応するX11 ウィンドウシステムの設定オプションを選択することができます。

第9章では、SPARCstation システムで複数のモニターを使用する方法について説明します。

---

## 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。

表 P-1 このマニュアルで使用している書体と記号

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.
<b>AaBbCc123</b>	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。	machine_name% <b>su</b> Password:
<i>AaBbCc123</i> 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 <b>ファイル名</b> 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。	% <b>grep</b> `^#define \ <b>XV_VERSION_STRING</b> '

---

## シェルプロンプトについて

シェルプロンプトの例を以下に示します。

表 P-2 シェルプロンプト

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

## 第1章

# TurboGXplus フレームバッファ

この章では、TurboGXplus™ カードを備えたシステムを設定する方法について説明します。また、画面の解像度に合わせて TurboGXplus の設定を変更する方法や、複数のモニターを使用するための設定方法についても説明します。

## TurboGXplus がサポートするモニター

TurboGXplus カードがサポートするモニターの一覧を以下に示します。

注 - 表 1-1 のモニターの機種および仕様は、予告なく変更される場合があります。詳細は、購入先にお問い合わせください。

表 1-1 TurboGXplus がサポートするモニター

モデル	パーツ番号	種類/大きさ/FCC	モニター ID センスコード	S標準の解像度と リフレッシュレート
X248A	365-1068-01	カラー 21 インチ	2	1280 x 1024、76 Hz (注 1)
GDM-20D10	365-1167-01	カラー 20 インチ	4	1152 x 900、76 Hz (注 1) 1152 x 900、66 Hz 1280 x 1024、67 Hz 1280 x 1024、76 Hz
GDM-1955A15	365-1081-01	カラー 19 インチ	3	1152 x 900、66 Hz
GDM-1962	365-1095-01	カラー 19 インチ	4	1152 x 900、76 Hz (注 1) 1152 x 900、66 Hz 1280 x 1024、67 Hz

表 1-1 TurboGXplus がサポートするモニター (続き)

モデル	パーツ番号	種類/大きさ/FCC	モニター ID センスコード	S標準の解像度と リフレッシュレート
GDM-1962B	365-1160-01	カラー 19 インチ	4	1152 x 900、76 Hz (注 1) 1152 x 900、66 Hz 1280 x 1024、67 Hz
GDM-1604A15	365-1079-01	カラー 16 インチ	3	1152 x 900、66 Hz
GDM-1662B	365-11593-01	カラー 16 インチ	6	1152 x 900、76 Hz (注 1) 1152 x 900、66 Hz
CPD-1790	365-1151-01	カラー 16 インチ	3	1152 x 900、66 Hz (注 1) 1024 x 768、77 Hz
X449	365-1286-01	カラー 15 インチ	0	1024 x 768、77 Hz
GDM-20S5	365-1168-01	グレースケール 20 インチ	2 or 4*	1280 x 1024、76 Hz or 1152 x 900、76 Hz (注 2) 1280 x 1024、67 Hz
17SMM4 A	365-1100-01	グレースケール 17 インチ	6	1152 x 900、76 Hz
M20P110	365-1099-01	グレースケール 19 インチ	4	1152 x 900、76 Hz
Non-Sun	--	Unknown	7	1152 x 900、66 Hz

注 1. 電源投入時のデフォルトの解像度です。

注 2. モニター ID センスコードは、背面にあるスイッチで選択することができます。

## デフォルトの画面解像度

モニター ID センスコードに対応するデフォルトの画面解像度を、表 1-2 に示します。

表 1-2 TurboGXplus のモニターセンスコード

コード	画面解像度
7	1152 x 900、66 Hz
6	1152 x 900、76 Hz
5	1024 x 768、60 Hz

表 1-2 TurboGXplus のモニターセンスコード (続き)

コード	画面解像度
4	1152 x 900、76 Hz
3	1152 x 900、66 Hz
2	1280 x 1024、76 Hz
1	1600 x 1280、76 Hz
0	1024 x 768、77 Hz

## 画面解像度の設定

TurboGXplus フレームバッファに画面解像度を設定する場合には、不揮発性メモリーに格納された PROM スクリプトである `nvrामrc` を実行する必要があります。PROM はデバイスの調査する際に、変数 `use-nvrामrc?` の値を調べ、真の場合は `nvrामrc` 内の FORTH コードを実行します。それ以外の場合は `probe-sbus` (Ultra 以前のシステムの場合)、`probe-all` (Ultra システムの場合)、`install-console`、`banner` を呼び出します。

`probe-sbus` または `probe-all` ステージと、`install-console` ステージの間に解像度を初期化する例を、以下に示します。

初めに `probe-sbus` または `probe-all` が呼び出されてデバイスを調べます。これによりデバイスツリーが作成され、デバイスが初期化されます。

次の行では、モニターの表示設定値を含む `vsetup` という FORTH ワードを定義しています。

コード例 1-1 `probe-sbus` ステージと `install-console` ステージ間の解像度の初期化

```
#!/bin/sh
eeprom fcode-debug\?=true
eeprom use-nvrामrc\?=true
eeprom nvrामrc='probe-sbus          (または probe-all)
: vsetup " 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET" ;
vsetup 4
```

コード例 1-1 `probe-sbus` ステージと `install-console` ステージ間の解像度の初期化 ( 続き )

```
" /sbus/cgsix@1" " override" execute-device-method drop
install-console
banner
```

以下の行は、モニタービデオの設定値を表しています。表 1-3 では、この行に含まれる設定値について説明しています。

```
" 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET"
```

表 1-3 ビデオ設定の仕様

値	説明
117000000	ピクセル周波数、ドットクロック (Hz)
71691	水平周波数 (Hz)
67	垂直周波数 (Hz)
16	水平フロントポーチ (ピクセル数)
112	水平同期幅 (ピクセル数)
224	水平バックポーチ (ピクセル数)
1280	水平表示ピクセル (ピクセル数)
2	垂直フロントポーチ (ライン数)
8	垂直同期幅 (ライン数)
33	垂直バックポーチ (ライン数)
1024	垂直表示ライン (ライン数)
COLOR	カラーモニターフラグ
0OFFSET	同期ペDESTALフラグなし

`vsetup 4` の行では、ビデオ文字列をスタック上にプッシュしています。数字の 4 は、解像度を変更するモニターのセンスコードを定義しています。サポートされているモニターコードについては、表 1-4 を参照してください。使用する数値は、TurboGXplus フレームバッファに装備されたモニターと一致させてください。

その次の行では、解像度を変更するデバイスのパスを示す文字列 `/sbus/cgsix@1` を、FORTH のスタックにプッシュしています。`cgsix@1` の `"1"` は、SBus スロット番号を表します。

SBus スロット 1 上の `cgsix` フレームバッファを変更する例を以下に示します。

コード例 1-1 SBus スロット 1 上の `cgsix` フレームバッファの変更

```
ok nvedit
0: probe-sbus                (または probe-all)
1: : vsetup " 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET" ;
2: vsetup 4
3: " /sbus/cgsix@1" " override" execute-device-method drop
4: install-console
5: banner
6: ^C
ok nvstore
ok setenv use-nvramrc? true
ok setenv fcode-debug? true
```

`"override"` 文字列は、FORTH のスタック上にあるデータを使用して解像度を再設定する `cgsix` の fcode PROM の、実際のエン트리ポイントです。`execute-device-method` は、実際に `override` を呼び出し、成功・失敗フラグを返します。このフラグは、`execute-device-method` の後に続く `drop` コマンドでは無視されます。

残りの 2 行 (`install-console` と `banner`) では、表示装置の端末ドライバをインストールして、リセット時または再起動時にバナーを表示します。

## UNIX スクリプトによるモニターの設定

UNIX スクリプトを使用して、TurboGXplus を 1280 x 1024、67 Hz の解像度に設定する例を以下に示します。

コード例 1-2 UNIX スクリプトの使用例

```
#!/bin/sh
eeprom fcode-debug?=true
eeprom use-nvramrc?=true
```

コード例 1-2 UNIX スクリプトの使用例 ( 続き )

```
eeeprom nvramrc='probe-sbus          (または probe-all)
: vsetup " 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET" ;
vsetup 4
"/sbus/cgsix@1" " override" execute-device-method drop
install-console
banner
`
```

## CPRM によるモニターの設定

PROM を使用して、TurboGXplus を 1280 x 1024、67 Hz の解像度に設定する例を以下に示します。

コード例 1-3 PROM の使用例

```
ok nvedit
0: probe-sbus          (または probe-all)
1: : vsetup " 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET" ;
2: vsetup 4
3: " /sbus/cgsix@1" " override" execute-device-method drop
4: install-console
5: banner
6: ^C
ok nvstore
ok setenv use-nvramrc? true
ok setenv fcode-debug? true
```

TurboGXplus がサポートする解像度のコードを以下に示します。

表 1-4 TurboGXplus の解像度コード

画面解像度	コード
1024 x 768、60 Hz	" 64125000,48286,60,16,128,160,1024,2,6,29,768,COLOR"
1024 x 768、70 Hz	" 74250000,56593,70,16,136,136,1024,2,6,32,768,COLOR"
1024 x 768、77 Hz	" 84375000,62040,77,32,128,176,1024,2,4,31,768,COLOR"
1152 x 900、66 Hz	" 94500000,61845,66,40,128,208,1152,2,4,31,900,COLOR"
1152 x 900、76 Hz	" 108000000,71808,76,32,128,192,1152,2,4,31,900,COLOR,0OFFSET"

表 1-4 TurboGXplus の解像度コード (続き)

画面解像度	コード
1280 x 1024、67 Hz	" 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET"
1280 x 1024、76 Hz	" 135000000,81128,76,32,64,288,1280,2,8,32,1024,COLOR,0OFFSET"
1600 x 1280、76 Hz	" 216000000,101890,76,24,216,280,1600,2,8,50,1280,COLOR,0OFFSE"

## PROM による 1 台のモニターの設定

スロット 2 の TurboGXplus カードに 16 インチのモニターを接続して、1024 x 768、60 Hz に設定する例を以下に示します。

コード例 1-1 PROM による 1 台のモニターの設定

```
ok nvedit
  0: probe-sbus                (または probe-all)
  1: : vsetup " 64125000,48286,60,16,128,160,1024,2,6,29,768,COLOR" ;
  2: vsetup 6
  3: " /sbus/cgsix@2" " override" execute-device-method drop
  4: install-console
  5: banner
  6: ^C
ok nvstore
ok setenv use-nvramrc? true
ok setenv fcode-debug? true
```

## UNIX スクリプトによる 1 台のモニターの設定

UNIX スクリプトを使用して、スロット 2 の TurboGXplus カードを 1024 x 768、60 Hz に設定する例を以下に示します。

コード例 1-2 UNIX スクリプトによる 1 台のモニターの設定

```
#!/bin/sh
eeprom fcode-debug?=true
eeprom nvramrc='probe-sbus                (または probe-all)
```

コード例 1-2 UNIX スクリプトによる 1 台のモニターの設定 ( 続き )

```
: vsetup " 64125000,48286,60,16,128,160,1024,2,6,29,768,COLOR" ;
vsetup 6
"/sbus/cgsix@2" " override" execute-device-method drop
install-console
banner
`
eeprom use-nvramrc\?=true
```

## UNIX スクリプトによる複数のモニターの設定

スロット 1 の TurboGXplus カードを 1152 x 900、76 Hz に設定し、スロット 3 の TurboGXplus カードを 1280 x 1024、67 Hz に設定して、2 台の 19 インチモニターを使用する UNIX スクリプトの例を以下に示します。

コード例 1-3 UNIX スクリプトによる複数のモニターの設定

```
#!/bin/sh
eeprom fcode-debug\?=true
eeprom nvramrc='probe-sbus (または probe-all)
: vsetup1 " 108000000,71808,76,32,128,192,1152,2,4,31,900,COLOR,0OFFSET" ;
vsetup1 4
"/sbus/cgsix@1" " override" execute-device-method drop
: vsetup2 " 117000000,71691,67,16,112,224,1280,2,8,33,1024,COLOR,0OFFSET" ;
vsetup2 4
"/sbus/cgsix@3" " override" execute-device-method drop
install-console
banner
`
eeprom use-nvramrc\?=true
```

複数のモニターの使用方法についての詳細は、第 9 章を参照してください。

## 第2章

# Creator グラフィックスアクセラレータ

---

この章では、Creator および Creator 3D グラフィックスアクセラレータの画面解像度を、モニターの特性に合わせて変更する方法について説明します。

Creator グラフィックスアクセラレータファミリーは、シリーズ 1、シリーズ 2、シリーズ 3 から成る 3 つの異なるバージョンで構成されています。各シリーズの主な違いは、画面解像度ですが、このほかにも性能の違いがあります。Creator シリーズ 1 は、ファミリー内で基礎的なグラフィックスアクセラレータとなります。Creator シリーズ 2 は、シリーズ 1 よりも高い性能と解像度を提供します。Creator シリーズ 3 は、ファミリー内で最も高い性能と画面解像度を有し、構成可能なガンマ補正と拡張オーバーレイオプションを提供します。

`ffbconfig` ユーティリティを使用して、Creator X11 の画面および対応するグラフィックスハードウェアを変更することができます。オプションはコマンド行で指定します。指定したオプションは、`OWconfig` ファイルに保存されます。これらのオプションは、次に Creator デバイスで `Xsun` を実行する際に、そのデバイスを初期化するために使用されます。`OWconfig` ファイルのオプションを変更した場合に、これらのオプションは `Xsun` セッションやシステムの再起動の後も持続します。

---

注 – `ffb` は、グラフィックスアクセラレータ Creator ファミリの UNIX デバイス名です。

---

`ffbconfig` ユーティリティによって、以下の指定が可能です。

- ビデオモード (画面解像度とリフレッシュレート)
- 画像の種類 (リニアまたは非リニア)
- 8 ビット擬似カラー画像 (オーバーレイ画像) を使用するかどうか
- 表示画像リスト中で、リニア画像を非リニア画像より前に表示させるかどうか
- デフォルトの表示画像の設定方法
- OpenGL 画像のサポート方法

- サーバーオーバーレイ画像 (SOV) が有効かどうか
- WID として使用するために予約された Creator X チャンネルピクセルの最大数
- 表示画像リストで、疑似カラーのオーバーレイ画像を疑似カラーのアンダーレイ画像より前に表示させるかどうか

Creator シリーズ 3 では、`ffbconfig` ユーティリティを使って以下の機能を追加することができます。

- ガンマの値、またはガンマ補正テーブルを含むファイルの指定による構成可能ガンマ補正
- 256 色フルカラーオーバーレイおよびハードウェアにより高速化された透過画像を実現する拡張オーバーレイオプション

## デフォルトの画面解像度

Creator システムは、デフォルトの画面解像度を調べるために VESA 標準 EDID (Extended Display Identification Data) をモニタから読み取ります。そのモニタの EDID が分からない場合は、モニタ ID センスコードを使ってデフォルトの画面解像度を調べます。

表 2-1 に、モニター ID センスコードに対応するデフォルトの画面解像度を示します。

表 2-1 Creator グラフィックアクセラレータのモニターセンスコード

コード	画面解像度
7	1152 × 900、66 Hz
6	1152 × 900、76 Hz
5	1024 × 768、60 Hz
4	1280 × 1024、67 Hz
3	1152 × 900、66 Hz
2	1280 × 1024、76 Hz
1	1152 × 900、66 Hz
0	1024 × 768、77 Hz

サン以外のモニターなど、Creator のシステムがモニターの種類を判断できない場合は、1152 × 900、66 Hz の解像度をデフォルトとします。

## サポートする画面解像度

表 2-2 に、Creator グラフィックスアクセラレータがサポートする画面解像度を示します。解像度によっては、Creator 3D シリーズ 2 以降でのみサポートされるものがあります。

表 2-2 Creator がサポートする画面解像度

画面解像度	垂直リフレッシュ レート	説明	ビデオモード形式	シンボリック名
1920 × 1200	70 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1920 x 1200 x 70	
1920 × 1080	72 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1920 x 1080 x 72	
1920 × 1200	75 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 3 でのみ対応	1920 x 1200 x 75	
1600 × 1280	76 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1600 x 1280 x 76	
1600 × 1000	76 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1600 x 1000 x 76	
1600 × 1000	66 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1600 x 1000 x 66	
1440 × 900	76 Hz	ノンインタレース、高解像度、Creator 3D および Creator シリーズ 2、3 で対応	1440 x 900 x 76	
1280 × 1024	76 Hz	ノンインタレース	1280 x 1024 x 67	1280
1280 × 1024	67 Hz	ノンインタレース	1280 x 1024 x 67	

表 2-2 Creator がサポートする画面解像度 (続き)

画面解像度	垂直リフレッシュ レート	説明	ビデオモード形式	シンボリック名
1280 × 1024	60 Hz	ノンインタレース、Creator シリーズ 2、3 で対応	1280 x 1024 x 60	
1280 × 1024	85 Hz	ノンインタレース、Creator シリーズ 3 で対応	1280 x 1024 x 85	
1280 × 800	76 Hz	ノンインタレース、Creator シリーズ 2、3 で対応	1280 x 800 x 76	
1152 × 900	76 Hz	ノンインタレース	1152 x 900 x 76	1152
1152 × 900	66 Hz	ノンインタレース	1152 x 900 x 66	
1024 × 800	84 Hz	ノンインタレース	1024 x 800 x 84	
1024 × 768	77 Hz	ノンインタレース	1024 x 768 x 77	
1024 × 768	75 Hz	ノンインタレース、高解像度、 Creator 3D および Creator シリーズ 2、3 で対応	1024 x 768 x 75	
1024 × 768	70 Hz	ノンインタレース	1024 x 768 x 70	
1024 × 768	60 Hz	SVGA	1024 x 768 x 60	svga
960 × 680	112 Hz	立体表示、ノンインタレース、56 Hz フィールドレート	960 x 680 x 112	stereo
960 × 680	108 Hz	立体表示、ノンインタレース、54 Hz フィールドレート	960 x 680 x 108	
768 × 575	50 Hz	インタレース - PAL	768 x 575 x 50	pal
640 × 480	60 Hz	インタレース - NTSC	640 x 480 x 60	ntsc
640 × 480	60 Hz	ノンインタレース、Creator 3D およ び Creator シリーズ 2、3 で対応	640 x 480 x 60	

モニターの種類によっては、Creator システムがサポートする解像度をサポートしないものがあります。Creator がサポートする解像度と接続されているモニターは、[ffbconfig](#) コマンドで確認することができます。

サポートされる解像度でモニターが確実に同期するためには、`ffbconfig` を実行する前に以下の Solaris のパッチをインストールする必要があります。表 2-3 には、各バージョンの Solaris に必要なパッチが示されています。

表 2-3 Creator モニター同期のための Solaris のパッチ

Solaris のバージョン	パッチ
Solaris 2.5.1	105790-19
Solaris 2.6	105360-33
Solaris 2.7	-
- ffb パッチ	106145-16
- xfb パッチ	106148-11
Solaris 8	N/A

注 - ffb パッチと xfb パッチが必要な場合、両方のパッチをペアでインストールする必要があります。

## 画面解像度を変更する (`-res`)

### ▼ Creator がサポートする解像度と接続されているモニターを確認する

- `ffbconfig` コマンドを以下のように入力します。

```
# ffbconfig -res \?
```

特定の解像度をモニターがサポートしているかどうかを確認するために、画面解像度を一時的に変更することができます。



注意 - ウィンドウシステムの実行中に、画面解像度を変更しないでください。画面表示に異常が発生する場合があります。

## ▼ 画面解像度を一時的に変更する

- `fbconfig` コマンドを以下のように入力します。

```
# fbconfig -res ビデオモード try
```

ビデオモードのオプションについては表 2-2 の「ビデオモード形式」と「シンボリック名」の欄を参照してください。ビデオモードを確認し、そのビデオモードを使用する場合は、5 秒以内に `y` と入力してください。

## ▼ 画面解像度を立体表示に変更する

- 以下のように入力します。

```
# fbconfig -res stereo
```

上記の例では、次回に `Xsun` を実行したときの画面解像度を、960 × 680、112 Hz の立体表示に変更します。

---

## 画像表示形式リストを変更する

`fbconfig` コマンドを使用して、X サーバーの画像表示形式リストの順序を変更することができます。表 2-4 に示す `fbconfig` オプションを使用して、特定のデバイスに合わせて転送された画像の表示形式リストを設定してください。

表 2-4 `fbconfig` の画像フラグのデフォルトの設定

名称	値	Solaris 2.5.1 の デフォルト	Solaris 2.6 の デフォルト
<code>linearorder</code>	<code>first/last</code>	<code>last</code>	<code>last</code>
<code>deflinear</code>	<code>true/false</code>	<code>false</code>	<code>false</code>
<code>overlayorder</code>	<code>first/last</code>	<code>last</code>	<code>last</code>

表 2-4 `ffbconfig` の画像フラグのデフォルトの設定

名称	値	Solaris 2.5.1 の デフォルト	Solaris 2.6 の デフォルト
<code>defoverlay</code>	<code>true/false</code>	<code>false</code>	<code>false</code>
<code>expvis</code>	<code>enable/disable</code>	<code>disable</code>	<code>enable</code>
<code>sov</code>	<code>enable/disable</code>	<code>disable</code>	<code>enable</code>

## 画像表示形式リストの順序を変更する (`-linearorder`、`-overlayorder`)

デフォルトでは、非リニア画像は、画像表示形式リスト上でリニア画像より前に表示されます。`ffbconfig` コマンドを使用して、画像表示形式リストの順序を変更することができます。

3D アプリケーションの多くはリニア画像を使用しますが、特定の 3D アプリケーションでは、`XSolarisGetVisualGamma(3)` を使用するリニア画像の検索を行わず、リストに最初に表示される 24 ビットトゥルーカラーを使用します。このようなアプリケーションを、正しい画像表示形式で動作させるには、リストの並べ換えをする必要があります。`-linearorder` オプションを使用して、リニア 24 ビットトゥルーカラーが最初に表示されるように指定してください。

画像表示形式リストの並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する場合は、`-linearorder` オプションの 1 つを指定して `ffbconfig` コマンドを入力します。

以下に例を示します。

```
# ffbconfig -linearorder first
```

デフォルトの画像表示形式リストでは、8 ビット疑似カラー画像が 8 ビット疑似カラーオーバーレイ画像よりも前になっています。この画像表示形式の順序を、`ffbconfig` コマンドを使用して変更することができます。

8ビット疑似カラーオーバーレイ画像を使用する特定のアプリケーションは、リストに最初に表示される8ビット疑似カラーオーバーレイ画像を使用します。このようなアプリケーションを正しい画像表示形式で動作させるには、8ビット疑似カラーオーバーレイ画像が、画像表示形式リストの最初に表示されるように、`-overlayorder` オプションを使用して画像表示形式リストの順序を変更します。

画像表示形式リスト内の画像表示形式の並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する場合は、`-overlayorder` オプションの1つを指定して `ffbconfig` コマンドを入力します。

以下に例を示します。

```
# ffbconfig -overlayorder first
```

## デフォルトの画像表示形式を変更する (`-deflinear`、`-defoverlay`)

画面のデフォルトの画像表示形式は、8ビット疑似カラーオーバーレイ画像です。デフォルトの画像表示形式は `ffbconfig` コマンドを使用して、リニアアンダーレイ画像またはオーバーレイ画像のいずれかに変更することができます。

- デフォルトの画像表示形式をリニア画像にする場合は、`ffbconfig` コマンドを以下のように指定します。

```
# ffbconfig -deflinear true
```

- デフォルトの画像表示形式をオーバーレイ画像にする場合は、`ffbconfig` コマンドを以下のように指定します。

```
# ffbconfig -defoverlay true
```



---

注意 – リニアオーバーレイ画像は存在しないため、`-deflinear true` と `-defoverlay true` を同時に指定した場合は、結果は不定となります。

---



---

注意 – 画像表示形式の順序オプション (`overlayorder` と `linearorder`) は、デフォルト画像表示形式オプション (`defoverlay` と `deflinear`) とは異なります。たとえば、オーバーレイ画像表示形式のグループをリストの始めに移動させても、その画像表示形式が自動的にデフォルトとなるわけではありません。特定のアプリケーションでは、変更した画像表示形式の順序がデフォルトになるものとみなし、デフォルトの画像表示形式で生成したカラーマップと、リストで最初にあった 8 ビット疑似カラー画像を一致させようとして、「BADMATCH」 X エラーを起こす場合があります。

---

## OpenGL の画像表示形式に変更する (`-expvis`)

日本語 Solaris 2.5.1 ハードウェアリリース以降では、OpenGL 拡張画像表示形式をサポートします。拡張画像表示形式では、8 ビット疑似カラー、24 ビットトゥルーカラー (リニアおよび非リニア)、24 ビットダイレクトカラー、8 ビット疑似カラーオーバーレイの 5 種類の画像グループは、同じ画像タイプの単一画像インスタンスから、複数画像インスタンスに拡張されます。同じ画像グループの異なるインスタンスは (単一バッファかダブルバッファ機能、または単一画像か立体画像機能、あるいは両方の組み合わせといった)、異なる GLX 機能を表します。画像インスタンスの数は、X サーバーが開始された画像モード (単一画像または立体画像) と、デバイス (Creator または Creator 3D) によって異なります。

### ▼ OpenGL の画像表示形式 (拡張画像表示形式) を有効にする

- 以下のように入力します。

```
# fbconfig -expvis enable
```

## SERVER\_OVERLAY\_VISUALS 属性に変更する (`-sov`)

SERVER\_OVERLAY\_VISUALS は、画像 ID、透過タイプ、透過値、画面のサーバーオーバーレイ画像 (SOV : server overlay visuals) を含むルートウィンドウの属性の 1 つです。 `fbconfig` コマンドを使用して、この属性を有効としたり、透過サーバーオーバーレイ画像表示形式の参照先の指定を切り替えることができます。

## ▼ SERVER\_OVERLAY\_PROPERTY のアドバタイズと SOV のエクスポートを設定する

- 以下のように入力します。

```
# ffbconfig -sov enable
```

---

## Creator シリーズ 3 のオプション

以下のオプションは Creator シリーズ 3 グラフィックアクセラレータでだけ使用できます。

### ガンマ補正を設定する (-g、-gfile)

ガンマ補正を設定するには、ガンマ補正值またはガンマ補正テーブルが保存されたファイルを指定します。

**-g** オプションを使用して、指定されたガンマ値に基づくガンマテーブルエントリを設定します。ガンマ値 2.22 はリニアガンマ補正を表し、Creator および Creator 3D 製品での則定値に一致します。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

## ▼ 値を使用してガンマ補正を設定する

- 以下のように入力します。

```
# ffbconfig -g 2.22
```

**-gfile** オプションは、0 から 255 までの 3 桁から成る 256 個の整数を含んだファイルから、ガンマテーブルエントリを明示的に設定します。この形式は、改行で区切られた 3 つの整数となります。各行はテーブル中のエントリとして使用される RGB 値を含んでいます。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

## ▼ ファイルを使用してガンマ補正を設定する

- 以下のように入力します。

```
# ffbconfig -gfile ファイル名
```

## 拡張オーバーレイを選択する (-extovl)

拡張オーバーレイオプションを有効にすると、シリーズ 1 または 2 でサポートされている標準オーバーレイモードを、シリーズ 3 でサポートされている新しい拡張オーバーレイ機能にシリーズ 3 Creator 3D を切り替えます。拡張オーバーレイモードは 256 色フルカラーオーバーレイとハードウェアによる高速サーバーオーバーレイ画像を実現します。アンダーレイウィンドウ ID の最大数は 32 から 64 に増え、3 種類の新しいウィンドウ ID (WID) も提供しています。拡張オーバーレイは、標準解像度で動作する DBZ ボード上でだけ使用できます。

## ▼ 拡張オーバーレイモードを設定する

- 以下のように入力します。

```
# ffbconfig -extovl enable
```

---

## ffbconfig 画像フラグの画像表示形式リストへの影響

### デフォルトの画像表示形式とその順序への影響

X サーバーの画像表示形式リストの順序は、必要に応じて `ffbconfig` 画像フラグの 1 つ (`linearorder`、`overlayorder`、`expvis`、`sov` など) を使用して変更することができます。この節では、このような画像オプションの画像表示形式リストへの影響の概要を説明します。

`ffbconfig` による変更をまったく行わない場合は、X サーバーの画像表示形式リストは、次の画像グループと順序に分類することができます。

- 8 ビット疑似カラー
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 8 ビット疑似カラーオーバーレイ
- 24 ビット非リニアトゥルーカラー
- 24 ビットダイレクトカラー
- 24 ビットリニアトゥルーカラー

デフォルトの画像表示形式は 8 ビット疑似カラー画像です。X サーバーの開始時の深さと、最初の 8 ビットオーバーレイ画像のそれぞれに合わせて、`deflinear` および `defoverlay` オプションを使用して、ポインタを最初の 8 または 24 ビットリニア画像を指すように変更することができます。`linearorder` および `overlayorder` フラグを使用して、同じ深さにある他の画像グループの前に 8 ビット疑似カラーオーバーレイのグループ、8 ビットリニアスタティックグレーのグループ、24 ビットリニアトゥルーカラーのグループ全体を置くことが可能です。たとえば、以下のように指定します。

```
# ffbconfig -overlayorder first
```

この場合、画像表示形式リストは次のように並べ換えられます。

- 8 ビット疑似カラーオーバーレイ
- 8 ビット疑似カラー
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 24 ビット非リニアトゥルーカラー
- 24 ビットダイレクトカラー
- 24 ビットリニアトゥルーカラー

## 選択されたグループ中での画像インスタンス番号に対する影響

`expvis` フラグによって、次の画像グループの画像インスタンス番号を変更することができます。

- 8 ビット疑似カラー
- 8 ビット疑似カラーオーバーレイ
- 24 ビット非リニアトゥルーカラー

- 24 ビットダイレクトカラー
- 24 ビットリニアトゥルーカラー

`expvis` によって変更されるインスタンスの数は、モニターが単一解像度になっているか立体解像度になっているかによって異なります。単一解像度になっている場合に `expvis` が有効になると、画像表示形式の順番は保存されますが、上記の各グループ内には「ダブルバッファ機能」画像インスタンスが追加されます。立体解像度になっている場合は、「ダブルバッファ・立体機能」と「単一バッファ・立体機能」の2種類の画像インスタンスが追加されます。「ダブルバッファ機能」画像インスタンスがある場合は、常に「単一バッファ機能」の前に来ます。また、単一画像インスタンスは、常に立体画像インスタンスよりも前に来ます。

たとえば、立体解像度の場合に、以下のように指定します。

```
# ffbcfg -overlayorder first -expvis enable
```

この場合、画像表示形式リストは次のようになります。

- 8 ビット疑似カラーオーバーレイ (単一、立体)
- 8 ビット疑似カラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 24 ビットリニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビット非リニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビットダイレクトカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)

---

注 - ダブルバッファ機能のオーバーレイ画像インスタンスはありません。

---

## 8 ビットオーバーレイグループへの SERVER\_OVERLAY\_VISUALS 属性と透過 SOV 画像表示 形式の追加

`sov` オプションが無効のまま使用できるオーバーレイ画像形式は、透過性のない画像形式だけです。`sov` オプションの有効化によって、透過的 SOV 画像インスタンスを画像表示形式リストに、`SERVER_OVERLAY_VISUALS` 属性をルートウィンドウ属性に追加することができます。透過 SOV 画像インスタンスは、「8 ビット疑似カラーオーバーレイ」画像グループに属しています。`SERVER_OVERLAY_VISUALS` 属性は、その画像の ID、透過タイプ、透過値、その画面で使用可能なすべてのオーバーレイ画像を含んでいます。

たとえば、立体解像度になっている場合に、以下のコマンドを入力します。

```
# ffbconfig -overlayorder first -expvis enable -sov enable
```

この場合、画像表示形式リストは次のようになります。

- 8 ビット疑似カラーオーバーレイ (単一、立体、単一 SOV、立体 SOV)
- 8 ビット疑似カラー (ダブルバッファー/単一、単一バッファー/単一、ダブルバッファー/立体、単一バッファー/立体)
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 24 ビットリニアトゥルーカラー (ダブルバッファー/単一、単一バッファー/単一、ダブルバッファー/立体、単一バッファー/立体)
- 24 ビット非リニアトゥルーカラー (ダブルバッファー/単一、単一バッファー/単一、ダブルバッファー/立体、単一バッファー/立体)
- 24 ビットダイレクトカラー (ダブルバッファー/単一、単一バッファー/単一、ダブルバッファー/立体、単一バッファー/立体)

SERVER\_OVERLAY\_VISUALS 属性には、以下の情報が含まれます。

```
=====
                          SERVER OVERLAY VISUALS (SOV) Info
=====
No. of SOV visuals = 4
SOV #0, ID 0x36, TRANSPARENT_TYPE 0, VALUE 0, LAYER 1
SOV #1, ID 0x37, TRANSPARENT_TYPE 0, VALUE 0, LAYER 1
SOV #2, ID 0x38, TRANSPARENT_TYPE 1, VALUE 255, LAYER 1
SOV #3, ID 0x39, TRANSPARENT_TYPE 1, VALUE 255, LAYER 1
=====
```

---

## ステレオコネクタ

ステレオコネクタを使用して、Creator グラフックスアクセラレータに立体視ゴーグルを接続することができます。Creator グラフックスアクセラレータ側のステレオコネクタには、次の 2 種類があります。Creator シリーズ 1 のボード上にあるメスの 3 極導体のミニフォンコネクタと、Creator シリーズ 2 のボード上にある 7 ピンの DIN コネクタです。

### Creator シリーズ 1 のステレオコネクタ

Creator シリーズ 1 のボード用のステレオコネクタは、図 2-1 に示すようにオスの 3 極導体のミニフォンコネクタです。

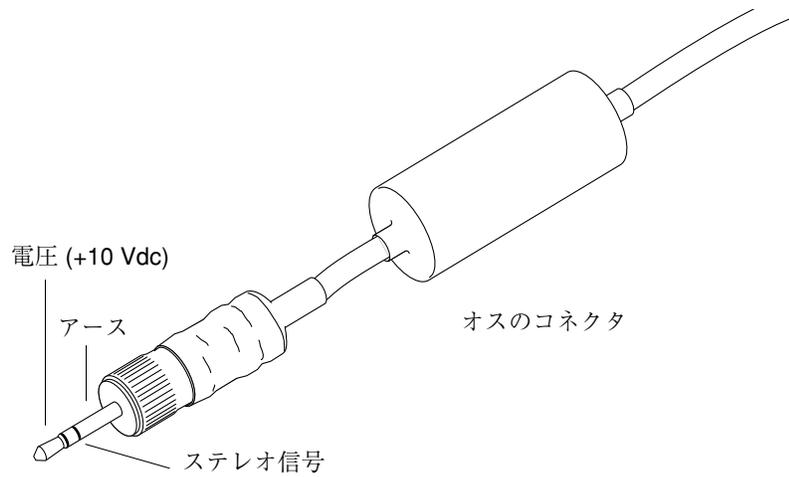


図 2-1 Creator シリーズ 1 のステレオコネクタ

## Creator シリーズ 2 とシリーズ 3 のステレオコネクタ

Creator シリーズ 2 とシリーズ 3 のボード用のステレオコネクタは、図 2-2 に示すように 7 ピンの DIN コネクタです。

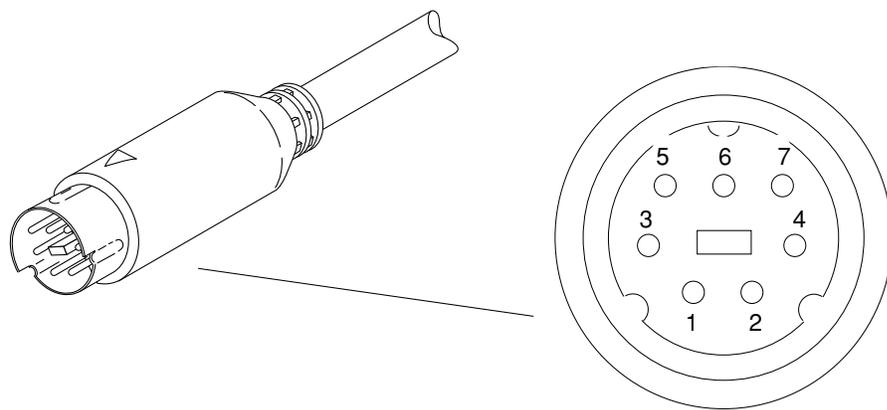


図 2-2 Creator シリーズ 2 とシリーズ 3 のステレオコネクタ

表 2-5 にステレオコネクタの信号を示します。

表 2-5 Creator シリーズ 2 とシリーズ 3 のステレオコネクタの信号

ピン	説明
1	接地
2	接続なし
3	+12V
4	ステレオ信号
5	接続なし
6	接続なし
7	接続なし

## ステレオ信号

ステレオ信号は、図 2-3 に示すように、左右の立体シャッタの切り換え周期が 50 パーセントデューティーの、TTL レベルの信号です。



図 2-3 Creator のステレオ信号

注 – Creator グラフィックスアクセラレータのステレオケーブルと立体視ゴーグルについての詳細は、購入先にお問い合わせください。



## 第3章

# Creator ウィンドウシステム

---

この章では、Creator とCreator 3D グラフィックスアクセラレータ上での、Solaris™ の X11 ウィンドウシステムの動作について説明します。

Creator アクセラレータは、GX、SX、ZX、S24 などのディスプレイアダプタの機能を集約した、高機能のアクセラレータです。Creator アクセラレータは、ZX グラフィックスアクセラレータと同様に、以下のようなフレームバッファの拡張機能を備えています。

- 複数のプレーングループ
- ハードウェアによるダブルバッファリング
- 非干渉の透過オーバーレイ
- 3次元加速処理
- 立体画像のサポート

Creator アクセラレータは、SX アクセラレータと同様に、X チャネルアーキテクチャーのディスプレイアダプタです。28 ページの「Creator の画像表示形式」を参照してください。

Creator アクセラレータは、GX アクセラレータと同様に、高速化された X11 の描画操作とハードウェアカーソルをサポートします。39 ページの「カーソルの使用方法」を参照してください。

Creator アクセラレータは、S24 アクセラレータと同様に、ガンマ補正された画像と、非補正の画像の両方を備えています。28 ページの「Creator の画像表示形式」を参照してください。

Creator アクセラレータは、GX、SX および ZX グラフィックスアクセラレータと同様に、複数のモニターの表示モードをサポートします。40 ページの「デバイスの設定」を参照してください。

Creator アクセラレータは、SB (Creator) と DBZ (Creator 3D) のどちらにも設定することができます。SB は "Single Buffer"、DBZ は "Double Buffer plus Z" を意味します。Z値は、ピクセルごとの深さを表します。それぞれの設定では、ボード上のビデオメモリーの大きさが異なります。Creator 3D アクセラレータでは、ハードウェアのダブルバッファリングと 3 次元描画を実行するためのメモリーが付加されています。

---

## Creator の画像表示形式

Creator アクセラレータの出荷時のデフォルト表示形式は、8 ビット疑似カラーです。他のデフォルトを設定するには、[Xsun\(1\)](#) の `defdepth` および `defclass` オプション、または [ffbconfig\(1m\)](#) の `-defoverlay` および `-deflinear` オプションを使用してください。

---

注 – Creator アクセラレータの別名は FFB (Fast Frame Buffer) です。このコード名は、Creator ソフトウェアのパッケージ名、ロード可能なデバイスのパイプラインモジュール名、設定プログラム名、デバイスのマニュアルページ名などに使用されています。

---

OpenWindows を起動する際に、通常の OpenWindows のコマンド行オプションを使用して、デフォルトの表示形式を変更することができます。使用可能な表示形式は、すべてデフォルトとして選択することができます。`openwin defdepth 24` オプション ([Xsun\(1\)](#) を参照) を使用して 24 ビットトゥルーカラーをデフォルトに設定した場合には、カラーマップのフラッシュ現象が減少します。

## 画像表示形式のリスト

Creator アクセラレータによって、X11 の画面の画像表示形式リスト上の 11 種類の表示形式が使用可能となります。画像表示形式は、[XGetVisualInfo\(3\)](#) または [XMatchVisualInfo\(3\)](#) を使用して照会することができます。また、表示形式の線形性 (線形性) については、[XSolarisGetVisualGamma\(3\)](#) によって照会することができます。

以下に使用可能な 11 種類の表示形式を示します。

- 8 ビット疑似カラー
- 8 ビットスタティックカラー

- 8ビットグレースケール
- 8ビットスタティックグレー
- 8ビットトゥルーカラー
- 8ビットダイレクトカラー
- 8ビットスタティックグレーリニア
- 24ビットトゥルーカラー
- 24ビットダイレクトカラー
- 24ビットトゥルーカラーリニア
- 8ビット疑似カラーオーバーレイ

---

注 – 上記の画像表示形式において、特にリニア画像に設定されていない場合は、ガンマ補正されていない非リニア画像となります。また8ビットの画像は、特にオーバーレイに指定されていない場合には、Creator アクセラレータのアンダーレイのプレーングループとなります。24ビットの画像は、常にアンダーレイとなります。

---

図 3-1 に、Creator アクセラレータの画像の、フレームバッファ内のピクセル記憶装置（プレーングループ）に対する関連を示します。X、B、G、R は、ピクセルのデータが格納される4種類の8ビットチャンネルを示します。図では、拡張オーバーレイが無効になっている Creator シリーズ 1、2、3 用のピクセル領域を示しています。

B、G、R チャンネルは、8ビットのピクセルデータ（赤のチャンネルだけ）と、24ビットのピクセルデータ（3種類のチャンネルを使用）のどちらも格納することができます。R チャンネルは、7種類の異なる画像（8R 画像）をウィンドウに格納します。BGR チャンネルは、ウィンドウに3種類の異なる画像（24ビットの画像）を格納します。X チャンネルは、8X の疑似カラーの1種類だけを格納します。

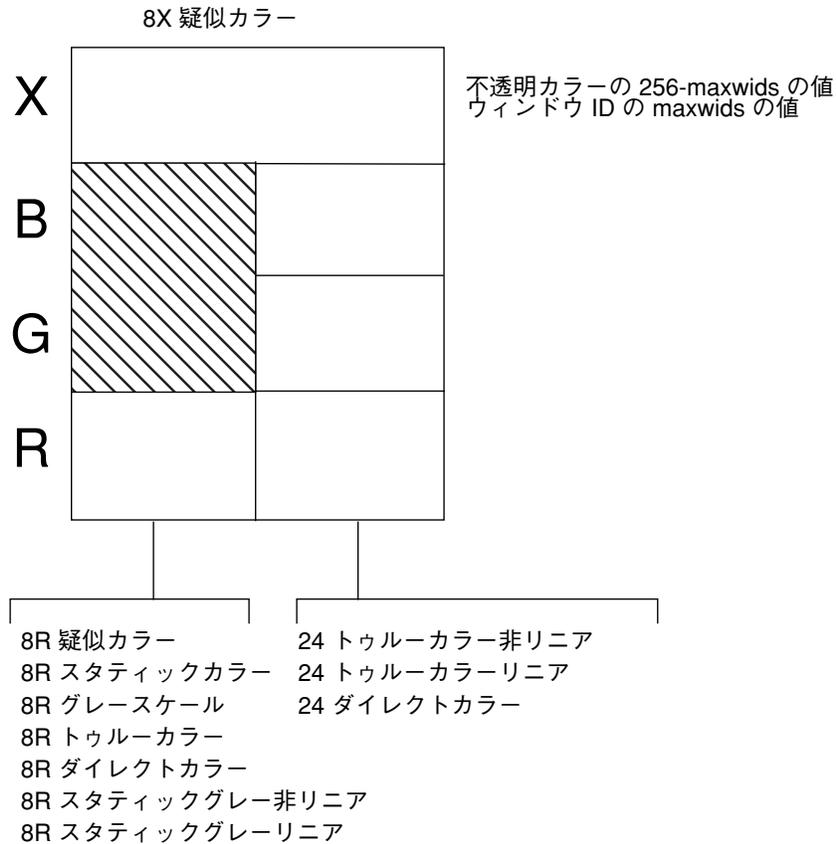


図 3-1 Creator アクセラレータの 11 種類の画像表示形式

アンダーレイ画像のカラーマップのサイズは 256 です。Creator シリーズ 1 および 2 ではオーバーレイ画像のカラーマップのサイズは 256 - maxwids です。Creator シリーズ 3 は、256 色フルカラーオーバーレイを使用するか、256 - maxwids 色のシリーズ 1 または 2 との互換モードで動作するように設定できます。

## オーバーレイとアンダーレイの構造

アンダーレイの 8 ビット疑似カラー画像は、ピクセルがフレームバッファの赤のチャンネルに格納されるため、8R 画像と呼ばれます。また、オーバーレイの 8 ビットの疑似カラーは X チャンネルに格納されるため、8X 画像と呼ばれます。

オーバーレイ画像のウィンドウ上のピクセルは、アンダーレイ画像のウィンドウ上のピクセルに影響しません。ただし、アンダーレイ画像のウィンドウ上のピクセルは、オーバーレイ画像のウィンドウ上のピクセルに影響します。これは、アンダーレイウィンドウが BGR チャンネル (あるいは R チャンネルだけ) に依存するカラーデータであるのに、X チャンネル上に WID 情報も持っているためです。x11 expose event (障害) の発生の原因となります。

オーバーレイウィンドウがアンダーレイウィンドウに遮断されている場合、アンダーレイデータの WID 部分によってオーバーレイウィンドウのカラーデータが破壊されてしまいます。アンダーレイウィンドウが再び取り除かれると、x11 expose event がオーバーレイウィンドウの破壊された部分に送り出されます。これは、互いに干渉しないアンダーレイとオーバーレイを備えた ZX アクセラレータとは異なります。Creator アクセラレータの 8 ビットのアンダーレイのウィンドウ上のピクセルは、24 ビットのアンダーレイのウィンドウ上のピクセルに影響します。

Creator アクセラレータは、X チャンネルアーキテクチャーに従います。このアーキテクチャーでは、8X プレーングループ表示に使用されていない色のピクセル値とコードを、アンダーレイ画像のピクセル表示を操作するウィンドウ ID として使用します。

Creator 3D シリーズ 3 には、ZX アクセラレータと同様の非干渉型のオーバーレイとアンダーレイを持つ拡張オーバーレイモードが備えられています。このモードが有効になると、WID プレーングループは X またはオーバーレイチャンネルを共有しなくなるため、x11 expose event (障害) は発生しなくなります。

`maxwids` とは、オーバーレイのピクセル値をハードウェアウィンドウ ID として、どの程度使用するかを特定する `fbconfig(1M)` 設定オプションです。詳細は、37 ページの「ハードウェアのウィンドウ ID」を参照してください。`maxwids` の最小値は 1、デフォルト値は 32 です。オーバーレイ画像は、通常のカラーマップのエントリ数が 256 より少ないため、部分画像となります。この画像のカラーマップに、クライアントが指定されたカラーマップエントリの数以上のピクセル値で描画すると、エラーは発生しませんが、表示された色の定義はされません。

## SX アクセラレータとの比較

Creator アクセラレータの画像表示アーキテクチャーは、SX アクセラレータのディスプレイアダプタである CG14 と類似しています。CG14 もまた、8 ビットまたは 24 ビットのアンダーレイ画像と単一 8 ビット疑似カラーのオーバーレイ画像を備えた、X チャンネルアーキテクチャーのディスプレイアダプタです。Creator アクセラレータの画像表示アーキテクチャーと CG14 には、以下の 2 つの大きな違いがあります。

## ガンマ補正

Creator アクセラレータには、ガンマ補正ありの画像表示形式と、ガンマ補正なしの画像表示形式があります。ガンマ補正された画像は、リニア画像と呼ばれます。以下にリニア画像の種類を示します。

- 24 ビットトゥルーカラーリニア画像
- 8 ビットスタティックグレーリニア画像

リニア画像と非リニア画像は、どちらも X11 画面の画像リスト上に存し、[XGetVisualInfo](#) によって照会することができます。リニア性は、X11 コアプロトコルで認識される画像上の特性ではないため、相対する非リニア画像からリニア画像を認識するために、拡張ルーチン [XSolarisGetVisualGamma\(1\)](#) が呼び出されます。詳細は、[XSolarisGetVisualGamma\(1\)](#) のマニュアルページを参照してください。

CG14 は、Creator アクセラレータのようなりニア画像を備えていません。CG14 は、特別なガンマ LUT を使用して、画面全体に影響するガンマ補正を行います。したがって、CG14 では、ガンマ補正された 24 ビットのウィンドウと補正されていないウィンドウを、画面上に同時に開くことはできません。Creator の画面上では、両方のウィンドウを同時に開くことが可能です。

## 単一のカラー LUT

CG14 では、ハードウェアのカラー LUT を 2 つ備えています。1 つは 8 ビットのアンダーレイ画像で使用され、もう 1 つは 8 ビットのオーバーレイ画像で使用されます。一方、Creator アクセラレータシリーズ 1 と 2 では、ハードウェアのカラー LUT を 1 つだけ備えています。これは、2 つのウィンドウのカラーマップが、同じピクセルの位置で同じ色を持っていない場合には、Creator アクセラレータ上のオーバーレイウィンドウが、8 ビットのアンダーレイウィンドウに対し、カラーマップのフラッシュ現象を起こすことを意味します。Creator シリーズ 3 には、Xserver が割り当てや共有を管理する 4 種類のカラー LUT があります。

Creator アクセラレータ上にあるカラー LUT が 1 つだけの場合には、透過オーバーレイのアプリケーション上でオーバーレイとアンダーレイが色を共有するようにプログラミングしてください。オーバーレイ画像は、アンダーレイ画像とは常に異なるため、透過オーバーレイのアプリケーションは最低 2 種類のカラーマップを必要とし、その 1 つはオーバーレイに、もう 1 つはアンダーレイに割り当てられます。オーバーレイウィンドウは、通常アンダーレイウィンドウの子ウィンドウであり、各ピクセルのレイヤー間の関係は、アプリケーションによって管理されます。このような状態

で、マウスポインタがアンダーレイとオーバーレイの境界の内側にある場合には、オーバーレイのカラーマップはハードウェアのカラー LUT にインストールされますが、アンダーレイのカラーマップはインストールされません。オーバーレイのカラーマップを通して表示された際に、アンダーレイのピクセルが正しい色を表示しているかどうかを確認してください。アンダーレイとオーバーレイ両方のカラーマップの同じ位置にそれぞれ色を割り当てることによって、確認することができます。

## Creator と ZX アクセラレータとの比較

ZX アクセラレータと異なり、Creator アクセラレータの出荷時のデフォルト画像はオーバーレイではありません。ZX アクセラレータでは、デフォルト画像は 8 ビットのオーバーレイの疑似カラーとなりますが、Creator アクセラレータでは、SX アクセラレータと同様に、8 ビットのアンダーレイの疑似カラーとなります。アンダーレイのウィンドウで影響を受けないポップアップウィンドウを持つアプリケーションを作成する場合には、単純にデフォルトの画像を使用することはできません。そのためアプリケーションは、影響を受けないオーバーレイ画像を検索するために、[XSolarisOvlSelectBestOverlay](#) を呼び出す必要があります。詳細は、X ウィンドウシステムのオーバーレイについて記述している、Solaris のマニュアルを参照してください。

---

注 - [XSolarisOvlSelectBestOverlay](#) は、Solaris 2.4 リリースから提供されています。Solaris 2.4 と Solaris 2.4 以前のオペレーティングシステム上でアプリケーションを実行する場合には、この関数の外部参照は、`#pragma weak` を定義します。これにより、プログラムはシンボルの値を確認することができます。プログラムが Solaris 2.3 またはそれ以前のバージョンで実行されている場合は、シンボルの値が 0 になります。この場合には、オーバーレイ画像を検索するために [XSolarisOvlSelectBestOverlay](#) を呼び出すことはできません。そのかわりに、アプリケーションは、[XGetVisualInfo](#) を使用して、256 以下のカラーマップのエントリを持つ最初の 8 ビット画像を検索することができます。ただし、この方法は Creator アクセラレータ特有のものであり、Creator アクセラレータを使用するためのコードは、他のデバイスへの移植性があるとは限りません。

---

## ウィンドウマネージャーについての注意事項

Creator アクセラレータのデフォルトの画像がオーバーレイでないため、オーバーレイのウィンドウがオーバーライドリダイレクトでない場合には、問題が発生することがあります (たとえば、ウィンドウマネージャの装飾ウィンドウで囲まれる場合など)。Solaris でサポートしている `olwm`、`mwm`、`dtwm` などのウィンドウマネージャーでは、ツールキットのサブウィンドウが、デフォルト画像の装飾ウィンドウで囲まれています。これは、アプリケーション内で、ポップアップウィンドウにデフォルト以外の画像が指定されている場合も同様です。

たとえば、デフォルトの画像が 8 ビットのアンダーレイの疑似カラーの場合に、アプリケーションがツールキットに対し、ポップアップウィンドウを 8 ビットのオーバーレイの疑似カラー画像に置くように指定しても、ウィンドウマネージャはそのポップアップウィンドウを、8 ビットのアンダーレイの装飾ウィンドウで囲みます。このように、ポップアップウィンドウは、他のアンダーレイのウィンドウに悪影響を与えません。

この問題を回避する方法を以下に示します。

- ウィンドウシステムを起動する前に以下のように入力して、デフォルトの画像をオーバーレイ画像に設定します。

```
# /usr/sbin/ffbconfig -defoverlay true
```

オーバーレイ画像では、アンダーレイの 8 ビット画像と比較してカラーマップのエントリ数が少なく、デフォルトのカラーマップをすぐに使い切ってしまうため、カラーマップのフラッシュ現象が増加します。

- アプリケーションを書き換えて、上書き・リダイレクトを確認するポップアップウィンドウを作成します。これらのウィンドウはツールキットを介さないで直接 Xlib 関数で管理します。

---

## ハードウェアのカラー LUT の使用方法

以下に示す画像表示形式では、カラー LUT を使用します。

- 8 ビット疑似カラー
- 8 ビットスタティックカラー

- 8 ビットグレースケール
- 8 ビットトゥルーカラー
- 8 ビットダイレクトカラー
- 24 ビットダイレクトカラー
- 8 ビット疑似カラーオーバーレイ

カラー LUT を 1 つだけ持つ Creator アクセラレータでは、画像表示形式のカラーマップは、相互にカラーマップのフラッシュ現象を発生させます。カラーマップのフラッシュ現象を回避する方法については、35 ページの「カラーマップのフラッシュ現象の削減」を参照してください。

他の Creator アクセラレータの画像表示形式では、カラー LUT のリソースを使用しないため、カラーマップのフラッシュ現象は発生しません。

---

## カラーマップのフラッシュ現象の削減

Creator アクセラレータの 24 ビットトゥルーカラー画像は、カラーマップのフラッシュ現象を発生させずに、同時に 1600 万色以上の表示をすることができます。Creator の描画エンジンは、24 ビットの描画用に最適化されています。

### 一般ユーザーへの注意事項

24 ビットトゥルーカラーの画像表示形式では、カラーマップがフラッシュすることなく、高速描画をすることができます。ただし、Creator アクセラレータの出荷時のデフォルトは、8 ビット疑似カラー画像表示になっています。この出荷時のデフォルトは、24 ビットの画像表示形式では正常に機能しない X ウィンドウシステムのアプリケーションを実行するための設定です。多くのデスクトップアプリケーションは、この画像表示形式で正常に動作します。

デスクトップ上でカラーマップのフラッシュ現象を減らすには、デフォルトの画像表示形式を 24 ビットトゥルーカラーに設定してウィンドウシステムを実行する方法があります。この方法は、Creator アクセラレータ上でウィンドウシステムを実行するのに適しています。

- 24 ビットトゥルーカラーで OpenWindows を実行するには、以下のコマンドを入力します。

```
% openwin -dev /dev/fbs/ffb0 defdepth 24
```

- 上記のモードで CDE を実行する場合は、`/usr/dt/config/Xservers` ファイルを編集し、サーバーの X 起動コマンドに `"defdepth 24"` を追加してください。

この画像表示形式を使用する際の注意点を以下に示します。

- X ウィンドウシステムのアプリケーションには、24 ビットの画像表示形式では正常に機能しないものがあります。このようなアプリケーションでは、プログラムが実行に失敗し、BadMatch エラーメッセージを表示します。プログラムによっては、コアダンプが発生したり、正しくない色を表示したりする場合があります。このようなアプリケーションを使用する場合は、アプリケーションを 8 ビット疑似カラーをデフォルトの画像表示形式として実行しなおすことによって、問題の原因を特定することができます。この状態で正しく動作した場合には、そのプログラムが 24 ビットトゥルーカラーのデフォルト画像表示形式では機能しないことがわかります。この場合、24 ビットトゥルーカラーのデフォルト画像表示形式で動作するように、プログラムをアップグレードする必要があります。プログラムがアップグレードされるまでは、出荷時のデフォルトの 8 ビット疑似カラーの画像表示形式を使用してください。
- デフォルトの画像表示の深さが 24 ビットの場合には、ピクセルマップおよびウィンドウのバッキングストアは、深さが 8 ビットの場合の 4 倍の領域を必要とします。これによって、サーバーの実行イメージが増大することはありませんが、スワップ領域の使用率が増大します。多くのピクセルマップやバッキングストアウィンドウを使用するプログラムでは、設定を 8 ビットモードにしてください。

## 開発者への注意事項

24 ビット透過モードでプログラミングを行ってください。デフォルトの画像表示形式が 24 ビットトゥルーカラーの場合は、プログラムは正常に実行されます。

24 ビット透過モードのプログラムが正常に動作しない場合には、いくつかの理由が考えられます。以下に、プログラムを正常に実行するための注意点を示します。

- デフォルトの画像表示形式が常に 8 ビット疑似カラーであることを前提にしない。

プログラムによっては、プログラムが 8 ビットだけに対応したシステムから移植され、アップグレードされないために、8 ビットの深さでのみ実行可能なものがあります。また、256 要素の配列でピクセルを格納するプログラムや、カラーマップのダブルバッファリングのために、カラーマップの変更を要求するプログラムもあります。

プログラムが 8 ビット疑似カラーを条件とする場合は、デフォルトの画像表示形式の深さとクラスを確認してください。デフォルトの画像表示形式が 8 ビット疑似カラーでない場合には、画像表示形式のリストを検索するようにしてください。

- 親ウィンドウのボーダーのピクセルを継承しない。

Creator アクセラレータのような複数のプレーングループを持つ装置では、プログラム中で作成するウィンドウの深さが、親ウィンドウ (通常はルートウィンドウ) の深さに合致しない場合があります。ボーダーのピクセル値を明示的に指定しない場合は、ウィンドウのボーダーのピクセル値には、親ウィンドウの値が継承されます。深さが異なる場合には、`XCreateWindow` は実行に失敗して `BadMatch` エラーを表示します。プログラムを実行する場合は、`XCreateSimpleWindow` ではなく、常に `XCreateWindow` を使用して、ボーダーのピクセル値を明示的に指定してください。

ウィンドウシステムの `"defdepth 8"` および `"defdepth 24"` の両方のモードで、アプリケーションプログラムをテストしてください。

---

## ハードウェアのウィンドウ ID

ピクセルの内容を表示するために、ハードウェアのウィンドウ ID (WID) が必要となります。

---

注 - ここで説明するウィンドウ ID と X プロトコルのウィンドウ ID とは異なります。X プロトコルのウィンドウ ID は、ウィンドウを識別する `XID` です。ハードウェアのウィンドウ ID とは、ウィンドウの外観を操作するフレームバッファに取り込まれる値です。

---

オーバーレイのピクセルコードには、可視的な色を表示する不透明なピクセルとして使用されるものがあります。それ以外のオーバーレイのピクセルコードは、アンダーレイのウィンドウの表示属性を制御するために使用されます。このコードは、ハード

ウェアのウィンドウ ID (WID) と見なされます。カラーマップの上位のコード (255 に近いコード) のいくつかは、WID として使用されます。WID の実際の数は、`ffbconfig -maxwids` オプションによって設定することができます。WID として使用されるオーバーレイコード 1 つにつき、オーバーレイのカラーマップのエントリ数が 1 つ減少します。`maxwids` のデフォルト値は 32 のため、オーバーレイには 224 の不透明なピクセルの値が存在します。

ハードウェア WID のうちの 1 つは、常にデフォルトの画像表示形式のウィンドウ用に確保されます。それ以外の WID は、以下のウィンドウに優先度に基づいて割り当てられます。

- デフォルト以外の画像表示形式を持つウィンドウ
- ダブルバッファのウィンドウ
- ハードウェア WID のクリッピングのために割り当てられた一意の WID をもつウィンドウ (このクリッピング手法は、サン の 3 次元レンダリングライブラリで使用されます)。

デフォルト以外の画像表示形式のウィンドウは、同一の画像表示形式の他のウィンドウと WID を共有することができます。ただし、ダブルバッファのウィンドウは、固有の WID を持つウィンドウのように、常に固有の WID を要求します。

上記のようなウィンドウを作成した場合には、作成可能な別の種類のウィンドウの数が減少してしまいます。使用可能なすべての WID が割り当てられてしまうと、`XCreateWindow` の呼び出しは失敗し、`BadAlloc` のエラーメッセージが表示されません。

`maxwids` には、 $2n$  (1、2、4、8、16、32) を指定してください。デフォルトの値は 32 です。

`maxwids` の値を小さくすることにより、オーバーレイの不透明なカラーピクセルの数が増加しますが、ユーザーが作成可能な XGL、ダブルバッファ、デフォルト以外の画像表示形式のウィンドウの数は減少することになります。

---

## カーソルの使用方法

Creator アクセラレータは、ハードウェアカーソルを備えています。このカーソルの画像は、ビデオ信号の出力に直接合成されます。一方、ソフトウェアカーソルは、フレームバッファに描画され、以前のフレームバッファの内容は一時的に保存されます。ソフトウェアカーソルは、ハードウェアカーソルと比較して、オーバーヘッドが発生しやすくなっています。ハードウェアカーソルでは、カーソルの動きに対する応答が最適化されています。

Creator のハードウェアカーソルの最大の大きさは、64 × 64 です。幅と高さが 64 以下のカーソルには、ハードウェアカーソルを使用してください。ハードウェアカーソルを使用するよう定義されていない場合は、カーソルはソフトウェアカーソルとして描画されます。

Creator アクセラレータのソフトウェアカーソルは、オーバーレイのプレーングループに描画されます。ソフトウェアカーソルは、オーバーレイウィンドウのピクセルに影響しますが、アンダーレイウィンドウのピクセルには影響しません。

X11 のカーソルは、クライアントのアプリケーションが要求する前景と背景の色を持っています。ハードウェアカーソルの色は要求した色と一致しますが、ソフトウェアカーソルの色は要求した色の近似色となる場合があります。

---

注 – Solaris の Visual グラフィックスライブラリの制限事項により、ソフトウェアカーソルが適切に削除されない場合があります。この制限事項を回避するには、DGA グラブウィンドウ上のカーソルを強制的にハードウェアカーソルにしてください (ただし、カーソルの一部が表示されなくなる場合があります)。

---

---

## ハードウェアのダブルバッファリング

ハードウェアのダブルバッファリングは、MBX と XGL でサポートされます。MBX と XGL のダブルバッファリングを同時に同じウィンドウで使用することはできません。

ハードウェアのダブルバッファリングは、Creator アクセラレータすべての構成の 8R ウィンドウで使用することができます。24 ビットのウィンドウにおけるハードウェアのダブルバッファリングは、Creator/DBZ (Creator3D) の構成でのみ有効です。8X のオーバーレイウィンドウは、Creator アクセラレータの構成では、ハードウェアのダブルバッファリングが行われることはなく、常にソフトウェアのダブルバッファリングが行われます。

MBX を介してダブルバッファリングを行うには、WID が 1 つ必要です。これにより、作成できる WID を必要とするウィンドウの数が減少します。詳細は、37 ページの「ハードウェアのウィンドウ ID」を参照してください。使用可能な WID がない場合には、MBX はソフトウェアのバッファリングに戻ります。ソフトウェアのバッファリングでは、バッファの交換 (フリップ) のために、バックバッファピクセルマップからウィンドウにコピーされたデータが使用されます。

MBX では、バッファの交換 (フリップ) は同時に発生します。サーバーへの要求は、モニターの垂直帰線期間が発生して、バッファの交換が発生するまで応答されません。X11 のクライアントは、処理を続行してサーバーに要求を送ることが可能ですが、バッファの交換が終了するまではそれらの要求は処理されません。

---

## デバイスの設定

`/usr/sbin/ffbconfig` を実行することにより、Creator アクセラレータのモニター表示モード、デフォルト画像表示形式、デフォルトのリニア順、WID の数を変更することができます。詳細については、`ffbconfig(1M)` のマニュアルページを参照してください。

モニターが立体表示モードに設定されていない場合には、DGA を介して立体表示を行うことはできません。

`ffbconfig` によって、共有の `OWconfig` ファイルを更新する際に、`/usr/openwin` ディレクトリが遠隔マウントされている場合は、エラーが発生する可能性があります。`ffbconfig` が、`setuid` により `root` の権限で実行可能となっている場合でも、同様にエラーが発生します。このエラーは、UNIX 上ではローカルマシンのルートユーザーが、遠隔マシンのルートユーザーとは異なるために発生するものです。これを回避するには、遠隔マシンにログインして `ffbconfig` を実行してください。

---

## 製品仕様について

この節では、様々なアプリケーション上で動作する Creator グラフィックスアクセラレータの製品仕様について説明します。

### Direct Xlib

Direct Xlib は、Creator アクセラレータではサポートされていません。Direct Xlib の代わりに、X 共有メモリー転送機能 (Solaris 2.5 の新機能) を使用してください。共有メモリーを転送するには、クライアント環境に以下の環境変数を設定してください。

```
setenv DISPLAY :0
setenv XSUNTRANSPORT shmem
setenv XSUNSMESIZE 512
```

---

注 - `setenv XSUNSMESIZE 512` では、クライアントが要求するバッファーの大きさを 512KB に指定しています。これ以外の大きさを指定することも可能ですが、転送速度とシステムが消費するメモリー量を考慮すると 512KB が適切な値となります。

---

フレームバッファーに高速ピクセル転送をする際に、`XPutImage` と Direct Xlib を使用するアプリケーションでは、代わりに、MITSHM 拡張機能の `XShmPutImage` を Creator アクセラレータ上で使用するようにしてください。この機能を使用することにより、X サーバーと同一のマシン上で作業を行う場合に、ピクセルの高速転送が可能になります。

24 ビットの画像に `XShmPutImage` を使用している場合には、共有メモリーの割り当て量を、デフォルトで設定されている量よりも大きくする必要があります。詳細は42ページの「X11perf のベンチマークテスト」で説明します。

## X11perf のベンチマークテスト

デフォルトのサーバーの画像が 24 ビットトゥルーカラーの場合に、X11perf のベンチマークテストを行うと、`-shmput<nn>` テストが実行に失敗します。これは、`-shmput<nn>` テストでは、デフォルトの `shmsys` 設定で割り当てられた共有メモリよりも、多くのメモリを必要とするためです。X11perf の X11R5 バージョンの場合はコアダンプが発生しますが、X11R6 バージョンの場合は、エラーが表示されて測定は実行されません。

`-shmput<nn>` テストを実行する場合は、共有メモリの量を増やす必要があります。

- メモリーの量を変更するには、`/etc/system` に以下の行を追加します。

```
set shmsys:shminfo-shmmax=8192000
```

## Creator でのピクセルコピー用ハードウェアの不在

Creator アクセラレータには、フレームバッファ内にピクセルをコピーするハードウェアが存在しません。ピクセルのコピーは、CPU レジスタにピクセルを読み込み、その後フレームバッファに再び書き込むという方法で行われます。このように、フレームバッファ内でのピクセルのコピーは、主に CPU で行われます。システムの負荷が大きいと、ウィンドウでのドラッグ操作の性能が低下する場合があります。たとえば、SunVideo ストリームを実行中に、ShowMe Whiteboard の “Snap Region” での画像ドラッグ操作の速度が低下する場合があります。一方、TurboGX アクセラレータにはピクセルをコピーするハードウェアが存在するため、このように処理速度が低下することはありません。

## 背景の設定が None の場合に発生する一時的なカラー障害

イメージツールのイメージパレットウィンドウを、マウスのセレクトボタンを使用してイメージツールのメインウィンドウへ移動させると、メインウィンドウが一時的にシアン色になる場合があります。この状態はすぐに修復されますが、ユーザーが見て気づく程度に持続します。この現象は、ドラッグするウィンドウが部分的にイメージツールの内側に入っている場合に発生するものです。これは、イメージツールのメインウィンドウが XView の `WIN_TRANSPARENT` ウィンドウであり、対応する X ウィンドウの背景が `None` に設定されているため、背景が `None` に設定されている場合に

は、アプリケーションは重なっているウィンドウが移動した後で、障害のある部分を修復します。クライアントが障害のあるイベントを処理し、描画要求を送信する必要があるために、この障害修復作業には多少時間がかかります。

イメージツールのイメージパレットウィンドウは、8 ビットのウィンドウです。ピクセルデータは、Creator アクセラレータの赤のチャンネルにあります。8 ビットウィンドウの背景のピクセル値は、通常、0x02 という小さな値です。赤のチャンネルにあるピクセルデータは、緑のチャンネルや青のチャンネルにあるデータに悪影響を与えることはありません。メインウィンドウの背景は、白 (0xffff) ですが、8 ビットのウィンドウが重なっている場合には、ピクセル値が 0xffff02 になります。

8 ビットウィンドウが重なっている場合は、その領域は 8 ビットウィンドウとして表示されます。8 ビットウィンドウを移動させるとすぐに、サーバーはその領域が 24 ビットウィンドウとして表示されるように、ウィンドウ ID を変更します。この処理の後に背景が修復されるため、一時的にこの領域にあった以前のピクセル値 (0xffff02) が、24 ビットのピクセル表示となります。このピクセル値は、青と緑として表示されるため、結果的にシアンとして表示されます。

---

注 – SX フレームバッファ上でも同様の現象が見られます。ただし、SX フレームバッファは、8 ビットのウィンドウを青のチャンネルに格納するため、一時的に表示される背景は、緑と赤で黄色となります。

---

黄色と白の色の違いは識別しにくいいため、SX フレームバッファでは、ユーザーがこの現象に気付くことはほとんどありません。ただし Creator アクセラレータ上のシアンと白の違いは、より顕著に見られます。ドラッグするウィンドウの一部分がイメージツールウィンドウの外側にある場合は、この状態が解消されるまでに多少時間がかかります。これは Creator のハードウェアが SX フレームバッファと比較して、保持する描画状態を多く持ち、またイメージツールのウィンドウの外側にドラッグした場合、ウィンドウマネージャーによってイメージツールのウィンドウのヘッダーが常に変更されるためです。さらにテキストの描画操作の間にコピー操作が行われる原因にもなり、描画操作とコピー操作は異なる描画状態を使用するため、Creator のハードウェアコンテキストに対する描画状態の読み込み・書き込みは連続的に行われる必要があり、Creator 上でのカラー領域の修復に要する時間は、SX と比較して長くなります。

上記の例では、イメージツールでの場合に特定して説明しましたが、背景が None に設定してあるウィンドウであれば、どの X ウィンドウでも同様です。(バグ ID 1215303)。

上記に示したような一時的なカラー障害は、背景が **None** に設定されているウィンドウに共通して発生します。この状態を回避するために、この種のウィンドウを使用しているアプリケーションでは、**None** 以外の背景を設定してください。X サーバーでは、このようなカラー障害を、発生後ただちに修復することができます。

## 第4章

# Creator グラフィックスアクセラレータにおける XIL の高速化

---

この章では、Creator または Creator3D グラフィックスアクセラレータに固有の XIL 関数について説明します。

XIL はサンの画像イメージとビデオのライブラリです。主要な XIL 関数では、UltraSPARC Visual Instruction Set (VIS) と Creator グラフィックスアクセラレータを使用する Ultra シリーズにおける動作が高速化されています。

---

## XIL データの種類

XIL は一般的な画像構造をサポートしていますが、VIS や Creator を使用した XIL の画像がすべて高速化されるわけではありません。VIS ポートがサポートする XIL データの種類を以下に示します。

- 1、2、3、4 バンドの符号なしバイト画像
- 1、2、3、4 バンドの符号付き簡略画像
- バンド調整された子画像
- 全画像処理領域
- XIL\_PIXEL\_SEQUENTIAL 型の画像

## 高速化された関数

VIS を使用して高速化された XIL 関数を以下に示します。画像分子は、1 バンドと 3 バンド XIL\_BYTE イメージに定義されています。

表 4-1 高速化された XIL 関数

関数	VIS 高速化		画像分子 の表示	その他	コメント
	8 ビット	16 ビット			
<code>xil_absolute</code>		1-4 バンド			
<code>xil_add</code>	1-4 バンド	1-4 バンド	x		
<code>xil_add_constant</code>	1-4 バンド	1-4 バンド	x		
<code>xil_affine</code>	1-4 バンド	1-4 バンド	x		注 6 参照
<code>xil_and</code>	1-4 バンド	1-4 バンド	x		
<code>xil_and_constant</code>	1-4 バンド	1-4 バンド	x		
<code>xil_band_combine</code>	1-4 バンド	1-4 バンド			
<code>xil_blend</code>	1-4 バンド	1-4 バンド			
<code>xil_cast</code>			x		
8 x 16	1-4 バンド	1-4 バンド			
16 x 8	1-4 バンド	1-4 バンド			
<code>xil_convolve</code>	1-4 バンド	1-4 バンド	x		注 1 参照
<code>xil_copy</code>	1-4 バンド	1-4 バンド	x		
<code>xil_decompress</code>					注 2、5 参照
<code>xil_get_pixel</code>					注 4 参照
<code>xil_lookup</code>	1-4 バンド	1-4 バンド	x	x	注 5 参照
<code>xil_max</code>	1-4 バンド	1-4 バンド	x		
<code>xil_min</code>	1-4 バンド	1-4 バンド	x		
<code>xil_multiply</code>	1-4 バンド	1-4 バンド	x		
<code>xil_multiply_constant</code>	1-4 バンド	1-4 バンド	x		
<code>xil_not</code>	1-4 バンド	1-4 バンド	x		
<code>xil_or</code>	1-4 バンド	1-4 バンド	x		

表 4-1 高速化された XIL 関数

関数	VIS 高速化		画像分子 の表示	その他	コメント
	8 ビット	16 ビット			
<code>xil_or_constant</code>	1-4 バンド	1-4 バンド	x		
<code>xil_rescale</code>	1-4 バンド	1-4 バンド	x	x	注 3、5 参照
<code>xil_rotate</code>	1-4 バンド	1-4 バンド	x		注 6 参照
<code>xil_scale</code>	1-4 バンド	1-4 バンド	x	x	注 5、6 参照
<code>xil_set_pixel</code>	1-4 バンド	1-4 バンド	x		注 4 参照
<code>xil_set_value</code>	1-4 バンド	1-4 バンド	x		
<code>xil_subtract</code>	1-4 バンド	1-4 バンド	x		
<code>xil_subtract_const</code>	1-4 バンド	1-4 バンド	x		
<code>xil_subtract_from_constant</code>	1-4 バンド	1-4 バンド	x		
<code>xil_tablewarp</code>			x		注 4 参照
<code>xil_threshold</code>	1-4 バンド	1-4 バンド	x	x	注 3、5 参照
<code>xil_transpose</code>	1-4 バンド	1-4 バンド	x		
<code>xil_xor</code>	1-4 バンド	1-4 バンド	x		
<code>xil_xor_const</code>	1-4 バンド	1-4 バンド	x		

## 表 4-1 に対する注意事項

注 1 — `xil_convolve` は、 $3 \times 3$ 、 $5 \times 5$ 、 $7 \times 7$  のカーネルにより高速化されています。

注 2 — `xil_decompress` は、以下の場合に高速化されます。

- `xil_decompress + color_convert` (JPEG 圧縮画像またはシーケンス、MPEG1 圧縮シーケンス)
- `xil_decompress + color_convert + display` (JPEG 圧縮画像またはシーケンス、MPEG1 圧縮シーケンス)
- `xil_decompress + color_convert + scale + display` (MPEG1 圧縮シーケンス)

このデコーダは、最高精度 24 ビット CCIR 601 YCC → RGB709 カラー空間変換に付着します。

以下の場合には、ストリームの伸張処理が高速化されません。

- JPEG または MPEG1 のストリームが、メモリー中で一連したバッファーとなっていない場合、たとえば `xil_cis_put_bits_ptr` によって分割されたストリームの伸張処理などは高速化されません。
- ストリーム中のフレームの 1 つが、何らかの理由 (たとえば伸張処理を実行中に、フレーム内で無効なデータが検出されるなど) によって高速化されない場合には、ストリーム中の後続のフレームの伸張処理も高速化されません。
- ストリーム中で、順方向または逆方向の検索を実行する場合は高速化されません。

注 3 — `xil_rescale + xil_threshold + xil_threshold +` 画像分子は、Creator のデプスキューイングハードウェアを使用して高速化されますが、以下の制限が適用されます。

- 第 1 のしきい値は 255 を上限値とします。また、下限値とマップ値を同一のものとしします。第 1 しきい値を  $N$  とすれば、 $N$  は  $0 \sim 255$  の間の任意の数となります。
- 第 2 のしきい値は  $0$  を下限値とします。また、上限値とマップ値を同一のものとしします。第 2 しきい値を  $M$  とすれば、 $M$  は  $0 \sim N$  間の任意の数となります。
- `xil_rescale` のパラメタ値は、変域 ( $x$ ) が  $0 \sim 255$  の時に、値域 ( $y$ ) に  $N$  および  $M$  の両方のしきい値が含まれるように設定する必要があります。これを図示すると、図 4-1 のようにパラメタ値の関数  $y = f(x)$  が、 $0 \leq x \leq 255$  の範囲内で直線  $y = N$ 、 $y = M$  の両方と交わる時に有効となります。
- 画像は、1 バンドの `XIL_BYTE` である必要があります。

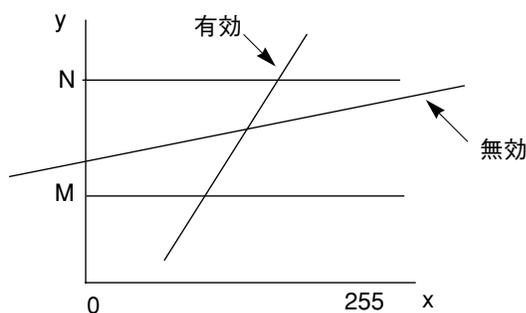


図 4-1 Creator のデプスキューイング

この分子を呼び出す、正しいコードの例を以下に示します。

```
float scale[1] = 1.5;
float offset[1] = -10;
float t1_lo[1] = 255;
float t1_hi[1] = 240;
float t1_map[1] = 240;
float t2_lo[1] = 0;
float t2_hi[1] = 15;
float t2_map[1] = 15;
xil_rescale(src,tmp1,scale,offset);
xil_threshold(tmp1,tmp2,t1_lo,t1_hi,t1_map);
xil_threshold(tmp2,tmp3,t2_lo,t2_hi,t2_map);
xil_display(tmp3,display);
```

同一の呼出しシーケンスが `scale[1] = .40` で行われた場合、 $y = 0.40 * x - 10$  では、 $y = -10 \sim y = (255 * 0.40) - 10 = 92$ 、 $x = 0 \sim x = 255$  の範囲にしか及ばないため、この分子は呼び出されません。この分子を実行するためには、 $x = 0 \sim x = 255$  に対して  $y = 15 \sim y = 240$  である必要があります。

注 4 — `xil_get_pixel`、`xil_set_pixel`、`xil_tablewarp` は、画像分子として扱われた場合のみ高速化されます。`xil_tablewarp` では、`interpolation = "nearest"` だけが高速化されます。

注 5 — 以下の分子は、すべて高速化されます (単一 xil 機能 + 表示を除く)。

- `xil_decompress + xil_colorconvert [+ display]` (JPEG 画像)
- `xil_decompress + xil_colorconvert [+ display]` (MPEG1 画像)
- `xil_decompress + xil_colorconvert + xil_scale + display` (MPEG1 画像)
- `xil_threshold + xil_threshold [+ display]`
- `[xil_rescale] + [xil_threshold] + [xil_threshold] + [xil_cast] + display`
- `xil_scale + xil_lookup + display` (高速化された分子は、 $8 \times 8$  ビットまたは  $16 \times 8$  ビットのルックアップに従う 8 ビットまたは 16 ビットのバイリニアスケールです。)

注 6 — `xil_affine`、`xil_scale`、`xil_rotate` の高速化には、いくつかの制限があります。`xil_affine` と `xil_rotate` は、最近傍補間処理をされた 1、3、4 バンドの画像およびバイリニア補間処理またはバイキュービック補間処理をされた 1 バン

ドと 3 バンドの画像に対してのみ高速化されます。 `interpolation = "general"` と設定した場合の `xil_scale` では、再サンプリングカーネルのサイズは、8 ビット画像に対して  $8 \times 8$ 、16 ビット画像に対して  $8 \times 4$  に制限されます。

---

## ダブルバッファのサポート

XIL 関数は、(Creator シリーズなど) ダブルバッファが搭載されているハードウェア上でのダブルバッファをサポートします。この節では、ダブルバッファのサポート機能と使用方法について説明します。

機能：

- `xil_create_double_buffered_window`
- `xil_set_active_buffer`
- `xil_get_active_buffer`
- `xil_swap_buffers`

表示画像をダブルバッファの表示画像として使用する場合は、

`xil_create_from_window` を使用するかわりに

`xil_create_double_buffered_window` を使用して画像を作成してください。ダブルバッファをサポートしていないハードウェア上で

`xil_create_double_buffer_window` を実行すると、無効なデバイスが返されます。開発者は、このことを理解して `xil_create_from_window` を呼び出す必要があります。

ダブルバッファをサポートするハードウェア上で

`xil_create_double_buffered_window` を実行して画像を作成した場合は、

`XilBufferId` が有効となり、`XIL_BACK_BUFFER` と `XIL_FRONT_BUFFER` の 2 つの値を選択することができます。このようなウィンドウでは、バッファのひとつが常時アクティブで、画像に書き込みをするすべての機能が、アクティブなバッファに書き込みます (たとえば、画像へのコピーは画像のアクティブなバッファへピクセルをコピーします)。一方、前面にあるバッファが画面に表示される部分ですが、作成時に実際にアクティブなバッファは `XIL_BACK_BUFFER` です。

`xil_create_double_buffered_window` でダブルバッファの画像を作成した後に、その画像を表示画像として使用する場合には、実際にアクティブなバッファが `XIL_BACK_BUFFER` であるため、特に必要な作業はありません。アクティブなバッ

ファーが `XIL_FRONT_BUFFER` である場合には、画像処理操作は失敗しませんが、その画像をダブルバッファの画像として使用することはできません。この画像を再びダブルバッファの画像として使用する場合は、以下のように呼び出してください。

```
# xil_set_active_buffer(dest,XIL_BACK_BUFFER)
```

これにより、フロントバッファからバックバッファへの変換が実行されます。この場合、ユーザーによる以下の呼び出しなしでは、画像の内容は表示されません。

```
# xil_swap_buffers(dest)
```

この操作により、画像の内容を見ることができるようになります。ダブルバッファの画像が不要になった場合には、以下のように呼び出してください。

```
# xil_set_active_buffer(dest,XIL_FRONT_BUFFER)
```

この操作により、ハードウェアの書き込みおよび表示バッファはデフォルト値にリセットされます。

以下にダブルバッファを使用する場合の、一般的なプログラム例を示します。

```
dest = xil_create_double_buffered_window(state,display,dwindow);
...
for (i=0; i<repeat; i++) {
...
xil_lookup(src,dest,lut);
xil_swap_buffers(dest);
}
xil_set_active_buffer(dest,XIL_FRONT_BUFFER);
```



## 第5章

# PGX グラフィックスアクセラレータ

---

この章では、PGX グラフィックスアクセラレータの画面解像度を変更する方法について説明します。

`m64config` ユーティリティを使用して、PGX および関連するグラフィックスハードウェアの画面解像度を変更することができます。オプションはコマンド行で指定されます。指定されたオプションを `OWconfig` ファイルに保存して、ウィンドウシステムのセッションやシステムの再起動の後も持続させることが可能です。

`m64config` ユーティリティを使用して、以下の指定が可能です。

- ビデオモード (画面解像度とリフレッシュレート)
- `OWconfig` ファイルの更新
- すべてのオプションの値をデフォルトの値にリセット
- `OWconfig` ファイルに保存されているすべての PGX のオプションの現在値を表示
- PGX のハードウェアの構成を表示

---

注 - `m64` は、PGX グラフィックスアクセラレータの UNIX デバイス名です。

---

---

## サポートする画面解像度

表 5-1 に、PGX グラフィックスアクセラレータがサポートする画面解像度を示します。

表 5-1 PGX がサポートする画面解像度

画面解像度	垂直 リフレッシュレート	説明
1600 x 1000	76 Hz	ノンインタレース
1600 x 1000	66 Hz	ノンインタレース
1440 x 900	76 Hz	ノンインタレース
1280 x 1024	76 Hz	ノンインタレース
1280 x 1024	75 Hz	ノンインタレース
1280 x 1024	67 Hz	ノンインタレース
1280 x 1024	60 Hz	ノンインタレース
1280 x 800	76 Hz	ノンインタレース
1152 x 900	76 Hz	ノンインタレース
1152 x 900	66 Hz	ノンインタレース
1024 x 768	75 Hz	ノンインタレース
1024 x 768	70 Hz	ノンインタレース
1024 x 768	60 Hz	SVGA (ノンインタレース)
800 x 600	75 Hz	ノンインタレース
768 x 575	50 Hz	PAL (インタレース)
640 x 480	60 Hz	NTSC (インタレース)

## 画面解像度を一時的に変更する

特定の解像度をモニターがサポートしているかどうかを確認する目的などで、画面解像度を一時的に変更する方法を以下に示します。

```
# /usr/sbin/m64config -res ビデオモード try
```

表 5-2 に、ビデオモードのオプションを示します。ビデオモードを確認し、10 秒以内に **y** を入力してください。

表 5-2 PGX の画面解像度の形式

ビデオモード		
幅 x 高さ x 垂直周波数	記号名	解像度
1600x1000x76		1600 x 1000、76 Hz
1600x1000x66		1600 x 1000、66 Hz
1440x900x76		1440 x 900、76 Hz
1280x1024x76	1280	1280 x 1024、76 Hz
1280x1024x75		1280 x 1024、75 Hz
1280x1024x67		1280 x 1024、67 Hz
1280x1024x60		1280 x 1024、60 Hz
1280x800x76		1280 x 800、76 Hz
1152x900x76	1152	1152 x 900、76 Hz
1152x900x66		1152 x 900、66 Hz
1024x768x75		1024 x 768、75 Hz
1024x768x70		1024 x 768、70 Hz
1024x768x60	svga	1024 x 768、60 Hz
800x600x75		1024 x 600、75 Hz
768x575x50i	pal	768 x 575、50 Hz、インタレース
640x480x60i	ntsc	640 x 480、60 Hz、インタレース
	none	デバイスに合わせてプログラムされた現在のビデオモード

たとえば画面解像度を 1152 x 900、76 Hz にする場合には、以下の例のどちらかを入力してください。

以下に、幅 x 高さ x 垂直周波数の形式の使用例を示します。

```
# /usr/sbin/m64config -res 1152x900x76 try
```

以下に、記号名形式の使用例を示します。

```
# /usr/sbin/m64config -res 1152 try
```

---

## PGX のハードウェア構成の表示

- PGX のハードウェア構成の情報を表示するには、以下のように入力してください。

```
# /usr/sbin/m64config -prconf
```

ハードウェア構成の情報を表す典型的な表示例を、以下に示します。

```
--- Hardware Configuration for /dev/fbs/m640 ---
ASIC: version 0x41004754
DAC: version 0x0
PROM: version 0x0
Card possible resolutions: 640x480x60, 800x600x75, 1024x768x60
    1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
    1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
    960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i, 1280x800x76
    1440x900x76, 1600x1000x66, 1600x1000x76, vga, svga, 1152, 1280
Monitor possible resolutions: 720x400x70, 720x400x88, 640x480x60
    640x480x67, 640x480x72, 640x480x75, 800x600x56, 800x600x60
    800x600x72, 800x600x75, 832x624x75, 1024x768x87, 1024x768x60
    1152x900x76, 1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
    stereo
Current resolution setting: 1280x1024x76
Current depth: 8
```

---

## 詳細情報

この章では、[m64config](#) ユーティリティーの簡単な例を 2 つ紹介しているだけです。[m64config](#) についての詳細は、[m64config](#) のマニュアルページを参照してください。

---

## PEX ライブラリバグの回避策

3D アプリケーションを PGX フレームバッファ上で実行することによって、不正確な描画やウィンドウシステムのクラッシュなどの障害が発生する場合があります。

これらの障害を回避するには、3D アプリケーションを実行する前に XGLNOPEX 環境変数を設定します。

```
setenv XGLNOPEX
```

この環境変数には、特定の値を入力しないでください。



## 第6章

### PGX32 グラフィックスアクセラレータ

---

この章では、PGX32 グラフィックスアクセラレータの画面解像度を変更する方法について説明します。

メニュー式インタフェースユーティリティの `GFXconfig` を使用して、いつでも画面解像度を変更することができます。詳細については、`GFXconfig` のマニュアルページを参照してください。

---

注 - `gfx` は、PGX32 グラフィックスアクセラレータの UNIX デバイス名です。

---

---

#### 対話形式による設定

- `GFXconfig` を使用して PGX32 カードを設定するには、以下のように入力します。

```
# GFXconfig -i
```

PGX32 設定ウィンドウが表示されます (図 6-1)。

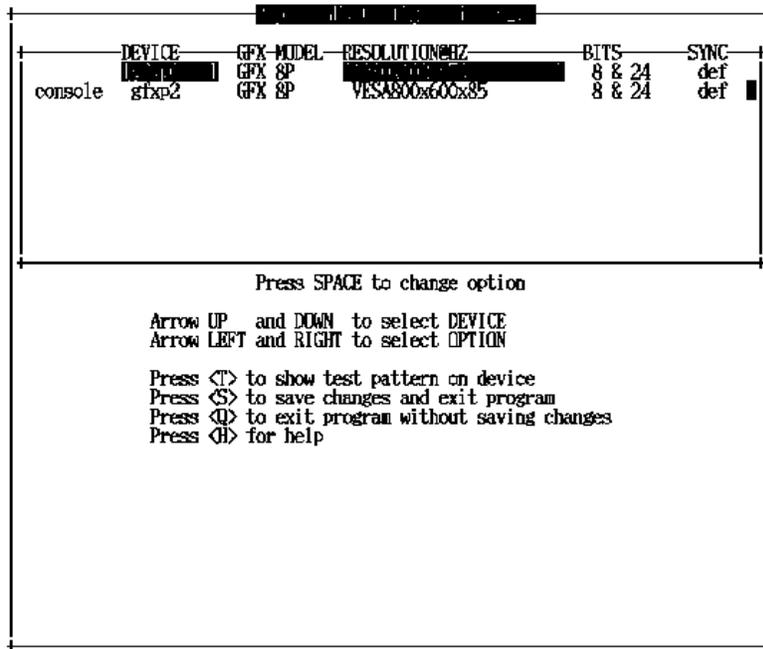


図 6-1 PGX32 設定ウィンドウ

PGX32 設定ウィンドウの操作方法は、下記の表 6-1 を参照してください。

表 6-1 PGX32 設定ウィンドウ

機能	説明
上向きと下向きの矢印	設定を変更する PGX32 デバイスを選択します。
左向きと右向きの矢印	設定を変更するパラメタ (たとえば画面解像度、ビットの深さ、同期など) を選択します。
スペースバー	選択した PGX32 デバイスの、パラメタの変更に使用する (変更可能な場合はメニューが表示されます)。
T	ディスプレイ全体にテストパターンを表示します。いずれかのキーを押せば、メイン画面に戻ります。
S	現在の設定を保存してから、設定ウィンドウを終了します。
H	ヘルプ
Q	変更を保存しないで、プログラムを終了します。

## 非対話形式による設定

特に多数のシステムに同一内容の設定をする場合や、システムに最適な設定がわかっている場合などは、PGX32 カードを非対話形式で設定したほうが便利です。

`GFXconfig` は、`m64config` ユーティリティと同じ規約に従います。

`m64config` は、Sun Ultra 5 と Sun Ultra 10 のマザーボードグラフィックス (8 ビットシステムと 24 ビットシステムの両方) や、PGX 8 ビット PCI フレームバッファなどの、すべての ATI ベースのグラフィックスで使用されています。対話方式による設定方法で使用するすべてのパラメータは、非対話形式でも使用できます。この場合は、適切なフラグの後に必要な値を続けて指定します。各パラメータの詳細は、表 6-2 を参照してください。

表 6-2 非対話形式の設定で使用するパラメータ

パラメータ	説明
<code>-dev</code> デバイス	構成するデバイスを表示します。
<code>-res</code> 解像度	解像度名を表示します。
<code>-res \?</code>	解像度を表示します。
<code>-file</code> ファイル名	設定ファイル ( <code>system</code> または <code>machine</code> ) を表示します。
<code>-depth</code> 深さ	ビットの深さ (8 または 24、デフォルトは 24) を表示します。
<code>-defaults</code>	デバイスのパラメータをデフォルトにリセットします。
<code>-24only</code> ( <code>true/false</code> )	すべてのウィンドウで 24 ビット画像を使用するように強制設定します。8 ビットアプリケーションの中には、この指定により正常に動作しなくなるものもあります。
<code>-gfile</code> ガンマファイル	ガンマファイルを一覧表示します。
<code>-gvalue</code> ガンマ値	ガンマ値を一覧表示します。
<code>-propt</code>	現在の設定を表示します。
<code>-prconf</code>	ハードウェア情報を表示します。
<code>-help</code>	使用方法 (ヘルプメッセージ) を表示します。

---

注 - デフォルトでは、解像度が 1280 × 1024 以下の場合のビット深度は 8/24 に、それよりも高解像度の場合は 8 に、それぞれ設定してあります。

---

## 使用例

- PGX32 の解像度を 1152 × 900、66 Hz に設定するには、以下のように入力します。

```
# GFXconfig -res 1152x900x66
```

解像度を確定する前に確認しておきたい場合は、解像度名の後に「try」という語を追加します。このオプションを指定すると、画面にテストパターンが表示されます。リターンキーを押すとテストパターンは解除されます。このパターンで実際に解像度を確認してから、その解像度を適用するかどうか判断します。入力例を以下に示します。

```
# GFXconfig -res 1152x900x66 try -dev /dev/fbs/gfxp0
```

- ツールカラー画像のみを使用して (8 ビット擬似カラーを使用しないで) 解像度を 1024 × 768、60 Hz に設定するには、以下のように入力します。

```
# GFXconfig -res 1024x768x60 -24only true
```

- `/dev/fbs/gfxp0` の現在の設定を確認するには、以下のように入力します。

```
# GFXconfig -propt -dev /dev/gfx/gfxp0
```

---

## 画面解像度のその他の変更方法

通常、ほとんどの場合デフォルトの画面解像度で十分です。ただし、以下の場合はデフォルトの解像度を変更する必要があります。

- X Window の深さをデフォルトから変更し、デフォルトとして用意されている値ではなくなった場合、画面の深さと X Window の深さが一致するように設定する必要があります。
- デフォルトの画面解像度ではモニターが正常に同期しない場合、画面解像度を別の値に変更する必要があります。

## ガイドライン

画面解像度をデフォルトの値から変更する際には、以下のガイドラインに従ってください。

- 8/24 モードで X Window 環境を動作させるには、画面解像度の深さを 24 ビットに設定します。
- デフォルトでは、画面解像度が 1280 × 1024 以下の場合は自動的に 24 ビットに設定されます。それよりも高い解像度の場合は、8 ビットモードに設定されます。
- 画面解像度を確定する前に、`GFXconfig -i` を使用して解像度のテストをしてください。

## 変更方法

対話形式、非対話形式以外の変更方法には、以下のようなものがあります。それぞれの方法について解説します。

- EDID 自動検出 (Auto-Detect) 機能
- output device を使用する方法
- ビデオモードを使用する方法
- ビデオタイミングを使用する方法

## PGX32 用の EDID 自動検出機能

DDC2B/EDID プロトコルを使用しているモニターの場合、デフォルトの解像度は自動検出機能を使って決定できます。このプロトコルを利用した場合、まず GFX が標準タイミング識別子 (Standard Timing Identifiers) を調べてサポート順位の最も高い解像度を検索し、これと設定タイミング (Established Timing) を照合します。この方法が失敗した場合は、カードの解像度は自動的に 1152 × 900、66 Hz に設定されます。

---

注 – モニターの電源をシステムの起動前に入れておかないと、PGX32 はモニターを正常に認識できません。

---

この方法を使用した場合、EDID を介して得た情報はすべて無効になります。

## output-device を使用する方法

PGX32 カードの解像度は、`output-device` 環境変数からも設定できます。この変数は `screen:rAxBxC` (A は水平解像度、B は垂直解像度、C はリフレッシュレート) という形式で記述します。

システムはここに代入された値を解像度の内部リスト (表 6-3 参照) と照合して、その値に対応するエントリを画面解像度として使用します。

たとえば、画面解像度を VESA 1024 × 768、75 Hz に設定する場合は、`ok` プロンプトで以下のように入力します。

```
ok setenv output-device screen:r1024x768x75
ok reset
```

---

注 – 新しい画面解像度はリセットするまで有効になりません。またリセット後は再度 `output-device` 変数を手動で変更するまで、この解像度が維持されます。

---

## ビデオモードを使用する方法

Boot PROM モードの `ok` プロンプトでは、インストール済みの 34 種類の解像度モードのうちの 1 つを選択するだけで、PGX32 カードの画面解像度を簡単に設定できます。解像度の設定値は 0~33 のビデオモードの形式で定義してあります (表 6-3)。

---

注 - 画面の深さが 24 ビットの時はビデオモードの 0~25 を、8ビットの時は 26~33 を使用します。

---

表 6-3 PGX32 の画面解像度

モード	解像度
0	640 × 480 @ 60
1	640 × 480 @ 72
2	640 × 480 @ 75
3	640 × 480 @ 85
4	800 × 600 @ 60
5	800 × 600 @ 72
6	800 × 600 @ 75
7	800 × 600 @ 85
8	1024 × 768 @ 60
9	1024 × 768 @ 70
10	1024 × 768 @ 75
11	1024 × 768 @ 77 *
12	1024 × 768 @ 85
13	1024 × 800 @ 85 *
14	1152 × 900 @ 60
15	1152 × 900 @ 66 *
16	1152 × 900 @ 70
17	1152 × 900 @ 75
18	1152 × 900 @ 76 *
19	1152 × 900 @ 85
20	1280 × 800 @ 76 *
21	1280 × 1024 @ 60
22	1280 × 1024 @ 67 *
23	1280 × 1024 @ 75
24	1280 × 1024 @ 76 *

表 6-3 PGX32 の画面解像度 (続き)

モード	解像度
25	1280 × 1024 @ 85
26	1600 × 1200 @ 66 *
27	1600 × 1200 @ 76 *
28	1600 × 1200 @ 60
29	1600 × 1200 @ 65
30	1600 × 1200 @ 70
31	1600 × 1200 @ 75
32	1600 × 1200 @ 76
33	1600 × 1200 @ 80 *

注 - `nvedit` コマンドについては、74 ページの「`nvedit` を使用して NVRAM を修正する」を参照してください。

たとえば、画面解像度を 1024 × 768、60 Hz (ビデオモードの 8) に設定するには、以下のように入力します。

```
ok nvedit
  0: 8 value video-mode  1: <ctrl-c>
ok nvstore
ok setenv use-nvramrc? true
ok reset
```

注 - 最後の 3 行のコマンドにより、NVRAM は有効になります。この 3 行を実行しない場合、`nvedit` で設定した変更内容は無視されます。

## ビデオタイミングを使用する方法

前述のいずれの方法もうまくいかない場合は、最終手段として画面解像度のタイミング数を厳密に指定する方法があります。この方法でも `nvedit` を使用します。他の方法に比べて複雑で、目的の解像度を設定するにはすべてのタイミングパラメタに関する

る知識が必要です。前述のビデオモードに用意されていない解像度を設定する必要がある場合のみ、この方法を使用してください。nvedit コマンドについては、74 ページの「nvedit を使用して NVRAM を修正する」を参照してください。

---

注 – この方法は、前述のいずれの方法も成功しなかった場合のみ使用してください。

---

たとえば、画面解像度を 1280 × 1024、76 Hz に設定するには、以下のように入力します。

```
ok nvedit
  0: : video-timing " 1280, 384, 32, 64, \
    1024, 43, 3, 8, 135000000, 0" ;
  1: <ctrl-c>
ok nvstore
ok setenv use-nvramrc? true
ok reset
```

---

注 – この入力では構文が非常に重要な意味を持ちます。例に示しているように、スペースを正確に入力する必要があります。

---

---

注 – 最後の 3 行のコマンドにより、NVRAM は有効になります。この 3 行を実行しない場合、nvedit で設定した変更内容は無視されます。

---

この方法では、以下の 10 種類のパラメタを設定する必要があります。

- 水平方向の解像度 (ピクセル単位)
- 水平方向の空白の合計
- 水平方向のフロントポーチ
- 水平同期幅
- 垂直解像度 (走査線単位)
- 垂直方向の空白の合計
- 垂直方向のフロントポーチ
- 垂直同期幅
- ドットクロック (Hz 単位)

- 同期値 (表 6-4 参照)。複数の同期値を選択する場合は、値の和をとって使用します。

表 6-4 同期値

同期値	意味
1	独立同期
1	緑色の同期
512	正の垂直方向同期パルス
1024	正の水平方向同期パルス
2048	複合同期

注 – この方法で必要となるタイミングパラメタについては、ご使用のモニターの機種など必要事項を確認した上で、サン・サービス (1-800-USA-4SUN) にお問い合わせください。

## PGX32 をコンソールとして設定する (オプション)

PGX32 ソフトウェアをコンソールデバイスとして使用するには、この節の以下の手順に従ってください。

### フレームバッファに PGX32 カードだけを使用する場合

#### Sun Ultra 5 システムと Sun Ultra 10 システム

Sun Ultra 5 または Sun Ultra 10 システムで PGX32 カードをシステムコンソールとして使用し、このカードをシステムの唯一のフレームバッファとして使用するには、まずこのシステムに標準装備されている 8 ビットまたは 24 ビットのオンボードグラフィックスデバイスを無効にします。

- マザーボードに組み込まれている 8 ビットまたは 24 ビットのグラフィックスデバイスを無効にするには、以下のように入力します。

```
ok setenv pcib-probe-list 1,3
ok reset
```

システムのリセット後は、すべてのコンソールメッセージが PGX32 カードに送られます。

- マザーボード上の 8 ビットまたは 24 ビットのグラフィックスデバイスをコンソールに復元する場合は、以下のように入力して `pcib-probe-list` にオンボードグラフィックスデバイスを再度追加します。

```
ok setenv pcib-probe-list 1,2,3
ok reset
```

## Sun Ultra 30 システムと Sun Ultra 60 システム

Sun Ultra 30 および Sun Ultra 60 では、システム内に他のフレームバッファが存在しない場合、有効な PCI スロットに PGX32 ポートが設置されていれば、自動的にその PGX32 カードがデフォルトのコンソールになります。

## PGX32 カードと二次フレームバッファを併用する場合

システムに他の二次フレームバッファがある場合は、PGX32 カードをコンソールデバイスに指定して併用できます。

## オンボードグラフィックス (Ultra 5 と Ultra 10 のみ)

Sun Ultra 5 および Sun Ultra 10 の標準のオンボードグラフィックスと PGX32 カードは、オンボードグラフィックスデバイスの方をコンソールとして使用する場合のみ、同一システム内で共存できます。PGX 32 カードの方をコンソールとして使用する場合は、68 ページの「フレームバッファに PGX32 カードだけを使用する場合」の手順に従って、オンボードグラフィックスデバイスを無効にする必要があります。

## UPA バスフレームバッファを備えたシステム

UPA フレームバッファが存在するシステムで PGX32 カードをコンソールとして設定するには、NVRAM の `output-device` 変数を、コンソールにする PGX32 カードの実際のパスに変更する必要があります。このパスを調べるには、`ok` プロンプトから / ツリー下の `TSI` 文字列を検索するのが最良の方法です。

たとえば、`pci` デバイスを探すには、以下のように入力します。

```
ok cd /pci@1f,4000
ok ls
```

現在の位置さえ適切であれば、文字列 `TSI` を含むエントリ (`TSI,gfxp@#` というエントリで、`#` の部分には PGX32 のスロットの位置を表す番号が入る) が少なくとも 1 つ表示されます。

このエントリを PGX32 カードのコンソールデバイスとして使用します。たとえば、デバイス `TSI,gfxp@#` へのパスが上記のように `/pci@1f,4000` である場合は、以下のように入力します。

```
ok setenv output-device /pci@1f,4000/TSI,gfxp@#
ok reset
```

---

注 - `#` の部分は、ご使用のシステムの PGX32 デバイスに対応する値で置き換えてください。

---

システムをリセットすると、すべてのコンソールメッセージは PGX32 カードに送られます。

何らかの理由でデフォルトのグラフィックスデバイスをコンソールに復元する場合は、`output-device` 変数を画面のデフォルト値に戻してください。以下のように入力します。

```
ok setenv output-device screen
ok reset
```

## その他の PCI フレームバッファ

システム内に他の PCI フレームバッファが存在する状態で PGX32 をコンソールデバイスに設定するには、68 ページの「フレームバッファに PGX32 カードだけを使用する場合」で解説した変更事項に加えて、二次フレームバッファのスロットより前に PGX32 のスロットを検索するように `pcia-probe-list` を変更する必要があります。

- フレームバッファのスロット番号を調べてから、`pcia-probe-list` で指定しているスロット番号の検索順序を、PGX32 のスロット番号が二次フレームバッファのスロット番号よりも前になるように設定します。

たとえば、PGX32 がスロット 3、二次フレームバッファがスロット 1 にある場合は、スロット 3 がスロット 1 よりも先に検索されるように `pcia-probe-list` を更新します。例えば以下のような検索順序であれば適切です。

```
ok setenv pcia-probe-list 3,2,1,4
ok reset
```

システムをリセットすると、すべてのコンソールメッセージは PGX32 カードに送られます。

---

## デスクトップ環境の起動

この節では、PGX32 カードで OpenWindows 環境、共通デスクトップ環境 (CDE)、X デisplayマネージャを起動する方法について説明します。

### OpenWindows 環境

以下の節では、OpenWindows 環境をコンソールとして起動する方法、および複数の PGX32 カードで起動する方法について説明します。PGX32 のデバイス名は、`gfxp#` です。

## PGX32 カードをコンソールとして使用する

- PGX32 カードをコンソールに設定している場合は、以下のように入力します。

```
# openwin
```

## 複数の PGX32 カードを使用する

- 2 つの PGX32 デバイス (たとえば `gfxp0` と `gfxp1`) で OpenWindows 環境を起動するには、以下のように入力します。

```
# openwin -dev /dev/fbs/gfxp0 -dev /dev/fbs/gfxp1
```

---

注 - 上の例では `gfxp` のデバイス番号を 0 と 1 にしていますが、この数字は実際のシステムの設定により異なります。ご使用のシステムごとに `/dev/fbs/` または `dmesg` を確認して、正しいデバイス番号を入力してください。

---

## 共通デスクトップ環境 (CDE)

CDE をインストール済みの環境であれば、`/usr/dt/config/Xservers` ファイルの修正により、CDE を PGX32 ディスプレイ上に表示できます。ZX カードがコンソールデバイスである場合は、`Xservers` ファイルを変更する必要はありません。

以下の例の `Xservers.gfx` ファイルでは ZX カードを、CDE を起動する唯一のフレームバッファに設定しています。

```
:0 Local Local_uid@console root /usr/openwin/bin/Xsun \  
:0 -dev /dev/fbs/gfxp0 -nobanner
```

---

注 - 実際の ZX デバイスの名前が `gfxp0` 以外である場合は、このファイル中のデバイス番号をご使用のシステムで設定している適切な値に置きかえる必要があります。

---

このファイルの最終行には、必要に応じてその他のコマンド行引数を追加できます。たとえば、CDE を複数のディスプレイで起動するような設定も可能です。

- 具体的には、上記のコマンドライン規約に続けて、ディスプレイデバイスを追加記述します。

以下の例では、`/dev/fbs/gfxp0` というデバイス名のディスプレイに CDE を表示し、`/dev/fbs/m640` (これは Sun Ultra 5 システムおよび Sun Ultra 10 システムに組み込みのグラフィックスデバイスです) を二次フレームバッファとして使用するよう設定しています。

```
:0 Local Local_uid@console root /usr/openwin/bin/Xsun \  
:0 -dev /dev/fbs/gfxp0 -dev /dev/fbs/m640
```

## X ディスプレイマネージャ

ZX カードは X ディスプレイマネージャ (`xdm`) もサポートします。簡潔な設定ファイルが `/usr/openwin/lib/X11/xdm/Xservers` として提供されます。

すでに適切な位置に `Xservers` ファイルが配置されていれば、ZX ソフトウェアのインストール時に `/usr/openwin/lib/X11/xdm/Xservers.nogfx` として保存されます。

デフォルトではこのインストール時に以下の行が追加され、これにより ZX は `xdm` を起動する唯一のフレームバッファになります。

```
:0 Local Local /usr/openwin/lib/xdm/StartOW \  
:0 -dev /dev/fbs/gfxp0
```

この最終行に続けて、必要に応じてその他のコマンド行引数を追加できます。たとえば、複数のディスプレイで `xdm` を起動するような設定も可能です。

- 具体的には、上記のコマンドライン規約に続けてディスプレイデバイスを追加指定します。

以下の例では、`/dev/fbs/gfxp0` というデバイス名のディスプレイに `xdm` を表示し、`/dev/fbs/m640` (Sun Ultra 5 システムおよび Sun Ultra 10 システムの組み込みのグラフィックスデバイス) を二次フレームバッファとして使用するよう設定しています。

```
:0 Local Local_uid@console root /usr/openwin/bin/Xsun \  
:0 -dev /dev/fbs/gfxp0 -dev /dev/fbs/m640
```

## nvedit を使用して NVRAM を修正する

- NVRAM を修正するには、次のように `ok` プロンプトで `nvedit` エディタを起動します。

```
ok nvedit
```

`nvedit` エディタの使用方法については、66 ページの「ビデオタイミグを使用する方法」を参照してください。NVRAM の変数の編集に使用するキーシーケンスを以下に示します。

表 6-5 NVRAM エディタ

キーシーケンス	説明
バックスペース	カーソルの前方の文字を削除する
<code>ctrl-l</code>	NVRAM の現在の値を一覧表示する
<code>ctrl-p</code>	前の行に移動する
<code>ctrl-n</code>	次の行に移動する
<code>ctrl-b</code>	前の文字に移動する
<code>ctrl-l</code>	行の先頭までを削除する
<code>ctrl-k</code>	現在の行と次の行を結合する
<code>ctrl-u</code>	現在の行を削除する
<code>ctrl-c</code>	<code>nvr</code> エディタを終了する ( <code>ok</code> プロンプトに戻る)

ここで変更した内容は、`ok` プロンプトで `nvstore` コマンドを利用して保存した場合にのみ有効になります。保存後は、この変更内容をシステムが使用する前に NVRAM を有効にしておく必要があります。環境変数の `use-nvramrc?` を `true` に設定しておいてください。

## 第7章

# Elite3D グラフィックスアクセラレータ

---

この章では、Elite3D グラフィックスアクセラレータの画面解像度を、モニターの種類に合わせて変更する方法について説明します。

`afbconfig` ユーティリティを使用して、Elite3D X11 の画面および対応するグラフィックスハードウェアを変更することができます。オプションはコマンド行で指定します。指定したオプションは、`OWconfig` ファイルに保存されます。これらのオプションは、次に Elite3D デバイスで `Xsun` を実行する際に、そのデバイスを初期化するために使用されます。`OWconfig` ファイルのオプションを変更した場合に、これらのオプションは `Xsun` セッションやシステムの再起動の後も持続します。

---

注 – `afb` は、グラフィックスアクセラレータ Elite3D ファミリの UNIX デバイス名です。

---

`afbconfig` ユーティリティによって、以下の指定が可能です。

- ビデオモード (画面解像度とリフレッシュレート)
- 画像の種類 (リニアまたは非リニア)
- 8 ビット擬似カラー画像 (オーバーレイ画像) を使用するかどうか
- 表示画像リスト中で、リニア画像を非リニア画像より前に表示させるかどうか
- デフォルトの表示画像の設定方法
- OpenGL 画像のサポート方法
- サーバーオーバーレイ画像 (SOV) が有効かどうか
- `WID` として使用するために予約された Elite3D X チャンネルピクセルの最大数
- 表示画像リストで、擬似カラーのオーバーレイ画像を擬似カラーのアンダーレイ画像より前に表示させるかどうか
- ガンマの値、またはガンマ補正テーブルを含むファイルの指定による構成可能ガンマ補正
- 256 色フルカラーオーバーレイおよびハードウェアにより高速化された透過画像を実現する拡張オーバーレイオプション

---

## デフォルトの画面解像度

Elite3D システムは、デフォルトの画面解像度を調べるために VESA 標準 EDID (Extended Display Identification Data) をモニタから読み取ります。そのモニタの EDID が分からない場合は、モニタ ID センスコードを使ってデフォルトの画面解像度を調べます。

表 7-1 に、モニター ID センスコードに対応するデフォルトの画面解像度を示します。

表 7-1 Elite3D グラフィックアクセラレータのモニターセンスコード

コード	画面解像度
7	1152 x 900、66 Hz
6	1152 x 900、76 Hz
5	1024 x 768、60 Hz
4	1280 x 1024、67 Hz
3	1152 x 900、66 Hz
2	1280 x 1024、76 Hz
1	1152 x 900、66 Hz
0	1024 x 768、77Hz

サン以外のモニターなど、Elite3D のシステムがモニターの種類を判断できない場合は、1152 x 900、66 Hz の解像度をデフォルトとします。

## サポートする画面解像度

表 7-2 に、Elite3D グラフィックスアクセラレータがサポートする画面解像度を示します。

表 7-2 Elite3D がサポートする画面解像度

画面解像度	垂直リフレッシュ レート	説明	ビデオモード形式	シンボリック名
1280 x 1024	76 Hz	ノンインタレース	1280 x 1024 x 76	1280
1280 x 1024	67 Hz	ノンインタレース	1280 x 1024 x 67	
1280 x 1024	60 Hz	ノンインタレース	1280 x 1024 x 60	
1280 x 1024	85 Hz	ノンインタレース	1280 x 1024 x 85	
1280 x 800	76 Hz	ノンインタレース	1280 x 800 x 76	
1152 x 900	76 Hz	ノンインタレース	1152 x 900 x 76	1152
1152 x 900	66 Hz	ノンインタレース	1152 x 900 x 66	
1024 x 800	84 Hz	ノンインタレース	1024 x 800 x 84	
1024 x 768	77 Hz	ノンインタレース	1024 x 768 x 77	
1024 x 768	75 Hz	ノンインタレース	1024 x 768 x 75	
1024 x 768	70 Hz	ノンインタレース	1024 x 768 x 70	
1024 x 768	60 Hz	SVGA	1024 x 768 x 60	svga
960 x 680	112 Hz	立体表示、ノンインタレース、 56 Hz フィールドレート	960 x 680 x 112s	stereo
960 x 680	108 Hz	立体表示、ノンインタレース、 54 Hz フィールドレート	960 x 680 x 108s	
768 x 575	50 Hz	インタレース - PAL	768 x 575 x 50i	pal
640 x 480	60 Hz	インタレース - NTSC	640 x 480 x 60i	ntsc
640 x 480	60 Hz	ノンインタレース	640 x 480 x 60	

モニターの種類によっては、Elite3D システムがサポートする解像度をサポートしないものがあります。Elite3D がサポートする解像度と接続されているモニターは、[afbconfig](#) コマンドで確認することができます。

サポートされる解像度でモニターが確実に同期するためには、`fbconfig` を実行する前に以下の Solaris のパッチをインストールする必要があります。表 7-3 には、各バージョンの Solaris に必要なパッチが示されています。

表 7-3 Elite3D モニター同期のための Solaris のパッチ

Solaris のバージョン	パッチ
Solaris 2.5.1	105791-30
Solaris 2.6	105363-29
Solaris 2.7	-
- ffb パッチ	106144-18
- xfb パッチ	106148-11
Solaris 8	N/A

注 - ffb パッチと xfb パッチが必要な場合、両方のパッチをペアでインストールする必要があります。

## 画面解像度を変更する (`-res`)

### ▼ Elite3D がサポートする解像度と接続されているモニターを確認する

- `afbconfig` コマンドを以下のように入力します。

```
# afbconfig -res \?
```

特定の解像度をモニターがサポートしているかどうかを確認するために、画面解像度を一時的に変更することができます。



注意 - ウィンドウシステムの実行中に、画面解像度を変更しないでください。画面表示に異常が発生する場合があります。

## ▼ 画面解像度を一時的に変更する

- `afbconfig` コマンドを以下のように入力します。

```
# afbconfig -res ビデオモード try
```

ビデオモードのオプションについては表 7-2 の「ビデオモード形式」と「シンボリック名」の欄を参照してください。ビデオモードを確認し、そのビデオモードを使用する場合は、5 秒以内に `y` と入力してください。

## ▼ 画面解像度を立体表示に変更する

- 以下のように入力します。

```
# afbconfig -res stereo
```

上記の例では、次回に `Xsun` を実行したときの画面解像度を、960 x 680、112 Hz の立体表示に変更します。

---

## 画像表示形式リストを変更する

`afbconfig` コマンドを使用して、X サーバーの画像表示形式リストの順序を変更することができます。表 7-4 に示す `afbconfig` オプションを使用して、特定のデバイスに合わせて参照された画像の表示形式リストを設定してください。

表 7-4 `afbconfig` の画像フラグのデフォルトの設定

名称	値	Solaris 2.5.1 または 2.5.1 SHWP の	
		デフォルト	Solaris 2.6 の デフォルト
<code>linearorder</code>	<code>first/last</code>	<code>last</code>	<code>last</code>
<code>deflinear</code>	<code>true/false</code>	<code>false</code>	<code>false</code>
<code>overlayorder</code>	<code>first/last</code>	<code>last</code>	<code>last</code>

表 7-4 `afbconfig` の画像フラグのデフォルトの設定

名称	値	Solaris 2.5.1 または	
		2.5.1 SHWP の デフォルト	Solaris 2.6 の デフォルト
<code>defoverlay</code>	<code>true/false</code>	<code>false</code>	<code>false</code>
<code>expvis</code>	<code>enable/disable</code>	<code>disable</code>	<code>enable</code>
<code>sov</code>	<code>enable/disable</code>	<code>disable</code>	<code>enable</code>

### 画像表示形式リストの順序を変更する (`-linearorder`、`-overlayorder`)

デフォルトでは、非リニア画像は、画像表示形式リスト上でリニア画像より前に表示されます。`afbconfig` コマンドを使用して、画像表示形式リストの順序を変更することができます。

3D アプリケーションの多くはリニア画像を使用しますが、特定の 3D アプリケーションでは、`XSolarisGetVisualGamma` (3) を使用するリニア画像を検索しないで、リストに最初に表示される 24 ビットトゥルーカラーを使用します。このようなアプリケーションを、正しい画像表示形式で動作させるには、リストの並べ換えが必要です。`-linearorder` オプションを使用して、リニア 24 ビットトゥルーカラーが最初に表示されるように指定してください。

画像表示形式リストの並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する方法は、`-linearorder` オプションの 1 つを指定して `afbconfig` コマンドを入力します。

以下に例を示します。

```
# afbconfig -linearorder first
```

デフォルトの画像表示形式リストでは、8 ビット疑似カラー画像が 8 ビット疑似カラーオーバーレイ画像よりも前になっています。この画像表示形式の順序を、`afbconfig` コマンドを使用して変更することができます。

8 ビット疑似カラーオーバーレイ画像を使用する特定のアプリケーションは、リストに最初に表示される 8 ビット疑似カラーオーバーレイ画像を使用します。このようなアプリケーションを正しい画像表示形式で動作させるには、8 ビット疑似カラーオーバーレイ画像が、画像表示形式リストの最初に表示されるように、`-overlayorder` オプションを使用して画像表示形式リストの順序を変更します。

画像表示形式リスト内の画像表示形式の並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する方法は、`-overlayorder` オプションの 1 つを指定して `afbconfig` コマンドを入力します。

以下に例を示します。

```
# afbconfig -overlayorder first
```

## デフォルトの画像表示形式を変更する (`-deflinear`、`-defoverlay`)

画面のデフォルトの画像表示形式は、8 ビット疑似カラーオーバーレイ画像です。デフォルトの画像表示形式は `afbconfig` コマンドを使用して、リニアアンダーレイ画像またはオーバーレイ画像のいずれかに変更することができます。

- デフォルトの画像表示形式をリニア画像にする場合は、`afbconfig` コマンドを以下のように指定します。

```
# afbconfig -deflinear true
```

- デフォルトの画像表示形式をオーバーレイ画像にする場合は、`afbconfig` コマンドを以下のように指定します。

```
# afbconfig -defoverlay true
```



注意 – リニアオーバーレイ画像は存在しないため、`-deflinear true` と `-defoverlay true` を同時に指定した場合は、結果は不定となります。



注意 – 画像表示形式の順序オプション (`overlayorder` と `linearorder`) は、デフォルト画像表示形式オプション (`defoverlay` と `deflinear`) とは異なります。たとえば、オーバーレイ画像表示形式のグループをリストの始めに移動させても、その画像表示形式が自動的にデフォルトとなるわけではありません。特定のアプリケーションでは、変更した画像表示形式の順序がデフォルトになるものとみなし、デフォルトの画像表示形式で生成したカラーマップと、リストで最初にあった 8 ビット疑似カラー画像を一致させようとして、「BADMATCH」 X エラーを起こす場合があります。

## OpenGL の画像表示形式に変更する (`-expvis`)

日本語 Solaris 2.6 ハードウェアリリース以降では、OpenGL 拡張画像表示形式をサポートします。拡張画像表示形式では、8 ビット疑似カラー、24 ビットトゥルーカラー (リニアおよび非リニア)、24 ビットダイレクトカラー、8 ビット疑似カラーオーバーレイの 5 種類の画像グループは、同じ画像タイプの単一画像インスタンスから、複数画像インスタンスに拡張されます。同じ画像グループの異なるインスタンスは (単一バッファかダブルバッファ機能、または単一画像か立体画像機能、あるいは両方の組み合わせといった)、異なる GLX 機能を表します。画像インスタンスの数は、X サーバーが開始された画像モードが単一画像か立体画像かによって異なります。

### ▼ OpenGL の画像表示形式 (拡張画像表示形式) を有効にする

- 以下のように入力します。

```
# afbconfig -expvis enable
```

## SERVER\_OVERLAY\_VISUALS 属性に変更する (-sov)

SERVER\_OVERLAY\_VISUALS は、画像 ID、透過タイプ、透過値、画面のサーバーオーバーレイ画像 (SOV : server overlay visuals) を含むルートウィンドウの属性の 1 つです。afbconfig コマンドを使用して、この属性を有効としたり、透過サーバーオーバーレイ画像表示形式の参照先の指定を切り替えることができます。

### ▼ SERVER\_OVERLAY\_PROPERTY のアドバタイズと SOV のエクスポートを設定する

- 以下のように入力します。

```
# afbconfig -sov enable
```

## ガンマ補正を設定する (-g、-gfile)

ガンマ補正を設定するには、ガンマ補正值またはガンマ補正テーブルが保存されたファイルを指定します。

-g オプションを使用して、指定されたガンマ値に基づくガンマテーブルエントリを設定します。ガンマ値 2.22 はリニアガンマ補正を表し、Elite3D 製品での固定値に一致します。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

### ▼ 値を使用してガンマ補正を設定する

- 以下のように入力します。

```
# afbconfig -g 2.22
```

-gfile オプションは、0 から 255 までの 3 桁から成る 256 個の整数を含んだファイルから、ガンマテーブルエントリを明示的に設定します。この形式は、改行で区切られた 3 つの整数となります。各行はテーブル中のエントリとして使用される RGB 値を含んでいます。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

## ▼ ファイルを使用してガンマ補正を設定する

- 以下のように入力します。

```
# afbconfig -gfile ファイル名
```

## 拡張オーバーレイを選択する (-extovl)

拡張オーバーレイモードでは、オーバーレイ画像に不透明の 256 色が与えられます。サーバーオーバーレイ画像 (SOV: server overlay visuals) には不透明の 255 色と透明の 1 色が与えられます。拡張オーバーレイモードではハードウェアによってアクセラレートされる透過性が提供され、SOV 画像を使用したウィンドウ表示の性能が向上します。拡張オーバーレイモードでは、非オーバーレイ画像を使用するウィンドウに対し 64 のウィンドウ ID (WID: Window ID) があり、オーバーレイ画像を使用するウィンドウには 3 つの WID があります。

拡張オーバーレイモードを解除した場合、オーバーレイ画像の色数は 256 から `maxwids` オプションで指定された値 (デフォルトは 32) を引いた数となります。SOV が可能な画像は、同じく 256 色から `maxwids` オプションの値を引いた数と、透過性のソフトウェアエミュレーションの数を足したものとなります。

## ▼ 拡張オーバーレイモードを選択する

- 以下のように入力します。

```
# afbconfig -extovl enable
```

表 7-5 に `afbconfig -extovl` オプションのデフォルトの設定を示します。

表 7-5 `afbconfig -extovl` オプションのデフォルトの設定

値	Solaris 2.5.1 のデフォルト	Solaris 2.6 のデフォルト
<code>enable/disable</code>	<code>disable</code>	<code>enable</code>

## オーバーレイ中での WID 数と色数を選択する (-maxwids)

`maxwids` オプションは、拡張オーバーレイモードが無効のときにだけ使用することができます。`maxwids` オプションは、WID として使用するために予約しておく Elite3D X チャネルのピクセル値の最大数を指定します。残りのピクセル値はオーバーレイウィンドウで使用できる色数です。`maxwids` オプションに指定できる値は 1、2、4、8、16、32、64 のうちのどれかです。

### ▼ WID の番号を選択する

- 次のように入力します。

```
# abconfig -maxwids WID の番号
```

表 7-6 に `afbconfig -maxwids` オプションのデフォルトの設定を示します。

表 7-6 `afbconfig -maxwids` オプションのデフォルトの設定

値	Solaris 2.5.1 のデフォルト	Solaris 2.6 のデフォルト
1、2、4、8、16、32、64	32	使用不可

## `afbconfig` 画像フラグの画像表示形式リストへの影響

### デフォルトの画像表示形式とその順序への影響

X サーバーの画像表示形式リストの順序は、必要に応じて `afbconfig` 画像フラグの 1 つ (`linearorder`、`overlayorder`、`expvis`、`sov` など) を使用して変更することができます。この節では、このような画像オプションの画像表示形式リストへの影響の概要を説明します。

`afbconfig` による変更をまったく行わない場合は、X サーバーの画像表示形式リストは、次の画像グループと順序に分類することができます。

- 8ビット疑似カラー
- 8ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8ビットリニアスタティックグレー
- 8ビット疑似カラーオーバーレイ
- 24ビット非リニアトゥルーカラー
- 24ビットダイレクトカラー
- 24ビットリニアトゥルーカラー

デフォルトの画像表示形式は8ビット疑似カラー画像です。Xサーバーの開始時の深さと、最初の8ビットオーバーレイ画像のそれぞれに合わせて、`deflinear` および `defoverlay` オプションを使用して、ポインタを最初の8または24ビットリニア画像を指すように変更することができます。`linearorder` および `overlayorder` フラグを使用して、同じ深さにある他の画像グループの前に8ビット疑似カラーオーバーレイのグループ、8ビットリニアスタティックグレーのグループ、24ビットリニアトゥルーカラーのグループ全体を置くことが可能です。たとえば、以下のように指定します。

```
# afbconfig -overlayorder first
```

この場合、画像表示形式リストは次のように並べ換えられます。

- 8ビット疑似カラーオーバーレイ
- 8ビット疑似カラー
- 8ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8ビットリニアスタティックグレー
- 24ビット非リニアトゥルーカラー
- 24ビットダイレクトカラー
- 24ビットリニアトゥルーカラー

## 選択されたグループ中での画像インスタンス番号に対する影響

`expvis` フラグによって、次の画像グループの画像インスタンス番号を変更することができます。

- 8ビット疑似カラー
- 8ビット疑似カラーオーバーレイ
- 24ビット非リニアトゥルーカラー

- 24 ビットダイレクトカラー
- 24 ビットリニアトゥルーカラー

`expvis` によって変更されるインスタンスの数は、モニターが単一解像度になっているか立体解像度になっているかによって異なります。単一解像度になっている場合に `expvis` が有効になると、画像表示形式の順番は保存されますが、上記の各グループ内には「ダブルバッファ機能」画像インスタンスが追加されます。立体解像度になっている場合は、「ダブルバッファ・立体機能」と「単一バッファ・立体機能」の2種類の画像インスタンスが追加されます。「ダブルバッファ機能」画像インスタンスがある場合は、常に「単一バッファ機能」の前に来ます。また、単一画像インスタンスは、常に立体画像インスタンスよりも前に来ます。

たとえば、立体解像度の場合に、以下のように指定します。

```
# afbconfig -overlayorder first -expvis enable
```

この場合、画像表示形式リストは次のようになります。

- 8 ビット疑似カラーオーバーレイ (単一、立体)
- 8 ビット疑似カラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 24 ビットリニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビット非リニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビットダイレクトカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)

---

注 - ダブルバッファ機能のオーバーレイ画像インスタンスはありません。

---

## 8 ビットオーバーレイグループへの SERVER\_OVERLAY\_VISUALS 属性と透過 SOV 画像表示形式の追加

`sov` オプションを有効にせずに使用可能なオーバーレイ画像形式は、透過性のない画像形式だけです。`sov` オプションの有効化によって、透過的 SOV 画像インスタンスを画像表示形式リストに、`SERVER_OVERLAY_VISUALS` 属性をルートウィンドウ属性に追加することができます。透過 SOV 画像インスタンスは、「8 ビット疑似カラーオーバーレイ」画像グループに属しています。`SERVER_OVERLAY_VISUALS` 属性は、その画像の ID、透過タイプ、透過値、その画面で使用可能なすべてのオーバーレイ画像を含んでいます。

たとえば、立体解像度になっている場合に、以下のコマンドを入力します。

```
# afbconfig -overlayorder first -expvis enable -sov enable
```

この場合、画像表示形式リストは次のようになります。

- 8 ビット疑似カラーオーバーレイ (単一、立体、単一 SOV、立体 SOV)
- 8 ビット疑似カラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 8 ビット混在 (スタティックカラー、非リニアスタティックグレーなど)
- 8 ビットリニアスタティックグレー
- 24 ビットリニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビット非リニアトゥルーカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)
- 24 ビットダイレクトカラー (ダブルバッファ/単一、単一バッファ/単一、ダブルバッファ/立体、単一バッファ/立体)

SERVER\_OVERLAY\_VISUALS 属性には、以下の情報が含まれます。

```
=====
                          SERVER OVERLAY VISUALS (SOV) Info
=====
No. of SOV visuals = 4
SOV #0, ID 0x36, TRANSPARENT_TYPE 0, VALUE 0, LAYER 1
SOV #1, ID 0x37, TRANSPARENT_TYPE 0, VALUE 0, LAYER 1
SOV #2, ID 0x38, TRANSPARENT_TYPE 1, VALUE 255, LAYER 1
SOV #3, ID 0x39, TRANSPARENT_TYPE 1, VALUE 255, LAYER 1
=====
```

---

## ステレオコネクタ

ステレオコネクタを使用して、Elite3D グラフィックアクセラレータに立体視ゴーグルを接続することができます。Elite3D ボード用のステレオコネクタは、7 ピンの DIN コネクタです (図 7-1 を参照)。

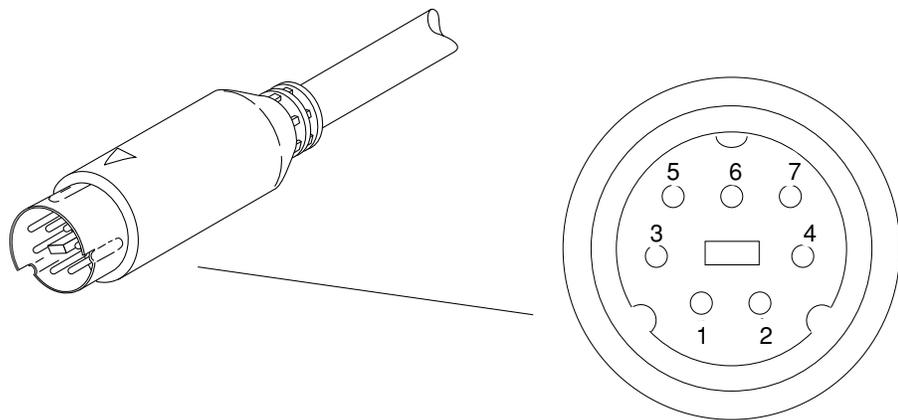


図 7-1 Elite3D のステレオコネクタ

表 7-7 にステレオコネクタの信号を示します。

表 7-7 Elite3D のステレオコネクタの信号

ピン	説明
1	接地
2	接続なし
3	+12V
4	ステレオ信号
5	接続なし
6	接続なし
7	接続なし

## ステレオ信号

ステレオ信号は、図 7-2 に示すように、左右の立体シャッタの切り換え周期が 50 パーセントデューティーの、TTL レベルの信号です。



図 7-2 Elite3D のステレオ信号

注 – Elite3D グラフィックアクセラレータのステレオケーブルと立体視ゴーグルについての詳細は、購入先にお問い合わせください。

## 第8章

# Expert3D グラフィックスアクセラレータ

---

この章では、Expert3D グラフィックスアクセラレータの画面解像度を、モニターの種類に合わせて変更する方法について説明します。後半では、Expert3D の内部および外部コネクタとその用途を説明します。

---

注 – Expert3D-Lite ソフトウェアパッケージおよび設定ユーティリティーは Expert3D のものと同じです。

---

Expert3D グラフィックスアクセラレータは、高い解像度とパフォーマンスを持ち、ハードウェアテクスチャマッピングを可能にするグラフィックスアクセラレータです。Expert 3D システムでは、ダブルバッファと Z バッファの場合最大 1920 x 1200、立体表示の場合最大 1280 x 1024 の画面解像度をサポートします。

---

注 – ifb は、Expert3D グラフィックスアクセラレータのUNIX デバイス名です。

---

---

## 設定ユーティリティー

Expert3D グラフィックスアクセラレータには、fbconfig と SUNWifb\_config の2つの設定ユーティリティーがあります。

## fbconfig ユーティリティー

fbconfig ユーティリティーは一般的なコマンド行インタフェースのツールで、フレームバッファの属性を表示および設定できます。fbconfig は、Expert3D のほか、Creator3D と Elite3D の設定にも使えます。

fbconfig ユーティリティーでは次のことができます。

- インストール済みのフレームバッファの一覧とデバイス毎に使用できる設定ユーティリティーの一覧の表示
- [SUNwifb\\_config](#) ユーティリティーで使用できるすべての機能

詳細は [fbconfig](#) のマニュアルページを参照してください。

## SUNWifb\_config ユーティリティー

SUNWifb\_config ユーティリティーは、Expert3D 固有のユーティリティーです。他のフレームバッファの設定には使えません。[SUNwifb\\_config](#) ユーティリティーによって、以下の指定が可能です。

- 画面解像度の変更
- 画像表示形式の種類 (リニアまた非リニア)
- 8 ビット擬似カラー画像 (オーバーレイ画像) を使用するかどうか
- 表示画像リスト中で、リニア画像を非リニア画像より前に表示させるかどうか
- デフォルトの表示画像の設定方法
- OpenGL 画像のサポート方法
- サーバーオーバーレイ画像 (SOV) が有効かどうか
- スレーブモードを有効にして、フレームバッファからの映像が接続されたディスプレイの映像と同期するようにする
- 立体表示モードを有効にする
- フレームバッファのメモリーの、ウィンドウの高速化蓄積バッファへの配置
- ガンマの値、またはガンマ補正テーブルを含むファイルの指定による構成可能ガンマ補正

詳細は、[SUNwifb\\_config](#) のマニュアルページを参照してください。

---

## デフォルトの画面解像度

通常 Expert3D はインストール時に、接続されたモニターに対応する適切な解像度とリフレッシュレートを自動的に設定します。ただし、サン以外のモニターが接続されている場合、解像度が適切に設定されないことがあります。

表 8-1 に、モニター ID センスコードに対応するデフォルトの画面解像度を示します。

表 8-1 Expert3D グラフィックアクセラレータのモニターセンスコード

コード	画面解像度
7	1152 x 900、66 Hz
6	1152 x 900、76 Hz
5	1024 x 768、60 Hz

サン以外のモニターなど、Expert3D のシステムがモニターの種類を判断できない場合は、1152 x 900、66 Hz の解像度をデフォルトとします。

---

## サポートする画面解像度

表 8-2 に、Expert3D がサポートする画面解像度を示します。画面解像度を下記の設定に変更するには、fbconfig または SUNWifb\_config ユーティリティを使ってください。

表 8-2 Expert3D がサポートする画面解像度

画面解像度	垂直リフレッシュレート	説明	ビデオモード形式	シンボリック名
1920 x 1200	75 Hz	ノンインタレース	1920x1200x75	
1920 x 1200	70 Hz	ノンインタレース	1920x1200x70	
1920 x 1080	76 Hz	ノンインタレース	1920x1080x76	
1920 x 1080	72 Hz	ノンインタレース	1920x1080x72	
1792 x 1344	75 Hz	ノンインタレース	1792x1344x75	

表 8-2 Expert3D がサポートする画面解像度 (続き)

画面解像度	垂直リフレッシュ レート	説明	ビデオモード形式	シンボリック名
1600 x 1280	76 Hz	ノンインタレース	1600x1280x76	
1600 x 1000	76 Hz	ノンインタレース	1600x1000x76	
1600 x 1000	66 Hz	ノンインタレース	1600x1000x66	
1440 x 900	76 Hz	ノンインタレース	1440x900x76	
1280 x 1024	112 Hz	立体表示、ノンインタレース、56 Hz フィールドレート	1280x1024x112s	
1280 x 1024	85 Hz	ノンインタレース	1280x1024x85	
1280 x 800	76 Hz	ノンインタレース	1280x800x76	1280
1280 x 1024	67 Hz	ノンインタレース	1280x1024x67	
1280 x 1024	60 Hz	ノンインタレース	1280x1024x60	
1280 x 800	76 Hz	ノンインタレース - Creator	1280x800x76	
1152 x 900	76 Hz	ノンインタレース	1152x900x76	1152
1152 x 900	66 Hz	ノンインタレース	1152x900x66	
1024 x 768	77 Hz	ノンインタレース	1024x768x77	
1024 x 768	75 Hz	ノンインタレース	1024x768x75	
1024 x 768	70 Hz	ノンインタレース	1152x900x70	
1024 x 768	60 Hz	SVGA	1024x768x60	svga
960 x 680	112 Hz	立体表示、ノンインタレース、56 Hz フィールドレート	960x680x112	stereo
960 x 680	108 Hz	立体表示、ノンインタレース、54 Hz フィールドレート	960x680x108	
768 x 575	50 Hz	インタレース - PAL	768x575x50i	pal
640 x 480	60 Hz	インタレース - NTSC	640x480x60i	ntsc
640 x 480	60 Hz	ノンインタレース	640x480x60	

---

## 画面解像度を変更する (-res)

### ▼ Expert3D がサポートする解像度と接続されているモニターを確認する

- `SUNwifb_config` コマンドを以下のように入力します。

```
/usr/lib/fbconfig/SUNwifb_config -res \?
```

特定の解像度をモニターがサポートしているかどうかを確認するために、画面解像度を一時的に変更することができます。

---

注意 – ウィンドウシステムの実行中に、画面解像度を変更しないでください。画面表示に異常が発生する場合があります。

---

### ▼ 画面解像度を一時的に変更する

- `SUNwifb_config` コマンドを以下のように入力します。

```
/usr/lib/fbconfig/SUNwifb_config -res video-mode try
```

ビデオモードのオプションについては表 8-2 の「ビデオモード形式」と「シンボリック名」の欄を参照してください。ビデオモードを確認し、そのビデオモードを使用する場合は、5 秒以内に `y` と入力してください。

### ▼ 画面解像度を立体表示に変更する

- 以下のように入力します。

```
/usr/lib/fbconfig/SUNwifb_config -res stereo
```

上記の例では、次回に `Xsun` を実行したときの画面解像度を、960 x 680、112 Hz の立体表示に変更します。

---

## 画像表示形式リストを変更する

`SUNwifb_config` コマンドを使用して、X サーバーの画像表示形式リストの順序を変更することができます。表 8-3 に示す `SUNwifb_config` オプションを使用して、特定のデバイスに合わせて参照された画像の表示形式リストを設定してください。

表 8-3 `SUNwifb_config` の画像フラグのデフォルトの設定

名称	値	デフォルト
<code>linearorder</code>	<code>first/last</code>	<code>last</code>
<code>deflinear</code>	<code>true/false</code>	<code>false</code>
<code>overlayorder</code>	<code>first/last</code>	<code>last</code>
<code>defoverlay</code>	<code>true/false</code>	<code>false</code>
<code>expvis</code>	<code>enable/disable</code>	<code>enable</code>
<code>sov</code>	<code>enable/disable</code>	<code>enable</code>

### 画像表示形式リストの順序を変更する (`-linearorder`、`-overlayorder`)

デフォルトでは、非リニア画像は、画像表示形式リスト上でリニア画像より前に表示されます。`SUNwifb_config` コマンドを使用して、画像表示形式リストの順序を変更することができます。

3D アプリケーションの多くはリニア画像を使用しますが、特定の 3D アプリケーションでは、`XSolarisGetVisualGamma` (3) を使用するリニア画像を検索しないで、リストに最初に表示される 24 ビットトゥルーカラーを使用します。このようなアプリケーションを、正しい画像表示形式で動作させるには、リストの並べ換えが必要です。`-linearorder` オプションを使用して、リニア 24 ビットトゥルーカラーが最初に表示されるように指定してください。

画像表示形式リストの並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する場合は、`-linearorder` オプションの 1 つを指定して `SUNWifb_config` コマンドを入力します。

```
/usr/lib/fbconfig/SUNWifb_config -linearorder first
```

デフォルトでは、8 ビット疑似カラーアンダーレイ画像は、画面表示形式リスト上で 8 ビット疑似カラーオーバーレイ画像より前に表示されます。`SUNWifb_config` コマンドを使用して、画像表示形式リストの順序を変更することができます。

8 ビット疑似カラーオーバーレイ画像を使うアプリケーションによっては、画面表示形式リスト上で最初に検索された 8 ビット疑似カラー画像を使うものがあります。これらのアプリケーションが正しい画像で動作するようにするには、`-overlayorder` オプションを使い、8 ビット疑似カラーオーバーレイ画像が 1 番目の 8 ビット疑似カラー画像として検知されるようにします。

画像表示形式リストの並べ換えは、ウィンドウシステムの再起動時に有効となります。

- リストの順序を変更する場合は、`overlayorder` オプションの 1 つを指定して `SUNWifb_config` コマンドを入力します。

```
/usr/lib/fbconfig/SUNWifb_config -overlayorder first
```

## デフォルトの画像表示形式を変更する (`-deflinear`、`-defoverlay`)

画面のデフォルトの画像表示形式は、8 ビット疑似カラーオーバーレイ画像です。デフォルトの画像表示形式は `SUNWifb_config` コマンドを使用して、リニアアンダーレイ画像またはオーバーレイ画像のいずれかに変更することができます。

- デフォルトの画像表示形式をリニア画像にする場合は、`SUNWifb_config` コマンドを以下のように指定します。

```
/usr/lib/fbconfig/SUNWifb_config -deflinear true
```

- デフォルトの画像表示形式をオーバーレイ画像にする場合は、`SUNwifb_config` コマンドを以下のように指定します。

```
/usr/lib/fbconfig/SUNwifb_config -defoverlay true
```



注意 – リニアオーバーレイ画像は存在しないため、`-deflinear true` と `-defoverlay true` を同時に指定した場合は、結果は不定となります。



注意 – 画像表示形式の順序オプション (`overlayorder` と `linearorder`) は、デフォルト画像表示形式オプション (`defoverlay` と `deflinear`) とは異なります。たとえば、オーバーレイ画像表示形式のグループをリストの始めに移動させても、その画像表示形式が自動的にデフォルトとなるわけではありません。特定のアプリケーションでは、変更した画像表示形式の順序がデフォルトになるものとみなし、デフォルトの画像表示形式で生成したカラーマップと、リストで最初にあった 8 ビット疑似カラー画像を一致させようとして、「BADMATCH」 X エラーを起こす場合があります。

## OpenGL の画像表示形式に変更する (`-expvis`)

Solaris では、OpenGL 拡張画像表示形式をサポートします。拡張画像表示形式では、8 ビット疑似カラー、24 ビットトゥルーカラー (リニアおよび非リニア)、24 ビットダイレクトカラー、8 ビット疑似カラーオーバーレイの 5 種類の画像グループは、同じ画像タイプの単一画像インスタンスから、複数画像インスタンスに拡張されます。同じ画像グループの異なるインスタンスは (単一バッファかダブルバッファ機能、または単一画像か立体画像機能、あるいは両方の組み合わせといった)、異なる GLX 機能を表します。画像インスタンスの数は、X サーバーが開始された画像モードが単一画像か立体画像かによって異なります。

### ▼ OpenGL の画像表示形式 (拡張画像表示形式) を有効にする

- 以下のように入力します。

```
/usr/lib/fbconfig/SUNwifb_config -expvis enable
```

## SERVER\_OVERLAY\_VISUALS 属性に変更する (-sov)

SERVER\_OVERLAY\_VISUALS は、画像 ID、透過タイプ、透過値、画面のサーバーオーバーレイ画像 (SOV : server overlay visuals) を含むルートウィンドウの属性の 1 つです。 `SUNwifb_config` コマンドを使用して、この属性を有効としたり、透過サーバーオーバーレイ画像表示形式の参照先の指定を切り替えることができます。

### ▼ SERVER\_OVERLAY\_PROPERTY のアドバタイズと SOV のエクスポートを設定する

- 以下のように入力します。

```
/usr/lib/fbconfig/SUNwifb_config -sov enable
```

---

## ガンマ補正を設定する (-g、-gfile)

ガンマ補正を設定するには、ガンマ補正值またはガンマ補正テーブルが保存されたファイルを指定します。

`-g` オプションを使用して、指定されたガンマ値に基づくガンマテーブルエントリを設定します。ガンマ値 2.22 はリニアガンマ補正を表し、Expert3D 製品での固定値に一致します。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

### ▼ 値を使用してガンマ補正を設定する

- 以下のように入力します。

```
SUNwifb_config -g 2.22
```

`-gfile` オプションは、0 から 255 までの 3 桁から成る 256 個の整数を含んだファイルから、ガンマテーブルエントリを明示的に設定します。この形式は、改行で区切られた 3 つの整数となります。各行はテーブル中のエントリとして使用される RGB 値を含んでいます。この値は画面ごとの値なので、リニア画像と補正の対象となるガンマ値はすべてこの値を使用します。

## ▼ ファイルを使用してガンマ補正を設定する

- 以下のように入力します。

```
SUNwifb_config -gfile filename
```

---

## 拡張オーバーレイを選択する (-extovl)

拡張オーバーレイモードを有効にすると、Expert3D システムを拡張オーバーレイ機能に切り替えます。拡張オーバーレイモードでは、256 色オーバーレイとハードウェアによって高速化されたサーバーオーバーレイ画像が使えるようになります。アンダーレイの Window ID の上限は、以前の 32 から 64 に上がり、新しく 3 つのオーバーレイ WID が使えます。

## ▼ 拡張オーバーレイモードを選択する

- 以下のように入力します。

```
SUNwifb_config -extovl enable
```

---

## 複数モニターの設定

複数のフレームバッファを使うには、`/usr/dt/config` にある `Xservers` ファイルを修正する必要があります。Expert3D デバイスは、ここで `ifbx` こととして定義されます (たとえば、Expert3D デバイスが 2 つある場合、`ifb0` と `ifb1` と定義されます)。

1. スーパーユーザーとして `/usr/dt/config/Xservers` ファイルを開きます。

```
# cd /usr/dt/config  
# vi Xservers
```

2. 使用するフレームバッファのデバイスロケーションをファイルに追加します。

以下の例では、Creator カード 1 個と Expert3D カード 2 個を使用する場合の Xservers の設定を示します。

```
:0 Local local_uid@console root /usr/openwin/bin/Xsun :0 -dev /dev/fbs/ffb0  
-dev /dev/fbs/ifb1 -dev /dev/fbs/ifb0 -nobanner
```

1 台のシステムで複数の Expert3D カードを設定する方法については、105 ページの「マルチビューポート」を参照してください。

---

## Expert3D カードをデフォルトのコンソールとして定義する

複数モニターのシステムでは、いずれかのモニターをデフォルトのコンソールとして指定できます。以下では、Expert3D カードをデフォルトのコンソールディスプレイとして設定する方法を示します。

1. OpenBoot のプロンプトで、以下のように入力します。

```
ok show_displays
```

以下の例のように使用できるフレームバッファが表示されます (Expert3D の場合)。

```
a) /PCI@@11, 2000/SUNW,Expert3D@1
b) /PCI@@15, 2000/SUNW,afb@1
q) NO SELECTION
```

以下の例のように使用できるフレームバッファが表示されます (Expert3D-Lite の場合)。

```
a) /pci@1f,0/SUNW,m64@13
b) /pci@1f,0/pci@5/SUNW,Expert3D@1
q) NO SELECTION
Enter Selection, q to quit: b
/pci@1f,0/pci@5/SUNW,Expert3D@1 has been selected.
Type ^Y ( Control-Y ) to insert it in the command line.
e.g. ok nvalias mydev ^Y
      for creating devalias mydev for
/pci@1f,0/pci@5/SUNW,Expert3D@1
ok nvalias myscreen /pci@1f,0/pci@5/SUNW,Expert3D@1
ok setenv output-device myscreen
output-device =          myscreen
```

2. コンソールとして使いたいデバイスのオプションを選択します。

たとえば、上の例で Expert3D デバイスをコンソールとして指定する場合は、**a** を選択します。

3. 選択したデバイスをコンソールデバイスとして設定します。

OpenBoot プロンプトで、以下のように入力します。

```
ok setenv output-device <Control-Y> <ENTER>
ok reset
```

Expert3D システム用のコマンドをエイリアスとして登録しておく、今後の OpenBoot コマンドを簡略化できます。show-displays コマンドで Expert3D デバイスをメニューから選択した後に、以下のように入力します。

```
ok nvalias screen1 <Control-Y>
ok nvstore
```

次に以下のように入力し、エイリアスが正しく設定されたことを確認します。

```
ok devaliases
```

以上の操作が終わると、Expert3D デバイスを screen1 として参照できるようになります。

```
ok setenv output-device screen1
```

---

## Expert3D の外部コネクタ

Expert3D ボードには、内部入出力コネクタと外部入出力コネクタの両方があります。内部コネクタはマルチビューイングに、外部コネクタは外部の画像との同期化とステレオに使用します。図 8-1 は、外部入出力コネクタを示します。

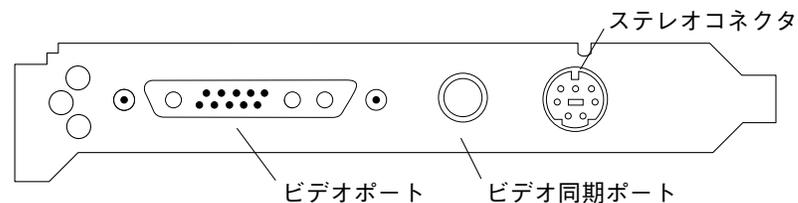


図 8-1 Expert3D の外部コネクタ

Expert3D-Lite グラフィックスカードには外部ビデオ同期ポートがありません。Expert3D-Lite のキットには DB15 から 13W3 への変換用モニターアダプターケーブルが同梱されています。図 8-2 は、外部入出力コネクタを示します。

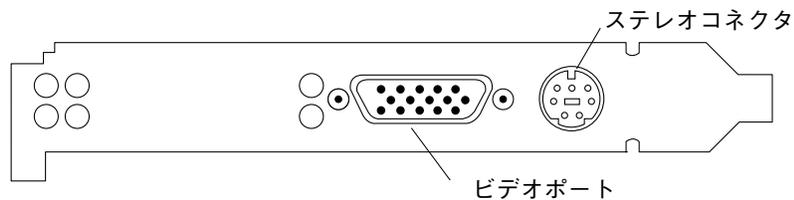


図 8-2 Expert3D-Lite の外部入出力コネクタ

## ステレオコネクタ

ステレオコネクタを使用して、Expert3D グラフィックスアクセラレータに立体視ゴーグルを接続できます。ステレオコネクタは、図 8-3 に示すように7 ピンの DIN コネクタです。

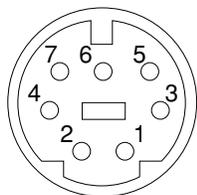


図 8-3 Expert3D のステレオコネクタ

表 8-4 にステレオケーブルの信号を示します。

表 8-4 Expert3D ステレオコネクタの信号

ピン	説明
1	接地
2	5.0V (ヒューズ付き、300mA まで供給、1.0A 上限)
3	+12V (ヒューズ付き、300mA まで供給、1.0A 上限)
4	STEREO 信号
5	接続なし
6	接続なし
7	接続なし

ステレオ信号は、図 8-4に示すように、左右の立体シャッタの切り替え周期が 50 パーセントデューティーの、TTL レベルの信号です。



図 8-4 Expert3D のステレオ信号

---

注 – Expert3D グラフィックアクセラレータのステレオケーブルと立体視ゴーグルについての詳細は、導入先にお問い合わせください。

---

## ビデオ同期ポート (Expert3D のみ)

ビデオ同期ポートは、ディスプレイシステムが垂直リフレッシュレートをロックしようとする際の定期信号を Expert3D ボードに入力するインタフェースです。このポートは 75 オーム BNC のオスのコネクタを使っています。このポートに接続するには、メスの BNC コネクタが付いた 75 オームの同軸ケーブルが必要です。

ビデオ同期ポートは次のフォーマットをサポートします。

- NTSC 複合ビデオ
- PAL 複合ビデオ
- 100 nS (TBV) 以上のパルスをもつ TTL、LVTTTL、または CMOS レベルの互換信号 (50 Hz から 180 Hz)

## マルチビューポート

図 8-5 の J1 および J2 に示される、Expert3D カードの内部のマルチビュー入力ポートおよびマルチビュー出力ポートは、Expert3D カードから他のカードへケーブル接続をする際に使います。これらのポートはシステムのシャーシ内からのみ接続できます。

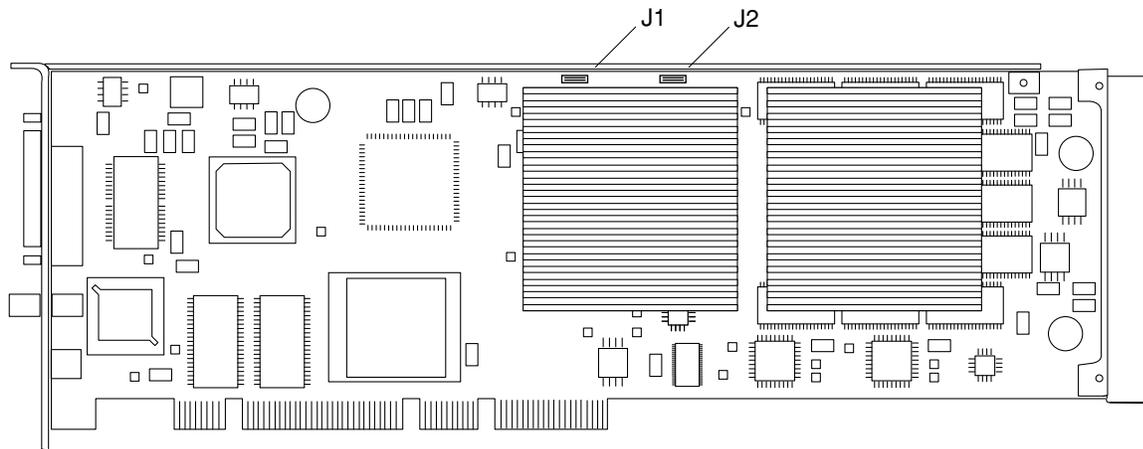


図 8-5 Expert3Dのマルチビュー入力コネクタとマルチビュー出力コネクタ

## 複数の Expert3D カードの設定

マルチビューを使用する場合、Expert3D カードは2つまでしかフレームロックできません。2つの Expert3D カードがマルチビューに使われている場合、片方がマスターとして割り当てられ、もう一方がスレーブとして割り当てられます。

- システムを起動するとき、すべての Expert3D カードは Solaris によってマスターモードに設定されます。Expert3D カードをマルチビューで使用するためには、[SUNwifb\\_config](#) ユーティリティを使ってカードの1つをスレーブモードにします。
- システム上で設定できるマスター Expert3D カードの数は1つのみです。
- すべての Expert3D モニターが、Expert3D でサポートする解像度に設定されているかを確認します。
- デバイスのロケーションおよび、ロケーションに基づいてつけられるデバイスの番号の情報については、[boot](#) このマニュアルページを参照してください。Expert3D のマスターデバイスとして、どのデバイス ([ifb0](#) または [ifb1](#)) を使ってもかまいません。

フレームロックを使って2つの Expert3D カードをマルチビュー用に設定するには、以下のようにします。

1. システムのマスターとして割り当てる Expert3D カードを選択します。

システムの起動またはコンソールヘッドとして使っている Expert3D カードをマスターとして割り当てることも 1 つの方法です。

2. システムにインストールされているすべての Expert3D カードが、マスターモードの Expert3D カードと同じ画面解像度に設定されていることを確認します。

マスターモードのカードと異なる画面解像度に設定されている Expert3D カードがある場合は、解像度がマスターと同じになるよう変更します。

`SUNwifb_config` ユーティリティを使って、Expert3D カードの解像度を調べられます。たとえば、以下のように入力します。

```
% /usr/lib/fbconfig/SUNwifb_config -dev /dev/fbs/ifb0 -prconf
```

この操作をすべての Expert3D カードに対して実行します。

`SUNwifb_config` ユーティリティに Expert3D カードの現在の解像度や、カードがマスターかスレーブかなどの情報が表示されます。

Expert3D カードの解像度を変更する場合は、`SUNwifb_config` ユーティリティを使って以下の例のように入力します。

```
% /usr/lib/fbconfig/SUNwifb_config -dev /dev/fbs/ifb1 -res \  
1280x1024x76 now nocheck
```

3. フレームロックマルチビューケーブルを各 Expert3D カードに接続します。

図 8-5を参照してください。

- a. マルチビューフレームロックケーブルのアセンブリの上部を、Expert3D カードのマルチビュー出力コネクタ J2 に接続します。
- b. マルチビューフレームロックケーブルの反対側をスレーブの Expert3D カードのマルチビュー入力コネクタ J1 に接続します。

4. マスターの Expert3D カードのディスプレイウィンドウで、他の Expert3D カードをスレーブモードに設定します。

`SUNWifb_config` ユーティリティーで以下のように入力します。

```
% /usr/lib/fbconfig/SUNWifb_config -dev /dev/fbs/ifb1 -slave on
```

複数のカードの設定を変更する場合、カード毎に別々に設定を行ってください。

以上の設定が終わると、システムでフレームロックとバッファースワップ同期用のソフトウェアアプリケーションを使えるようになります。

Expert3D カードをスレーブモードからマスターモードに戻すには、以下のいずれかの操作を実行します。

- システムの電源を再投入します。
- `SUNWifb_config` を使い、たとえば以下のように入力します。

```
% /usr/lib/fbconfig/SUNWifb_config -dev /dev/fbs/ifb1 -slave off
```

複数のカードの設定を変更する場合、カード毎に別々に設定を行ってください。

## 第9章

### 複数のモニターの接続

---

この章では、1 台の SPARCstation システムで複数のモニターを使用する方法について説明します。

増設 SBus スロット (システムによっては PCI スロット) が利用できる場合は、ほとんどの SPARCstation と UltraSPARC システムで、複数のモニター構成がサポートされます。

この手順を行うには、UNIX の知識と、`vi` や `emacs` のような基本的なエディタの使用経験が必要です。

---

### 複数のモニターを設定する

#### Sun Ultra システム

複数のフレームバッファを使うには、`/usr/dt/config` にある `Xservers` ファイルを修正する必要があります。

1. スーパーユーザーとして `/usr/dt/config/Xservers` ファイルを開きます。

```
# cd /usr/dt/config
# vi Xservers
```

2. 使用するフレームバッファのデバイスロケーションをファイルに追加します。

以下の例では、Creator カード 1 個と Expert3D カード 2 個を使用する場合の Xservers の設定を示します。

```
:0 Local local_uid@console root /usr/openwin/bin/Xsun :0 -dev /dev/fbs/ffb0
-dev /dev/fbs/ifb1 -dev /dev/fbs/ifb0 -nobanner
```

## Sun SPARCstation Sbus システム

システムは起動時に、SBus デバイスを検索する順序を決定する `sbus-probe-list` を探します。SPARCstation 10 および SPARCstation 20 では、SBus アドレス `f` は、CPU のために予約され、`sbus-probe-list` の最初のアドレスである必要があります。

システム情報および `sbus-probe-list` を参照するには、以下のように入力します。

```
% eeprom
.
.
.
sbus-probe-list=0123
.
.
```

アドレス指定の数字は、0、1、2、3 です。SPARCstation システムでは、0 は CPU のために予約され、SPARCstation 10 および SPARCstation 20 システム以外では、`sbus-probe-list` の最初のアドレスである必要があります。

SPARCstation 10 および SPARCstation 20 システムの場合には、以下のようにシステム情報と `sbus-probe-list` (`f` で始まる) が表示されます。

```
% eeprom
.
.
.
sbus-probe-list=f0123
.
.
```

---

## デバイスファイル名

複数のモニターで OpenWindows ソフトウェアを使用している場合は、フレームバッファデバイスが UNIX デバイスファイル名に割り当てられる方式を理解している必要があります。OpenWindows ソフトウェアで複数のフレームバッファを使用する場合は、使用開始時に、フレームバッファの UNIX デバイスファイル名をコマンド行で指定してください。

UNIX の起動メッセージは、フレームバッファを `/dev/fb` の形式 (fb はフレームバッファの種類) で識別します。通常 `/dev/fb` は、フレームバッファの種類によって `/dev/fbs/cgsix0`、`/dev/fbs/bwtwo0`、`/dev/fbs/leo0` などの別のデバイスファイル名を持っています。2 番目のフレームバッファが追加されると、システムは各フレームバッファの SBus スロット番号と `sbus-probe-list` の EEPROM 変数に基づいて、`/dev/fb` を決定します。`/dev/fb` は、`sbus-probe-list` に定義されている最初の SBus スロットのフレームバッファです。

GX フレームバッファをすでに備えているシステムに TurboGXplus カードを追加した場合は、`sbus-probe-list` によって `/dev/fbs/cgsix0` と `/dev/fbs/cgsix1` が判別されます。

たとえば、SPARCstation 10 システムの `sbus-probe-list` に既定値 `f0123` があり、SBus スロット 2 と 3 に TurboGXplus カードが実装されているとします。スロット 2 の TurboGXplus カードは、`/dev/fb` および `/dev/fbs/cgsix0` に、スロット 3 の TurboGXplus カードは、`/dev/fbs/cgsix1` になります。

この章で示したコマンド行の例は、フレームバッファの種類を示すデバイスファイル名を使用しています。システムに適したデバイスファイル名に置き換えて使用してください。

---

## 利用可能なフレームバッファを確認する

システムに実装されているフレームバッファデバイスの名前がわからない場合には、以下のコマンドを実行して確認してください。

```
% /etc/dmesg | more
```

この結果、使用可能なフレームバッファの種類とそれが実装されているスロットなどを示すシステム構成が表示されます。メッセージのリストは、非常に長くなる場合があります。cg または leo (カラーフレームバッファ用) や bw (モノクロフレームバッファ用) で始まる行を探してください。

```
.
cgsix0 at SBus0: SBus slot 1 0x0 SBus level 5 sparc ipl 7
cgsix0 is /sbus@1,f8000000/chsix@1,0
cgsix0: screen 1152x900, single buffered, 1M mappable, rev1
.
```

---

注 - `dmesg` コマンドを実行すると、非常に多くのメッセージが表示されるため、必要なシステム構成のメッセージを見落としてしまう場合があります。そのような場合は、(必要に応じて再起動をした後に) 上記のコマンドをもう一度実行してください。

---

---

## コンソールから OpenWindows を起動する

コンソールから OpenWindows を起動するように設定した `.login` ファイルの例を示します。

```
#
# if possible, start the windows system.  Give user a chance to
bail out
#
if ( 'tty' == "/dev/console" && $TERM == "sun" ) then
  if ( ${?OPENWINHOME} == 0 ) then
    setenv OPENWINHOME /usr/openwin
  endif
  echo ""
  echo -n "Starting OpenWindows in 5 seconds (type Control-C to
interrupt)"
  sleep 5
  echo ""
  $OPENWINHOME/bin/openwin
  clear# get rid of annoying cursor rectangle
  logout# logout after leaving windows system

endif
```

---

## 複数のモニターで OpenWindows を実行する

### ▼ OpenWindows バージョン 3 上で複数のモニターを使用する

1. 以下のコマンドを実行し、OpenWindows バージョン 3 の環境を設定します。

(`/usr/local` の部分は、OpenWindows バージョン 3 ソフトウェアが実際に置かれているパス名に置き換えてください。)

```
% setenv OPENWINHOME /usr/local/openwin
```

2. 以下のコマンドを実行し、使用するフレームバッファのデバイスファイルが存在することを確認します。

```
# ls -l /dev/fbs/cgsix1
```

デバイスファイルが存在する場合は、以下のようなメッセージが表示されます。このメッセージが表示された場合は、手順 6 に進んでください。"not found" というメッセージが表示された場合は、手順 3 に進んでください。

```
crw-rw-rw- 1 root      67,   0 Jan 10 1991 /dev/fbs/cgsix1
```

3. スーパーユーザー (root) になり、以下に示すように、システムを停止してから再起動します。

```
# boot -r
```

これによりシステムが接続されたハードウェアデバイスすべてを確認し、2 番目のフレームバッファのデバイスファイルが作成されます。

4. 以下のコマンドを実行し、新しく作成したファイルを確認します。

```
# ls -l /dev/fbs/cgsix1
```

デバイスファイルが正常に作成された場合は、以下のようなメッセージが表示されます。

```
crw-rw-rw- 1 root      67,   0 Jan 10 1991 /dev/fbs/cgsix1
```

5. `exit` コマンドを実行し、一般ユーザーに戻ります。
6. 以下のコマンドを実行し、使用する画面を指定します。

```
% $OPENWINHOME/bin/openwin -dev /dev/fb -dev /dev/fbs/cgsix1
```

---

注 - デバイスを指定する順序は重要です。最初のデバイスは、左の画面に相当します。2 番目のデバイスは、右の画面に相当します。デバイス名 (たとえば、`/dev/fbs/cgsix1`) は、別のものでも構いません。システムに適したデバイスファイル名を使用してください。

---

## ポーリング順序を変更する

この節では、SBus のポーリング順序とそれを変更する方法を説明します。

### SBus アドレス

SPARCstation IPX や LX のように、2 スロットの SPARCstation システムには 0、1、2、3 の 4 つの SBus アドレスがあります。SBus アドレス 0 は、メイン論理ボードに割り当てられ、システム用に予約されています。SBus スロット (SBus アドレス) の 1 と 2 は、ユーザーが組み込む SBus カード用であり、実際に使用可能なスロットです。SBus アドレス 3 には、メイン論理ボードのフレームバッファが割り当てられています。

SPARCstation 2 システムには、システムボード上にフレームバッファがありません。このシステムのスロット 1、2、3 は、SBus カードを組み込むために使用されます。

SPARCstation 10 と SPARCstation 20 には、システムボード上にフレームバッファがありません。SBus アドレス `f` は、CPU 用に予約されていて、`sbus-probe-list` 中の最初のアドレスである必要があります。SPARCstation 10 および SPARCstation 20 のシステムでは、SBus アドレス線の 0、1、2、3 は、ユーザーが組み込む SBus カード用として使用されます。

### ポーリングの順序

ポーリングの順序は、OpenBoot™ PROM の `sbus-probe-list` パラメタによって決定します。SPARCstation 10 および SPARCstation 20 の場合は、`f` で始まる必要があります。このパラメタは、スロット 0 から 3 の順序でポーリングするように設定されています。スロット 1、2、3 の順序は変更することができますが、必ずスロット 0 から始まる必要があります。

たとえば、SPARCclassic™、SPARCstation IPX、SPARCstation LX の構成では、SBus スロットにフレームバッファカードを実装すると、システムはシステムボード上のフレームバッファよりも先に SBus スロットのフレームバッファを探します。SBus スロットにフレームバッファを認識すると、システムはそのスロットでビデオ接続を確立し、それ以上は探しません。システムがシステムボード上のフレームバッファを最初に探すようにする場合は、ポーリングの順序を 0、3、1、2 に変更します。

---

注 – SPARCstation IPX、SPARCstation LX システムの `sbus-probe-list` を変更し、コンソールデバイスとして 3 を選択する場合には、モニターをシステムボード上のフレームバッファに接続してください。システムは、ポーリングの順序に関わらず、他のモニターを認識しません。

---

## `sbus-probe-list` の変更

以下の手順で、`sbus-probe-list` を変更します。



---

注意 – 以下に示した手順は、SunOS™ ソフトウェアに精通したユーザーを対象としています。モニターが 1 台だけの場合や、`sbus-probe-list` を変更する必要がない場合には、この手順を無視してください。`eeprom` コマンドの入力後に表示されるシステム情報を変更すると、システムの構成が変更されます。

---

### ▼ `sbus-probe-list` を変更する

- スーパーユーザーになり、以下を入力してください。

```
# eeprom sbus-probe-list=0xyz (システムが SPARCstation 10 の場合は  
fwxyz)
```

`xyz` または `wxyz` は、SBus スロットを調べる順序です。

SPARCclassic、SPARCstation IPX、SPARCstation LX の構成では、0312 は SBus スロット 3 (システムボード上のフレームバッファ) を最初に調べ、その後に SBus スロット 1、2 の順に調べます。以下に入力例を示します。

```
# eeprom sbus-probe-list=0312
```

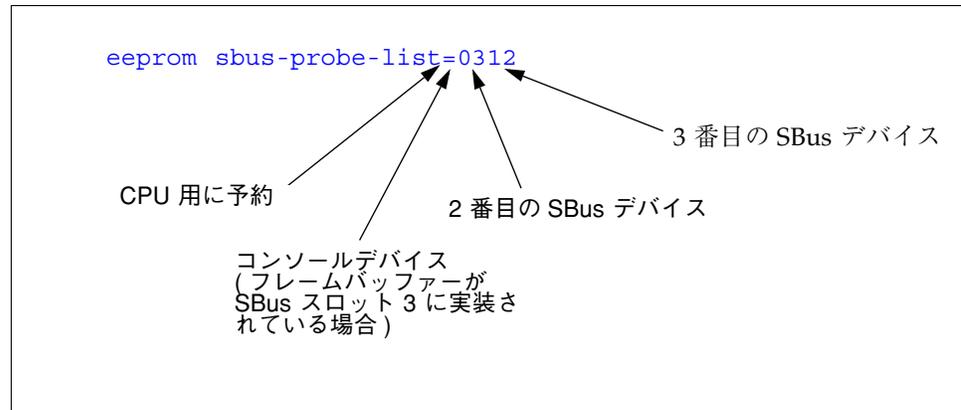


図 9-1 `sbus-probe-list` の書式

`eeprom sbus-probe-list` 中の 0 を除いた最左端の文字は、最初に調べられるデバイスを示します。これは物理的な位置に関わらず、コンソールデバイスとなります。

---

注 – SPARCstation 10 または SPARCstation 20 システムを使用している場合は、`sbus-probe-list` の最左端の文字は CPU のために予約されている `f` (0 ではなく) になります。

---

1. モニターが、コンソールとして認識されているフレームバッファに接続されていることを確認してください。
2. システムを再起動し、変更を有効にします。



# 索引

---

## 数字

- 1 台のモニターの設定, UNIX スクリプトによる, 7
- 1 台のモニターの設定, PROM による, 7
- 24 ビットトゥルーカラー画像, 29, 35
- 24 ビットトゥルーカラーリニア画像, 32
- 8 ビット疑似カラー, 28
- 8 ビットスタティックグレーリニア画像, 32
- 8R 画像, 30
- 8X 画像, 30

## A

- `afbconfig` ユーティリティー, 75

## B

- BadAlloc エラーメッセージ, 38

## C

- CDE, Creator グラフィックスアクセラレータ上での実行, 36
- `cgsix` フレームバッファー, 5
- Creator グラフィックスアクセラレータ, 9 ~ 25
  - XIL の高速化, 45 ~ 49

- ウィンドウシステム, 27 ~ 44
- ウィンドウマネージャーについて, 34
- オーバーレイとアンダーレイの構造, 30
- カーソルの使用方法, 39
- 画像表示形式, 28 ~ 30
- ステレオコネクタ, 23
- 製品仕様, 41
- デフォルトの画面解像度, 10, 35

## D

- DBZ (Creator 3D), 28
- `defdepth` モード, 37
- Direct Xlib, Creator アクセラレータによるサポートなし, 41

## E

- Elite3D グラフィックスアクセラレータ, 75 ~ 90
  - ステレオコネクタ, 89
  - デフォルトの画面解像度, 76
- Expert3D Graphics Accelerator, 91 ~ 108
  - changing screen resolution, 95
  - configuration utilities, 91
  - connectors, 103
  - default screen resolutions, 93
  - default visual, 97
  - extended overlay, 100

gamma correction, 99  
multiple monitors, 100  
stereo, 95  
supported screen resolutions, 93  
visual list order, 96

## F

fbconfig utility, 92  
FFB (Fast Frame Buffer), 28  
fbconfig ユーティリティ, 9, 31, 40

## G

GFXconfig ユーティリティ, 59

## I

ifbconfig utility, 92

## M

m640、組み込みグラフィックスデバイス, 73  
m64config ユーティリティ, 53 ~ 57  
maxwids, 31, 38  
MBX によるダブルバッファリングのサポート, 39  
MITSHM 拡張機能, 41

## N

nvedit、NVRAM の変更を使用する, 67, 74  
nvramrc, 3  
NVRAM、nvedit で変更する, 67, 74

## O

OpenBoot PROM, 115

## OpenWindows

複数のモニターの使用, 113  
OWconfigファイル, 9, 75

## P

PGX グラフィックスアクセラレータ  
ビデオモード, 64  
, 53 ~ 57  
サポートする画面解像度, 54  
PGX32 グラフィックスアクセラレータ  
PGX32 上の共通デスクトップ環境 (CDE), 72  
画面解像度, 65  
コンソールとして, 71, 72  
設定ウィンドウ, 60  
ビデオタイミング, 66  
, 59 ~ 74  
PGX32 上の共通デスクトップ環境 (CDE), 72  
PROM による 1 台のモニターの設定, 7  
PROM によるモニター設定, 6

## S

SB (Creator), 28  
sbus-probe-list, 116  
ShowMe Whiteboard Snap Region でのドラッグ  
速度の低下, 42  
SunForum カード, 72

## T

TurboGXplus フレームバッファ, 1 ~ 8  
画面解像度, 2  
サポートされているモニター, 1  
デフォルトの画面解像度, 2

## U

UltraSPARC Visual Instruction Set (VIS), 45

UNIX スクリプトによる 1 台のモニターの設定, 7  
UNIX スクリプトによるモニターの設定  
(TurboGXplus フレームバッファ), 5

## V

Visual Instruction Set (VIS), 45

## X

X 共有メモリー転送機能, 41  
X ディスプレイマネージャ (xdm), 73  
X チャネルアーキテクチャー, 31  
X11 perf のベンチマークテスト, 42  
XGetVisualInfo, 32, 33  
XGL によるダブルバッファリングのサポート, 39  
XGLNOPEX 環境変数, 57  
XIL  
    Creator グラフィックスアクセラレータにおける高速化, 45~49  
    高速化された関数, 46  
    データの種類, 45  
XSolarisGetVisualGamma, 32  
XSolarisOvlSelectBestOverlay, 33

## あ

アンダーレイの 8 ビット疑似カラー画像, 30

## う

ウィンドウシステム  
    Creator グラフィックスアクセラレータ, 27~44

## お

オーバーレイの 8 ビット疑似カラー画像, 30  
オーバーレイのピクセルコード, 37

## か

画像分子, 46  
画面解像度  
    Creator および Creator 3D グラフィックスアクセラレータ, 11  
    Elite3D グラフィックスアクセラレータ, 77, 93  
    PGX グラフィックスアクセラレータ, 54  
    TurboGXplus フレームバッファ, 2  
カラー LUT, 32  
カラーマップのフラッシュ, 35  
カラーマップのフラッシュ現象の減少, 28  
ガンマ補正, 32

## こ

高速化された XIL 関数, 46

## さ

サポートされるモニター  
    TurboGXplus フレームバッファ, 1  
サン製品以外のモニター  
    Creator グラフィックスアクセラレータ, 10  
    Elite3D グラフィックスアクセラレータ, 76  
    TurboGXplus フレームバッファ, 1

## す

ステレオケーブル, 90  
ステレオコネクタ, 23, 89

## た

ダブルバッファのサポート, 50  
ダブルバッファリング, ハードウェア, 39

## て

デバイスファイル名, 111

デフォルトの画像表示形式

Creator グラフィックスアクセラレータ, 35

デフォルトの画面解像度

Creator グラフィックスアクセラレータ, 10

Elite3D グラフィックスアクセラレータ, 76

TurboGXplus フレームバッファ, 1

り

リニア画像, 15, 32, 79, 80, 96

利用可能なフレームバッファ, 112

は

ハードウェアカーソル, 39

ハードウェアのウィンドウ ID (WID), 37

ハードウェアのダブルバッファリング, 39

ひ

ビデオタイミング、PGX32 グラフィックスアクセラレータ, 66

ビデオモード、PGX32 グラフィックスアクセラレータ, 64

非リニア画像, 15, 79, 81

ふ

複数のモニター, 109 ~ 117

OpenWindows, 113

TurboGXplus フレームバッファ上での画面解像度のプログラミング, 3

設定, 109

不透明なピクセル, 37

ほ

ポーリング順序を変更する, 115

も

モニター

TurboGXplus によるサポート, 1

モニター ID センスコード, 10, 76