

MS104-SH4

MS104-VGA/LCD の使用方法

1版 2005年02月18日

目次

ダイジェスト版

1. 概要	1
1.1 Linux について	1
1.2 X Window System システムについて	1
1.3 Microwindows について	2
2. MS104-VGA/LCD について	3
2.1 MS104-VGA/LCD の概要	3
2.2 MS104-VGA/LCD とモニタの接続	4
2.3 MS104-VGA/LCD の出力対応	5
2.4 デバイスドライバについて	6
3. MS104-VGA/LCD の起動	10
3.1 MS104-VGA/LCD の動作環境	10
3.2 MS104-VGA/LCD の設定	11
3.3 MS104-VGA/LCD の動作	12
4. RedBoot	36
4.1 RedBoot の概要	36
4.2 RedBoot のコンフィグレーション	37
4.3 RedBoot のコンパイル	39
4.4 RedBoot の設定	47
5. Linux	51
5.1 Linux カーネルの概要	51
5.2 Linux カーネルのコンフィグレーション	52
5.3 アドレス・割込み番号の設定	54
5.4 Linux カーネルのコンパイル	55
6. Microwindows	57
6.1 Microwindows 概要	57
6.2 Microwindows の開発について	59
6.3 Microwindows のコンフィグレーション	60
6.4 Microwindows のインストール	62
6.5 Microwindows アプリケーションプログラムの作成	66
6.5 タッチパネルのキャリブレーション	72

7. 保証とサポート	77
付録 A Microwindows API 一覧	78

1. 概要

本アプリケーションノートはグラフィックコントローラボード「MS104-VGA/LCD」を MS104-SH4 用 Linux で使用方法について述べます。

MS104-VGA/LCD は、S1D13506 (セイコーエプソン社) グラフィックコントローラを搭載した PC/104 準拠周辺ボードです。MS104-VGA/LCD は、VGA、LCD、NTSC、S-Video 出力をサポートし、タッチパネルコントローラ、プザーデバイスをボード上に実装しています。

MS104-SH4 と MS104-VGA/LCD を組み合わせることにより高度な GUI を備えたシステムを容易に構築することができます。本アプリケーションノートでは、MS104-SH4 用 Linux を使用して、組み込み用ウィンドウシステム「Microwindows」の動作方法、および、サンプルプログラム作成について説明します。

本アプリケーションノートを実行するには、必ず「MS104-SH4 Linux 開発環境キット Linux-KIT-A02」がインストールされている必要があります。

1.1 Linux について

Linux とは 1991 年に Linus Torvalds 氏によって開発された、オープンソースの UNIX 互換オペレーティングシステムです。Linux はオープンソース、ロイヤリティフリーという特性から、世界中のプログラマたちにより日々改良され、現在では Windows を脅かす存在にまで成長しました。今では大手企業のサーバーや、行政機関などにも広く採用されています。また、Linux の特長として CPU アーキテクチャに依存しないということがあげられます。そのため、数多くのターゲット(CPU)に移植されており、デジタル家電製品を中心に非 PC 系製品にも採用されるようになりました。Linux の詳細については、一般書籍やインターネットから多くの情報を得られますので、それらを参考にしてください。

1.2 X Window System について

X Window System はマサチューセッツ工科大学の Athena Widget Project が中心となって開発した UNIX 系 OS で利用されるグラフィカルユーザインターフェース(GUI)環境のことです。現在、X Window System の開発は The Open Group で行われています。

X Window System の特長は OS と完全に分離したソフトウェアであり、クライアント・サーバ型のソフトウェアであるということです。X Window System は OS を透過して直接グラフィックメモリにアクセスすることにより、処理速度の向上をしています。そのため、X Window System が何らかの原因で処理が中断すると OS もハングしてしまいます場合があります。

X Window System のサーバはキーボードやモニター、マウスなどの入出力デバイスの処理を行い、クライアントは各アプリケーションの処理、つまり、ウィンドウの表示を行うプログラムの処理を担当します。

X Window System ではモニタ上でウィンドウがどのように見えるかや、マウスがウィンドウをどのように動かすかなど、見栄えや操作方法については定義されていません。そのため、ウィンドウマネージャと呼ばれるアプリケーションプログラムが見栄えや操作方法を定義、管理します。

1.3 Microwindows について

Microwindows は Century Software 社のオープンソースのプロジェクトで、組み込み機器向けに設計されたウィンドウシステムです。Microwindows は 3 階層で設計されており、最下層はモニタやマウス、キーボードなどの入出力デバイスのドライバです。中間層は線描、領域塗りつぶし、クリッピング、カラーモデルなどのグラフィックエンジン、最上位層は 2 つの API をサポートしています。

API は Microwindows と呼ばれる Win32 API を模した API と Nano-X と呼ばれる X Window System の API を模した API の 2 種類です。Nano-X は Linux で採用されています。

Microwindows はフレームバッファ (グラフィックスイメージを保持するメモリ領域) をサポートしているため、非常に高速に動作し、PC 用の X Window System に比べ、Nano-X サーバのサイズが 100~200KB 程度と非常にサイズが小さくなります。

2. MS104-VGA/LCD について

2.1 MS104-VGA/LCD の概要

MS104-VGA/LCD は VGA、LCD、NTSC、S-Video 出力をサポートし、タッチパネルコントローラ、ブザーデバイスも実装した PC/104 周辺ボードです。

MS104-VGA/LCD は MS104-SH4 CPU ボードと組み合わせることにより、高度な GUI を構築することが可能です。

MS104-SH4 用 Linux (apLinux) には MS104-VGA/LCD 用デバイスドライバが組み込まれており、組み込み用ウィンドウシステム『Microwindows』と組み合わせれば、Linux ベースのウィンドウシステムを作成することができます。

また、弊社 LVDS/DVI ボード『MS104-LVDS/DVI』と LCD キット『LCD-KIT-A02』を組み合わせることにより、LCD を使用した開発環境を構築することができます。

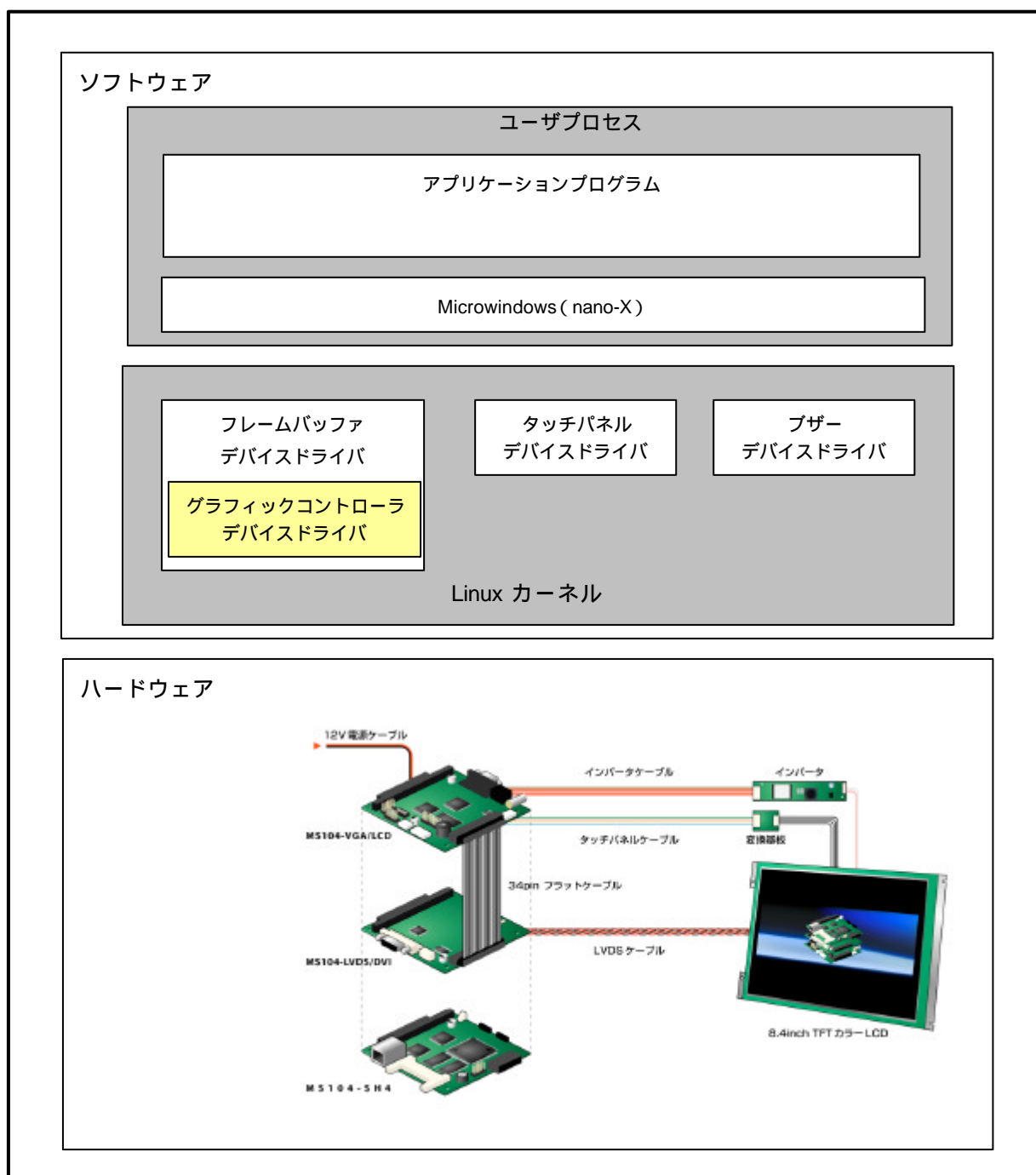


Fig 2.1-1 MS104-VGA/LCDの概要

2.2 MS104-VGA/LCD とモニタの接続

MS104-VGA/LCD を VGA、LCD、NTSC、S-Video 出力をサポートしています。各モニタと接続する際は下図を参考にしてください。

下図に LCD および DVI モニタに接続例を示します。LCD および DVI モニタの接続にはディスプレイインターフェイスボード『MS104-LVDS/DVI』とタッチパネル付 TFT-LCD キット『LCD-KIT-A02』が必要となります。

LCD、DVI は同時に出力することができます。

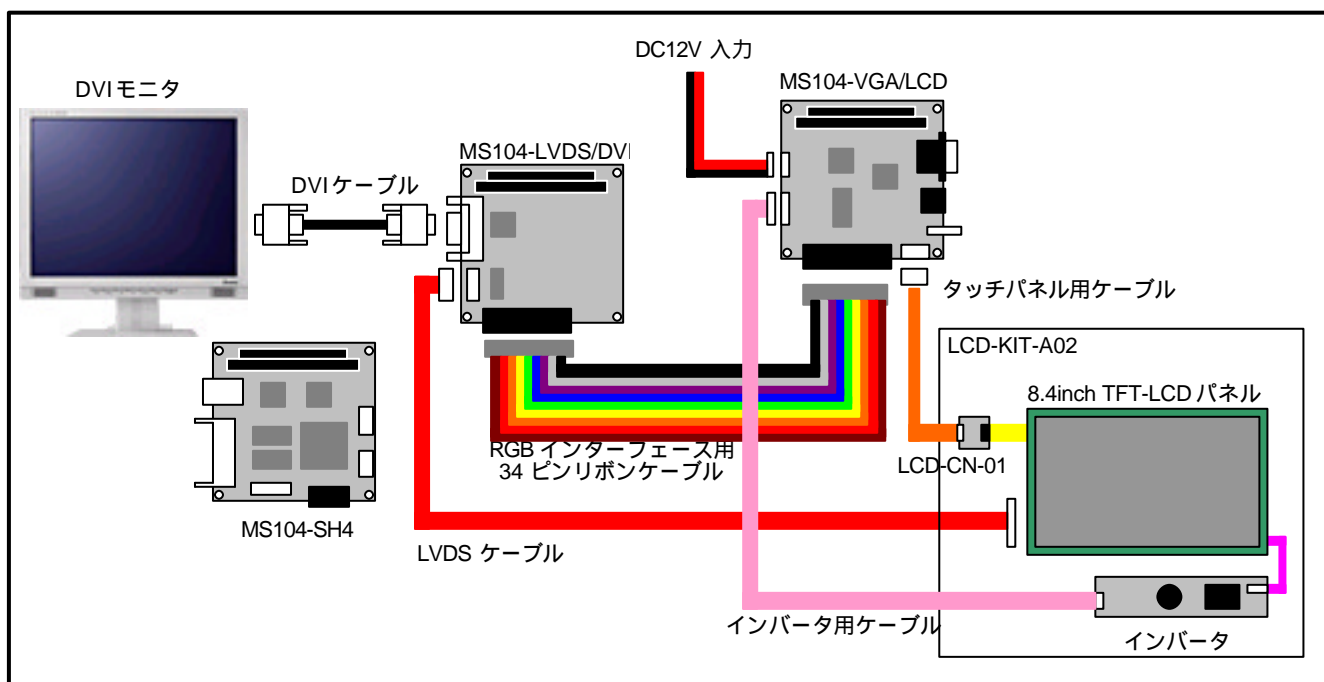


Fig 2.2-1 LCD、DVI との接続

下図に VGA、NTSC、S-Video モニタの接続例を示します。

VGA、NTSC、S-Video は同時に出力することができません。出力先はソフトウェア（RedBoot、Linux）により選択されます。

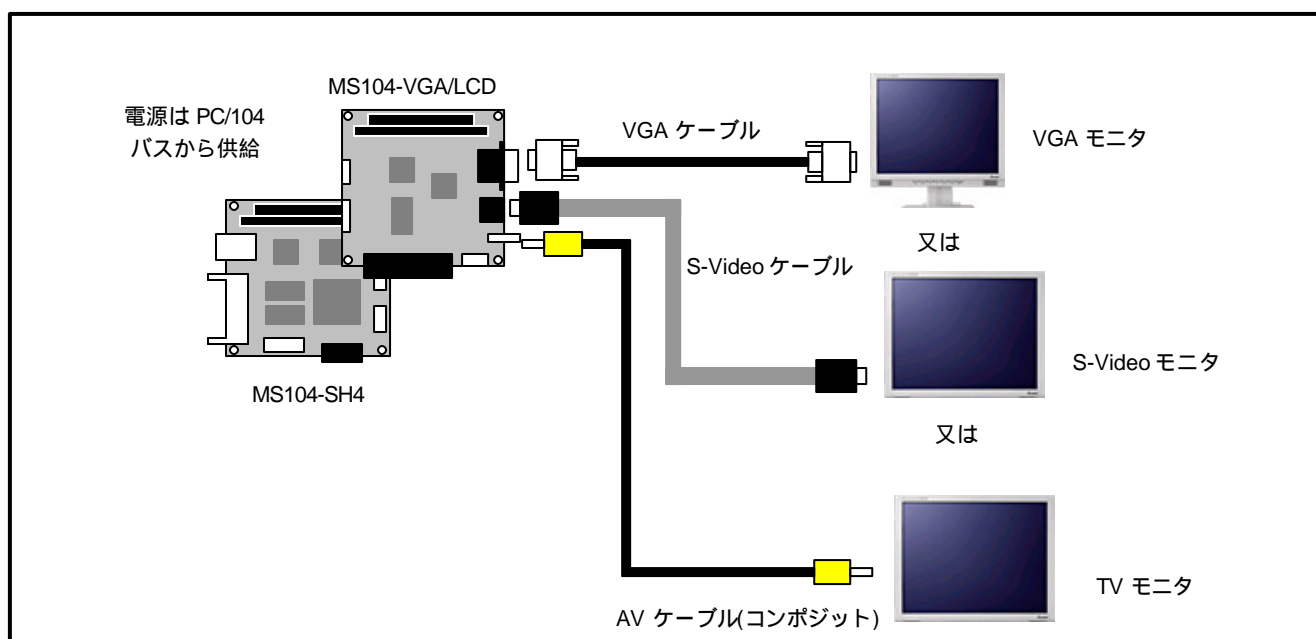


Fig 2.2-2 VGA、NTSC、S-Video との接続

2.3 MS104-VGA/LCD の出力対応

MS104-SH4 用 RedBoot・Linux で対応している出力の一覧を示します。

出力先	解像度	色深度	RedBoot		Linux	
			垂直 周波数 (Hz)	水平 周波数 (kHz)	垂直 周波数 (Hz)	水平 周波数 (kHz)
LCD	640 × 480 1	8	47	24.7	47	24.7
		15	47	24.7	-	-
		16	47	24.7	47	24.7
	800 × 600	8	60.3	37.9	60.3	37.9
		15	60.3	37.9	-	-
		16	60.3	37.9	60.3	37.9
VGA	640 × 480	8	60.1	29.5	60.1	29.5
		15	60.1	29.5	-	-
		16	60.1	29.5	60.1	29.5
	800 × 600	8	60.3	37.9	60.3	37.9
		15	60.3	37.9	-	-
		16	60.3	37.9	60.3	37.9
NTSC	640 × 480	8	62	15.7	62	15.7
		15	62	15.7	-	-
		16	62	15.7	62	15.7
S-Video	640 × 480	8	62	15.7	62	15.7
		15	62	15.7	-	-
		16	62	15.7	62	15.7
LCD & VGA	640 × 480 1	8	31	16.5	31	16.5
			60.1	29.5	60.1	29.5
	800 × 600	8	20.1	12.6	20.1	12.6
			60.3	37.9	60.3	37.9
LCD & NTSC	640 × 480 1	8	31	16.5	31	16.5
			62	15.7	62	15.7
LCD & S-Video	640 × 480 1	8	31	16.5	31	16.5
			62	15.7	62	15.7

Table 2.3-1 MS104-VGA/LCD出力対応

- 1 弊社「LCD-KIT-A02」では解像度が異なるため、ご使用いただけません。

2.4 デバイスドライバについて

MS104-VGA/LCD ボードに対して、RedBoot では Hardware Abstrat Layer(HAL)上に出力用コードを追加し、Linux ではフレームバッファ、タッチパネル、ブザーデバイスドライバを追加しています。

この章では、デバイスドライバの概要について説明します。

MS104-VGA/LCD の色深度、解像度はソフトウェア(RedBoot、Linux、Microwindows)に依存します。そのため、各設定に合わせてソフトウェアを変更する必要があります

RedBoot

RedBoot は MS104-VGA/LCD ボードに対し、ロゴの出力を行います。解像度、色深度、出力先といった設定はフラッシュ ROM に格納されているコンフィグレーションデータによって決まります。また、ロゴは XPM (X11 Pixmap) 形式の画像ファイルのみ使用することができ、このロゴファイルはコンパイル後変更することができません。

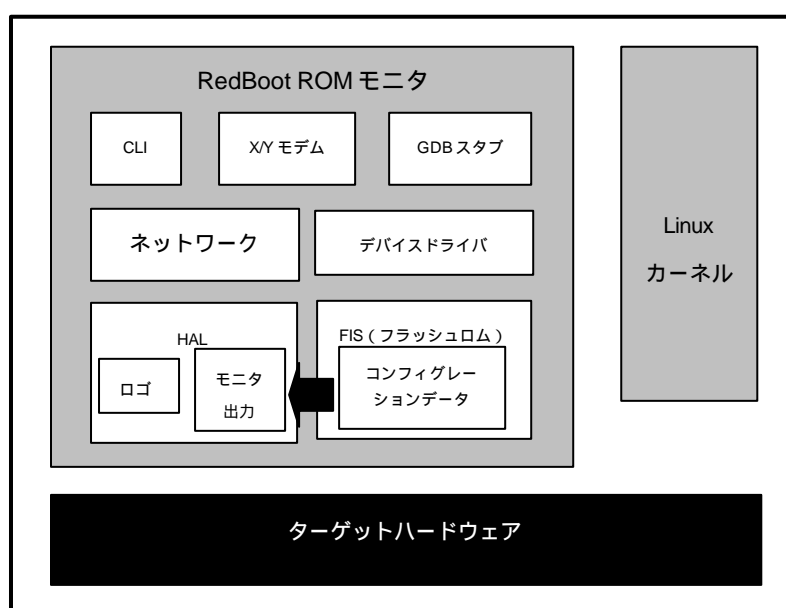


Fig 2.4-1 RedBoot アーキテクチャ

Table 2.4-2 RedBoot コンフィグレーションデータ

設定			RedBoot コンフィグレーションデータ						
ディスプレイ	解像度	色深度	ms104vga_bpp	ms104vga_res	ms104vga_output_lcd	ms104vga_tput_lcd_sim	ms104vga_output_ntsc	ms104vga_out_put_svideo	ms104vga_output_vga
VGA	640x480 1	8	8	640480	false	false	false	false	true
		15	15	640480	false	false	false	false	true
		16	16	640480	false	false	false	false	true
	800x600	8	8	800600	false	false	false	false	true
		15	15	800600	false	false	false	false	true
		16	16	800600	false	false	false	false	true
LCD	640x480	8	8	640480	true	false	false	false	false
		15	15	640480	true	false	false	false	false
		16	16	640480	true	false	false	false	false
	800x600	8	8	800600	true	false	false	false	false
		15	15	800600	true	false	false	false	false
		16	16	800600	true	false	false	false	false
NTSC	640x480	8	8	640480	false	false	true	false	false
		15	15	640480	false	false	true	false	false
		16	16	640480	false	false	true	false	false
S-Video	640x480	8	8	640480	false	false	false	true	false
		15	15	640480	false	false	false	true	false
		16	16	640480	false	false	false	true	false
LCD & VGA	640 x 480	8	8	640480	false	true	false	false	true
	800 x 600	8	8	800600	false	true	false	false	true
LCD & NTSC	640 x 480	8	8	640480	false	true	true	false	false
LCD & S-Video	640 x 480	8	8	640480	false	true	false	true	false

Fig 2.4-3 RedBoot グラフィックデバイスドライバソースファイル構成

```

RedBoot ソースコードルートディレクトリ
|--packages
  |--hal
    |--sh
      |--ms104sh4
        |-- v2_0
          |-- include
            |-- s1d13506.h           : グラフィックコントローラ (S1D13506) 用ヘッダファイル
            |-- vga_lcd_support.h   : MS104-VGA/LCD ボード用ヘッダファイル
          |-- src
            |-- banner.xpm          : ロゴ画像ファイル
            |-- vga_lcd_support.c   : MS104-VGA/LCD ボード用ソースファイル
    
```

RedBoot ソースファイルは弊社 Linux 開発環境キット「Linux-KIT-A02」、もしくは、弊社組込み Linux ホームページ <http://www.apnet.co.jp/e-linux/index.html> から入手することができます。

Linux

MS104-VGA/LCD 用に加えられたデバイスドライバはフレームバッファデバイスドライバ、タッチパネルデバイスドライバ、ブザーデバイスドライバがあります。

また、各モニタ出力の設定は、Linux 起動時にカーネルパラメータによって変更することができます。

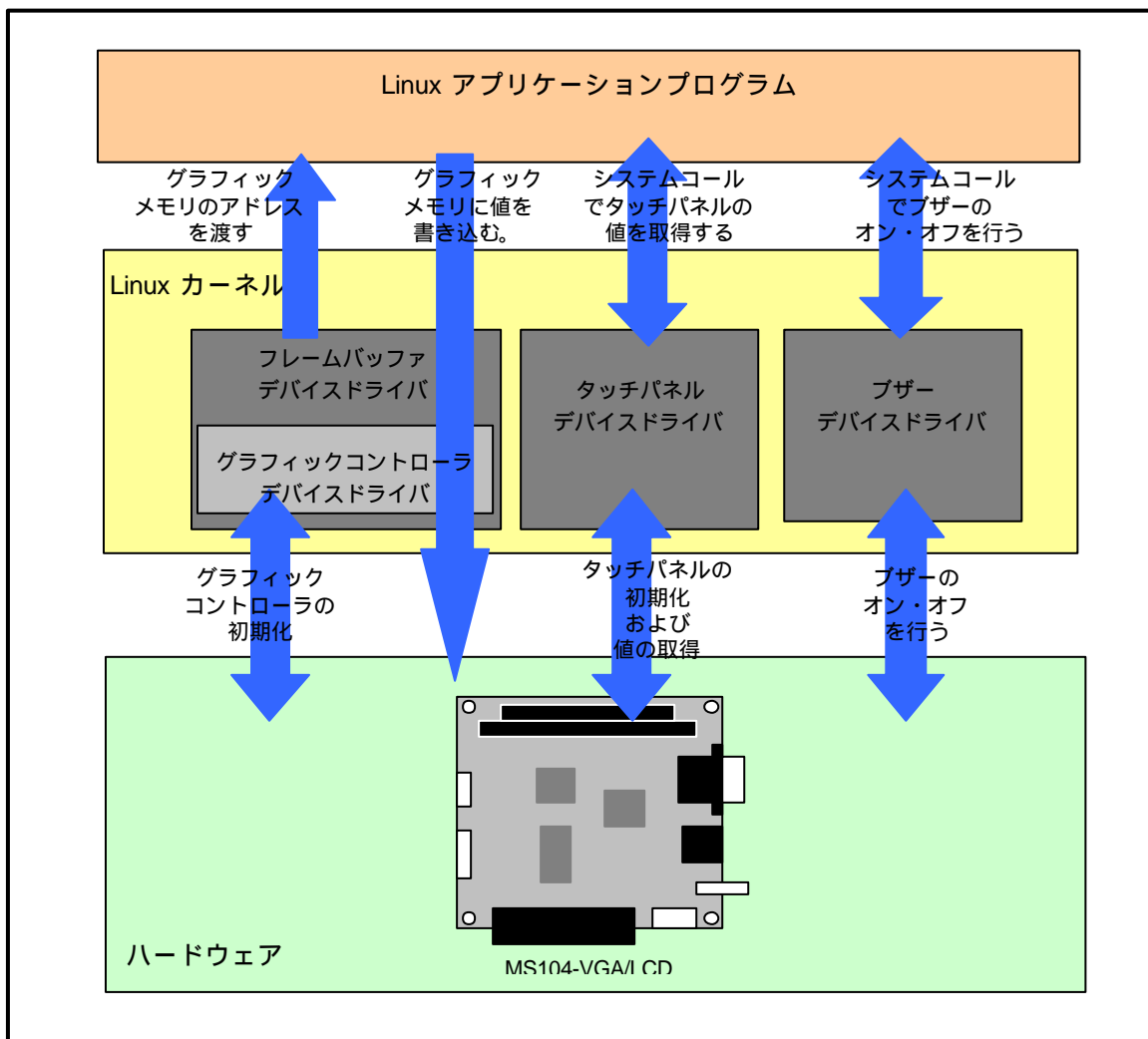


Fig 2.4-4 Linux デバイスドライバ

Table 2.4-5 Linux デバイスドライバ

デバイス名	デバイスファイル	メジャー番号	マイナー番号	アドレス	割り込み番号
フレームバッファ	/dev/fb0	29	0	0x400000	-
タッチパネル	/dev/mk712	10	15	0xa00	11
ブザー	/dev/buzzer	10	254	0xa08	-

各出力に対するカーネルパラメータの一覧を示します。

Table 2.4-6 Linux デバイスドライバ出力対応

設定			カーネルパラメータ
ディスプレイ	解像度	色深度	
VGA	640x480	8	video=e1356fb:system:ms104sh4,display:crt,bpp:8,640x480@60
		16	video=e1356fb:system:ms104sh4,display:crt,bpp:16,640x480@60
	800x600	8	video=e1356fb:system:ms104sh4,display:crt,bpp:8,800x600@60
		16	video=e1356fb:system:ms104sh4,display:crt,bpp:16,800x600@60
LCD	640x480	8	video=e1356fb:system:ms104sh4,display:tft,bpp:8,640x480@47
		16	video=e1356fb:system:ms104sh4,display:tft,bpp:16,640x480@47
	800x600	8	video=e1356fb:system:ms104sh4,display:tft,bpp:8,800x600@60
		16	video=e1356fb:system:ms104sh4,display:tft,bpp:16,800x600@60
NTSC	640x480	8	video=e1356fb:system:ms104sh4,display:ntsc,bpp:8,tvfmt:0,tvfilt:7,640x480@62
		16	video=e1356fb:system:ms104sh4,display:ntsc,bpp:16,tvfmt:0,tvfilt:7,640x480@62
S-Video	640x480	8	video=e1356fb:system:ms104sh4,display:ntsc,bpp:8,tvfmt:1,tvfilt:7,640x480@62
		16	video=e1356fb:system:ms104sh4,display:ntsc,bpp:16,tvfmt:1,tvfilt:7,640x480@62
LCD & VGA	640 x 480	8	video=e1356fb:system:ms104sh4,display:crt,bpp:8,simult,640x480@60
	800 x 600	8	video=e1356fb:system:ms104sh4,display:crt,bpp:8,simult,800x600@60
LCD & NTSC	640 x 480	8	video=e1356fb:system:ms104sh4,display:ntsc,bpp:8,tvfmt:0,tvfilt:7,simult,640x480@62
LCD & S-Video	640 x 480	8	video=e1356fb:system:ms104sh4,display:ntsc,bpp:8,tvfmt:1,tvfilt:7,simult,640x480@62

1 弊社タッチパネル付 TFT-LCD モニタ「LCD-KIT-A02」では解像度が異なるため、ご使用いただけません。

Fig 2.4-7 Linux フレームバッファデバイスドライバソースファイル構成

```
Linux カーネルソースディレクトリ
|--drivers
  |--video
    |-- dummycon.c      : ダミーコンソールドライバ
    |-- epson1356fb.c   : EPSON S1D13506 グラフィックコントローラデバイスドライバ
    |-- epson1356fb.h   : EPSON S1D13506 グラフィックコントローラデバイスドライバ用ヘッダファイル
    |-- fbcmap.c        : フレームバッファ用カラーマップ
    |-- fbcon-cfb16.c   : 16 ビット True カラー用低レベルフレームバッファドライバ
    |-- fbcon-cfb8.c    : 8 ビットパワードピクセル用低レベルフレームバッファドライバ
    |-- fbcon.c         : 低レベルフレームバッファコンソールドライバ
    |-- fbmem.c         : フレームバッファ mmap システムコールの実装
    |-- font_8x16.c     : 8 x 16 ピクセルフォント
    |-- font_8x8.c      : 8 x 8 ピクセルフォント
    |-- fonts.c         : フォント定義ファイル
    |-- modedb.c        : スタンダードビデオモードのデータベース管理
```

Fig 2.4-8 Linux タッチパネルデバイスドライバソースファイル構成

```
Linux カーネルソースディレクトリ
|--drivers
  |--char
    |--mk712.c      : タッチパネルドライバ
```

Fig 2.4-9 Linux ブザーデバイスドライバソースファイル構成

```
Linux カーネルソースディレクトリ
|--arch
  |--sh
    |--kernel
      |--buzzer_ms104vga.c : ブザーデバイスドライバ
```

Linuxソースファイルは弊社Linux開発環境キット『Linux-KIT-A02』、もしくは弊社MS104-SH4 ファイルセンター『<http://www.apnet.co.jp/e-linux/dl/filecenter/index.html>』から入手することができます。

MS104-SH4 ファイルセンター『<http://www.apnet.co.jp/e-linux/dl/filecenter/index.html>』にアクセスするには、ユーザ登録『https://www.apnet.co.jp/cgi-bin/reg_ms104/index.html』が必要です。

3. MS104-VGA/LCD の起動

この章では、「MS104-VGA/LCD」と「MS104-SH4」を使用して、各モニタに Microwindows のデモプログラムを出力する手順を説明します。

3.1 MS104-VGA/LCD の動作環境

ホスト PC

RedBoot/Linux のコンソール、および、TFTP、NFS サーバとして使用します。シリアルポート、ネットワーク、TFTP、NFS サーバが使用可能な PC をご用意ください。

電源

MS104-VGA/LCD は PC/104 バスから電源の配給を受けることができます。MS104-SH4 に必要な電源は DC5V ± 5% です。MS104-VGA/LCD と合わせて使用するため、2A 程度の電源をご用意ください。

LAN

NFS を使用してホスト PC と MS104-SH4 でデータのやり取りを行います。MS104-SH4 をネットワークに接続できる LAN ケーブルをご用意ください。

Table 3.1-1 MS104-VGA/LCD、MS104-SH4 の推奨動作環境

使用機器等	環 境
PC/104 グラフィックボード	MS104-VGA/LCD
CPU ボード	MS104-SH4
HOST PC	PC/AT 互換機
OS	Linux(推奨 FedoraCore1)
メモリ	使用 OS による
ソフトウェア	ターミナルソフト TFTPサーバ NFSサーバ
ドライブ	CD-R読み込み可能なドライブ
LANポート	10Base-T or 100Base-TX 1ポート
RS232C ケーブル	クロスケーブルを使用
シリアル変換コネクタ	MS104-SH4 付属品
LAN ケーブル	ホスト PC と接続時はクロスケーブルを使用 ハブと接続時はストレートケーブルを使用
VGA モニタ & ケーブル	垂直周波数 60.1Hz 水平周波数 29.5kHz 対応モニタ
電源	DC5V ± 5% 2A 程度

3.2 MS104-VGA/LCD の設定

MS104-SH4 用に MS104-VGA/LCD ボードの設定を行います。

メモリアドレスを H'400000 に設定します。SW1 の 1 番を ON、2 番を OFF にして下さい。

I/O アドレスを H'0a00 に設定します。SW1 の 3 番を OFF、4 番を ON、5 番を OFF、6 番を ON にして下さい。

PC/104 バスの BALE を有効にします。SW1 の 7 番を ON にして下さい。

インターフェースモードの設定を PC/104 互換モードにします。SW1 の 8 番を OFF にして下さい。

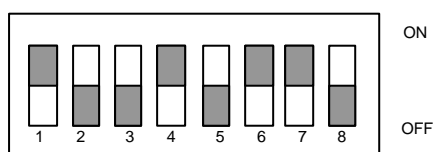


Fig 3.2-1 SW1 の設定

タッチパネルの割込みに IRQ4 を利用します。JP1 の 4 番にジャンパピンを接続してください。

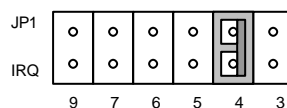


Fig 3.2-2 JP1 の設定

LCD コントローラ (EPSON 社製 S1D13506) を ISA バス (PC/104 バス) モードにします。JP3 の 1 番、3 番、4 番 15 番にジャンパピンを接続してください。

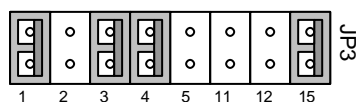


Fig 3.2-3 JP3 の設定

3.3 MS104-VGA/LCD の動作

この章では、各モニタに対し、色深度 8bpp 対応の Microwindows デモプログラムを起動する手順を示します。

MS104-VGA/LCD 対応 Linux カーネル

MS104-VGA/LCD を動作させる前に、MS104-VGA/LCD に対応した Linux カーネルを MS104-SH4 搭載のフラッシュロムに格納しておきます。

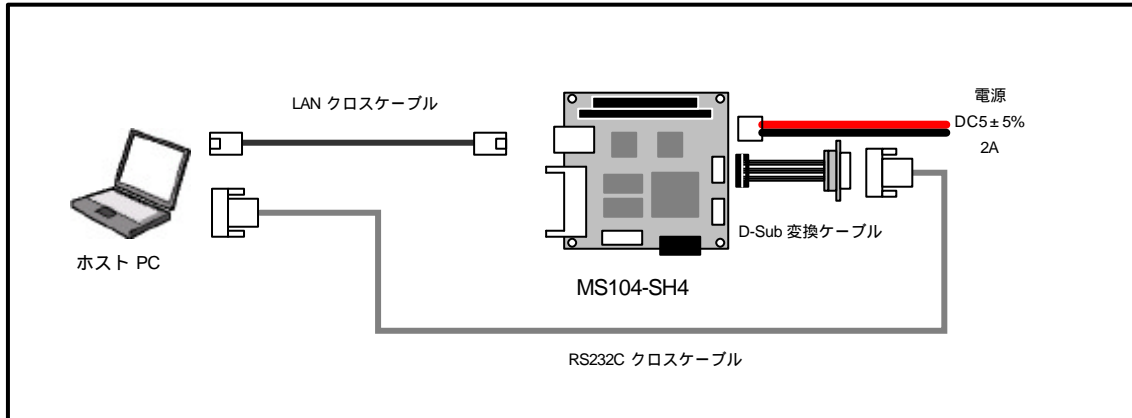


Fig 3.3-1 MS104-SH4 の接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROM もしくは、「MS104-SH4 ファイルセンター」から MS104-VGA/LCD 用 Linux カーネルと RAM ディスクイメージ (ルートファイルシステム) を取得し、TFTP サーバディレクトリ 『/tftpboot』にコピーします。

下記のコマンドは「MS104-SH4」の添付 CD-ROM から Linux カーネルと RAM ディスクイメージ (ルートファイルシステム) をコピーする場です。

```
[alpha@ap_linux root]# mount /dev/cdrom /mnt/cdrom 
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/vmlinuz-ms104sh4-x.x-vga /tftpboot/ 
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/ramdisk-ms104sh4-x.x-vga.gz /tftpboot/ 
```

TFTP サーバの起動については MS104-SH4 Linux 導入マニュアル『MS104-SH4 Linuxstart.pdf』をご覧ください。MS104-SH4 Linux 導入マニュアル『MS104-SH4 Linuxstart.pdf』、もしくは弊社 MS104-SH4 ファイルセンター『<http://www.apnet.co.jp/e-linux/dl/filecenter/index.html>』から入手することができます。MS104-SH4 ファイルセンター『<http://www.apnet.co.jp/e-linux/dl/filecenter/index.html>』にアクセスするには、ユーザ登録『https://www.apnet.co.jp/cgi-bin/reg_ms104/index.html』が必要です。「x.x」はバージョン番号を示します。バージョン 2.0 の場合は「2.0」になります。

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-1 MS104-SH4 の接続」を参考に接続します。

MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF)と Ethernet ポートをそれぞれ、ホスト PC のシリアルポートと Ethernet ポートに接続してください。

MS104-SH4 の電源を ON にし、RedBoot の起動ログが表示されたら、自動起動を停止するために『Ctrl+C』とタイプします。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.
== Executing boot script in 5.000 seconds - enter ^C to abort
^C
RedBoot>
```

RedBoot に Linux カーネルイメージをダウンロードし、フラッシュROMに格納します。

TFTPサーバの IP アドレスは「192.168.1.201」と仮定します。

```
RedBoot> load -r -b 0x8c210000 -m tftp -h 192.168.1.201 vmlinuz-ms104sh4-x.x-vga
Raw file loaded 0x8c210000-0x8c2fd006, assumed entry at 0x8c210000
RedBoot> fis create -b 0x8c210000 -l 0x100000 -r 0x8c210000 -e 0x8c210000 -f 0x80040000 vmlinuz
An image named 'vmlinuz' exists - continue (y/n)? y
... Erase from 0x80040000-0x80140000: .....
... Program from 0x8c210000-0x8c310000 at 0x80040000: .....
... Unlock from 0x80fe0000-0x81000000: .
... Erase from 0x80fe0000-0x81000000: .
... Program from 0x8dfdf000-0x8dfff000 at 0x80fe0000: .
... Lock from 0x80fe0000-0x81000000: .
RedBoot>
```

「x.x」はバージョン番号を示します。バージョン 2.0 の場合は「2.0」になります。

RedBoot に RAM ディスクイメージをダウンロードし、フラッシュROMに格納します。

TFTPサーバの IP アドレスは「192.168.1.201」と仮定します。

```
RedBoot> load -r -b 0x8c360000 -m tftp -h 192.168.1.201 ramdisk-ms104sh4-x.x-vga.gz
Raw file loaded 0x8c360000-0x8c6d02fd, assumed entry at 0x8c360000
RedBoot> fis create -b 0x8c360000 -l 0x400000 -r 0x8c360000 -e 0x8c360000 -f 0x80140000 ramdisk.gz
An image named 'ramdisk.gz' exists - continue (y/n)? y
... Erase from 0x80140000-0x80540000: .....
... Program from 0x8c360000-0x8c760000 at 0x80140000: .....
... Unlock from 0x80fe0000-0x81000000: .
... Erase from 0x80fe0000-0x81000000: .
... Program from 0x8dfdf000-0x8dfff000 at 0x80fe0000: .
... Lock from 0x80fe0000-0x81000000: .
RedBoot>
```

「x.x」はバージョン番号を示します。バージョン 2.0 の場合は「2.0」になります。

RedBoot の『**fconfig**』コマンドを使用して、『**libboot**』コマンドで Linux カーネルが起動するよう RedBoot のコンフィグレーションデータを変更します。

```
RedBoot> fconfig
Run script at boot: false
Use BOOTP for network configuration: false
Gateway IP address: 0.0.0.0
Local IP address: 192.168.1.200
Local IP address mask: 255.255.255.0
Default server IP address: 0.0.0.0
Console baud rate: 38400
DNS server IP address: 0.0.0.0
GDB connection port: 9000
Force console for special debug messages: false
Booting Linux Kernel at Power ON: false
Load Linux Kernel & File System: true
Linux Kernel Destination address: 0x8c210000
Linux Kernel Source address: 0x80040000
Linux Kernel Size: 0x100000
RAM disk Destination address: 0x8c360000
RAM disk Source address: 0x80140000
RAM disk Size: 0x400000
Modify Linux Kernel Parameter: false
Use VGA/LCD monitor(MS104-VGA/LCD): false
Network debug at boot time: false
Update RedBoot non-volatile configuration - continue (y/n)? y
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .
RedBoot>
```

『**libboot**』コマンドを使用して、Linux が起動するか確認します。

```
RedBoot> libboot
Now loading Linux kernel:
Linux kernel source address : 0x80040000
Linux kernel destination address : 0x8c210000
Linux kernel size : 0x00100000
.
.
Uncompressing Linux... Ok, booting the kernel.
.
.
```

VGA

解像度 800 × 600、色深度 8bpp 対応の Linux カーネルを起動し、Microwindows のデモプログラムを動作させます。
 以下に MS104-SH4、MS104-VGA/LCD および VGA モニタの接続を示します。

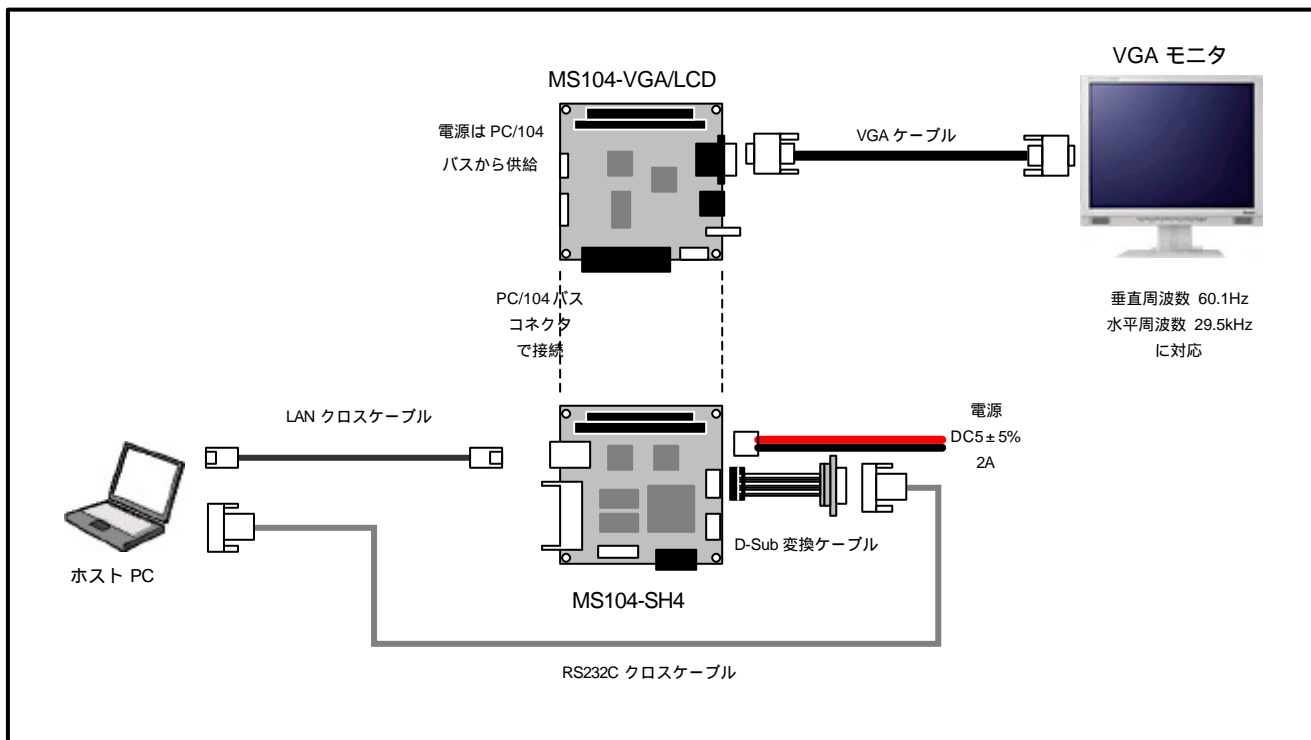


Fig 3.3-2 VGA モニタとの接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROM から Microwindows デモプログラム『mdemo』を NFS サーバディレクトリ『/nfs』にコピーします。

NFS サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。

```
[alpha@ap_linux root]# mount /dev/cdrom /mnt/cdrom 
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/mdemo /nfs/. 
```

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-2 VGA モニタとの接続」を参考に接続します。

MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF)と Ethernet ポートをそれぞれ、ホスト PC のシリアルポートと Ethernet ポートに接続してください。

MS104-SH4 の電源を ON し、RedBoot の起動ログが表示されることを確認します。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot の『fconfig』コマンドを使用して、MS104-VGA/LCD の VGA 出力に対応するよう RedBoot のコンフィグレーションデータを変更します。

『fconfig ms104vga true』を実行して、RedBoot の MS104-VGA/LCD サポート機能を有効にします。

```
RedBoot> fconfig ms104vga true
ms104vga: Setting to true
Update RedBoot non-volatile configuration - continue (y/n)? y
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .

RedBoot>
```

『fconfig ms104vga_bpp 8』を実行して、色深度を 8bpp に変更します。

```
RedBoot> fconfig ms104vga_bpp 8

RedBoot>
```

『fconfig ms104vga_res 800600』を実行して、解像度を 800×600 に変更します。

```
RedBoot> fconfig ms104vga_res 800600

RedBoot>
```

『fconfig ms104vga_output_vga true』を実行して、VGA 出力を有効にします。

```
RedBoot> fconfig ms104vga_output_vga true

RedBoot>
```

VGA 以外の出力とその他の機能を無効にします。

```
RedBoot> fconfig ms104vga_custom_pc104 false
.
RedBoot> fconfig ms104vga_output_lcd false
.
RedBoot> fconfig ms104vga_output_lcd_sim false
.
RedBoot> fconfig ms104vga_output_ntsc false
.
RedBoot> fconfig ms104vga_output_svideo false
.
RedBoot>
```

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認します。


```
RedBoot> fconfig -l -n
boot_script: false
.
kernel_para: false
ms104vga: true
ms104vga_bpp: 8
ms104vga_custom_pc104: false
ms104vga_output_lcd: false
ms104vga_output_lcd_sim: false
ms104vga_output_ntsc: false
ms104vga_output_svideo: false
ms104vga_output_vga: true
ms104vga_res: 800600
net_debug: false
RedBoot>
```

LCD、NTSC、S-Video、VGA 出力のサポート設定は同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

LCD 同時出力と LCD 出力を同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

MS104-VGA/LCD の VGA 出力に対応する RedBoot のコンフィグレーションデータの設定については「Table 2.4-2 RedBoot コンフィグレーションデータ」に一覧がありますのでご覧ください。

RedBoot の『reset』コマンドを実行し、再起動させます。

```
RedBoot> reset   
+Monitor : VGA output, 8bpp, 800 x 600  
Ethernet eth0: MAC address 00:0c:7b:20:00:11  
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0  
Default server: 0.0.0.0, DNS server IP: 0.0.0.0  
  
RedBoot(tm) bootstrap and debug environment [ROM]  
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003  
  
Platform: MS104-SH4 (SH7750R) Version x.x  
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.  
  
RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available  
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.  
RedBoot>
```

RedBoot のコンフィグレーション設定が正しければ、下記のロゴがVGA モニタに出力されます。



Fig 3.3-3 RedBoot ロゴ

『libboot』 コマンドで Linux カーネルを起動します。

```
RedBoot> libboot 
Now booting Linux kernel:
Base address 0x8c001000 Entry 0x8c210000
MOUNT_RDONLY : 0x00000001
RAMDISK_FLAGS : 0x00000000
ORIG_ROOT_DEV : 0x00000100
LOADER_TYPE   : 0x00000001
INITRD_START  : 0x00360000
INITRD_SIZE   : 0x00c00000
Cmdline : console=ttySC1,38400,N,8 mem=32M ether=8,0x1300,0,0,eth0 video=e1356fb:system:ms104sh4,
display:crt,bpp:8,800x600@60
Uncompressing Linux... Ok, booting the kernel.
.
.
```

Linux の起動を確認し、root 権限でログインします。

```
MS104SH4 login: root 
```

Linux から NFS マウントを実行します。

NFS サーバの IP アドレスは「192.168.1.201」と仮定します。

```
[MS104SH4@root]# mount -t nfs -o nolock -o rsize=2048 192.168.1.201:/nfs /mnt/nfs 
```

Microwindowsのサンプルプログラムを起動します。

```
[MS104SH4@root]# /mnt/nfs/mdemo 
```

VGA 上に以下の画面が出力されます。



Fig 3.3-4 Microwindows デモプログラム

LCD

解像度 800 × 600、色深度 8bpp 対応の Linux カーネルを起動し、Microwindowsのデモプログラムを動作させます。
 以下に MS104SH4、MS104-VGA/LCD、MS104-LVDS/DVIおよび LCD-KIT-A02 の接続を示します。

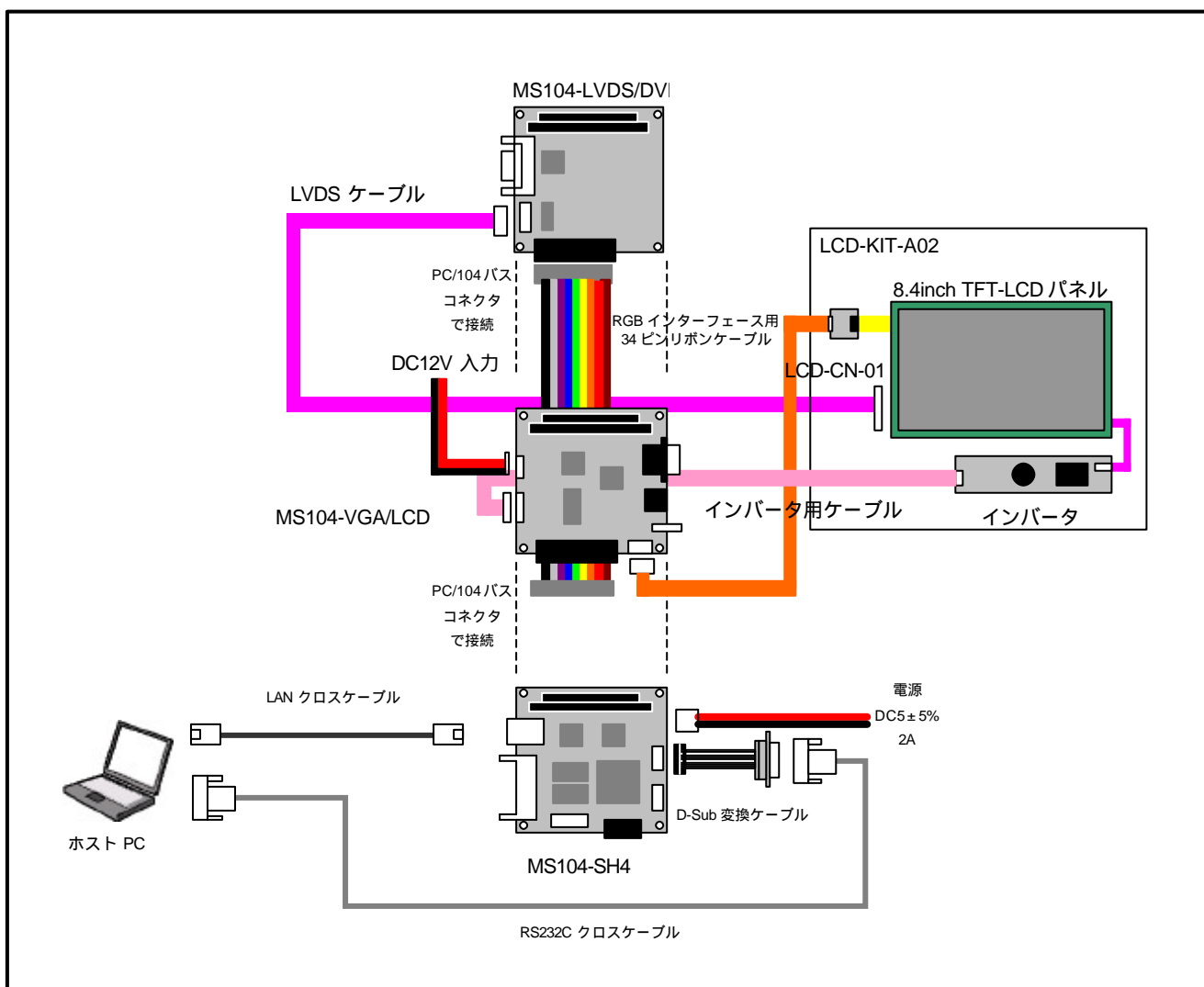


Fig 3.3-5 LCD-KIT-A02 との接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROM から Microwindows デモプログラム『mdemo』を NFS サーバディレクトリ『/nfs』にコピーします。

NFS サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。

```
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/mdemo /nfs/ 
```

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-5 LCD-KIT-A02 との接続」を参考に接続します。

MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF)と Ethernet ポートをそれぞれ、ホスト PC のシリアルポートと Ethernet ポートに接続してください。

MS104-SH4 の電源を ON し、RedBoot の起動ログが表示されることを確認します。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot の『fconfig』コマンドを使用して、MS104-VGA/LCD の VGA 出力に対応するよう RedBoot のコンフィグレーションデータを変更します。

『fconfig ms104vga true』を実行して、RedBoot の MS104-VGA/LCD サポート機能を有効にします。

```
RedBoot> fconfig ms104vga true ←入力
ms104vga: Setting to true
Update RedBoot non-volatile configuration - continue (y/n)? y ←入力
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .

RedBoot>
```

『fconfig ms104vga_bpp 8』を実行して、色深度を 8bpp に変更します。

```
RedBoot> fconfig ms104vga_bpp 8 ←入力
.

RedBoot>
```

『fconfig ms104vga_res 800600』を実行して、解像度を 800×600 に変更します。

```
RedBoot> fconfig ms104vga_res 800600 ←入力
.

RedBoot>
```

『fconfig ms104vga_output_lcd true』を実行して、LCD 出力を有効にします。

```
RedBoot> fconfig ms104vga_output_lcd true ←入力
.

RedBoot>
```


VGA 以外の出力とその他の機能を無効にします。

```
RedBoot> fconfig ms104vga_custom_pc104 false
.
RedBoot> fconfig ms104vga_output_lcd_sim false
.
RedBoot> fconfig ms104vga_output_ntsc false
.
RedBoot> fconfig ms104vga_output_svideo false
.
RedBoot> fconfig ms104vga_output_vga false
.
RedBoot>
```

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認します。

```
RedBoot> fconfig -l -n
boot_script: false
.
kernel_para: false
ms104vga: true
ms104vga_bpp: 8
ms104vga_custom_pc104: false
ms104vga_output_lcd: true
ms104vga_output_lcd_sim: false
ms104vga_output_ntsc: false
ms104vga_output_svideo: false
ms104vga_output_vga: false
ms104vga_res: 800600
net_debug: false
RedBoot>
```


← MS104-VGA/LCD サポート
← 色深度設定 (8 or 15 or 16)
← カスタム PC/104 バスの使用
← LCD 出力サポート
← LCD 同時出力サポート
← NTSC コンポジット出力サポート
← S-Video 出力サポート
← VGA 出力サポート
← 解像度設定 (800600 or 640480)

LCD、NTSC、S-Video、VGA 出力のサポート設定は同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

LCD 同時出力と LCD 出力を同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。


MS104-VGA/LCD の VGA 出力に対応する RedBoot のコンフィグレーションデータの設定については「Table 2.4-2 RedBoot コンフィグレーションデータ」に一覧がありますのでご覧ください。

RedBoot の『reset』コマンドを実行し、再起動させます。

```
RedBoot> reset   
+Monitor : LCD output, 8bpp, 800 x 600  
Ethernet eth0: MAC address 00:0c:7b:20:00:11  
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0  
Default server: 0.0.0.0, DNS server IP: 0.0.0.0  
  
RedBoot(tm) bootstrap and debug environment [ROM]  
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003  
  
Platform: MS104-SH4 (SH7750R) Version x.x  
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.  
  
RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available  
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.  
RedBoot>
```

RedBoot のコンフィグレーション設定が正しければ、LCD モニタ上に「Fig 3.3-3 RedBoot ログ」が出力されます。

『liboot』コマンドで Linux カーネルを起動します。

```
RedBoot> liboot   
Now booting Linux kernel:  
Base address 0x8c001000 Entry 0x8c210000  
MOUNT_RDONLY : 0x00000001  
RAMDISK_FLAGS : 0x00000000  
ORIG_ROOT_DEV : 0x00000100  
LOADER_TYPE : 0x00000001  
INITRD_START : 0x00360000  
INITRD_SIZE : 0x00c00000  
Cmdline : console=ttySC1,38400,N,8 mem=32M ether=8,0x1300,0,0,eth0 video=e1356fb:system:ms104sh4,  
display:tft,bpp:8,800x600@60  
Uncompressing Linux... Ok, booting the kernel.  
.  
.
```

Linux の起動を確認し、root 権限でログインします。

```
MS104SH4 login: root 
```

Linux から NFS マウントを実行します。

NFS サーバの IP アドレスは「192.168.1.201」と仮定します。

```
[MS104SH4@root]# mount -t nfs -o nolock -o rsize=2048 192.168.1.201:/nfs /mnt/nfs 
```

Microwindows のサンプルプログラムを起動します。

```
[MS104SH4@root]# /mnt/nfs/mdemo 
```

LCD モニタ上に「Fig 3.3-4 Microwindows デモプログラム」の画面が出力されます。

LCD モニタのタッチパネルにさわって、デモプログラムのウィンドウが変化することをお確かめください。

NTSC

解像度 640 × 480、色深度 8bpp 対応の Linux カーネルを起動し、Microwindows のデモプログラムを動作させます。
 以下に MS104-SH4、MS104-VGA/LCD および TV モニタの接続を示します。

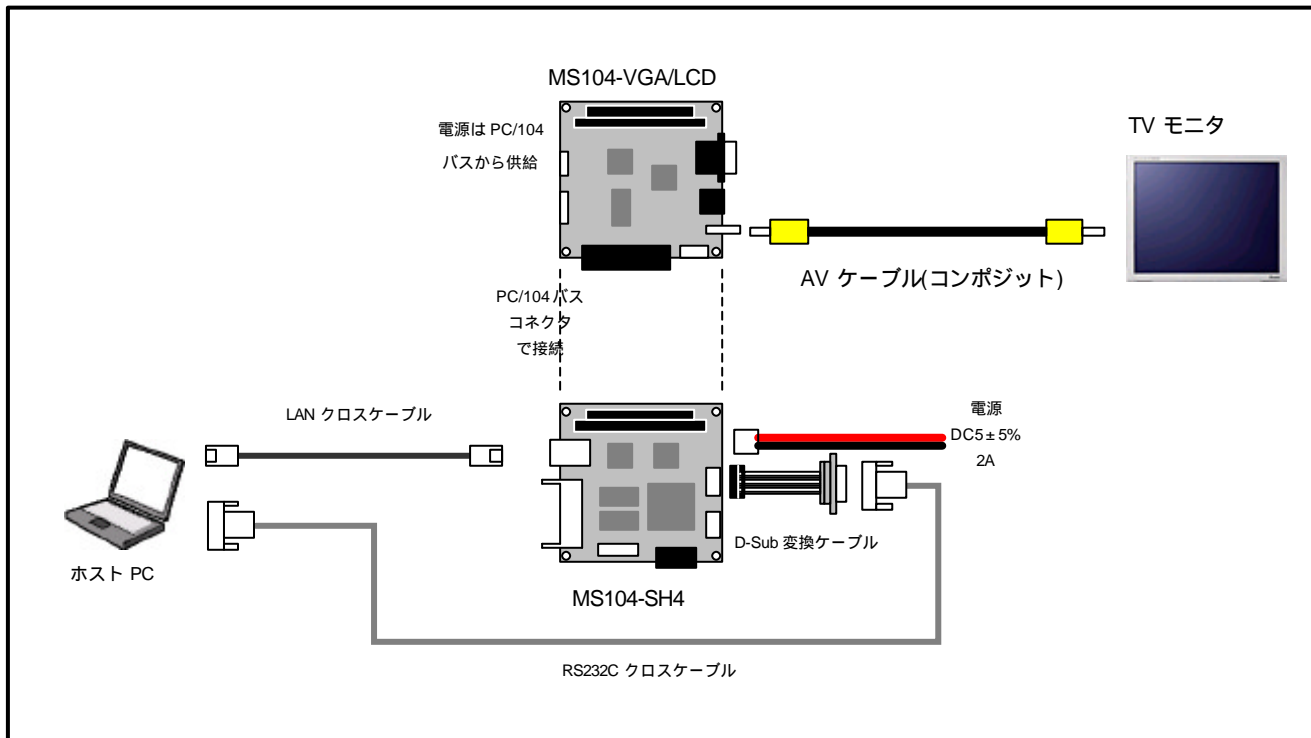


Fig 3.3-6 TV モニタとの接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROM から Microwindows デモプログラム『mdemo』を NFS サーバディレクトリ『/nfs』にコピーします。

NFS サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。

```
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/mdemo /nfs/. 
```

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-6 TV モニタとの接続」を参考に接続します。

MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF) と Ethernet ポートをそれぞれ、ホスト PC のシリアルポートと Ethernet ポートに接続してください。

MS104-SH4 の電源を ON し、RedBoot の起動ログが表示されることを確認します。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot の『fconfig』コマンドを使用して、MS104-VGA/LCD の VGA 出力に対応するよう RedBoot のコンフィグレーションデータを変更します。

『fconfig ms104vga true』を実行して、RedBoot の MS104-VGA/LCD サポート機能を有効にします。

```
RedBoot> fconfig ms104vga true ←入力
ms104vga: Setting to true
Update RedBoot non-volatile configuration - continue (y/n)? y ←入力
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .

RedBoot>
```

『fconfig ms104vga_bpp 8』を実行して、色深度を 8bpp に変更します。

```
RedBoot> fconfig ms104vga_bpp 8 ←入力
.

RedBoot>
```

『fconfig ms104vga_res 640480』を実行して、解像度を 640×480 に変更します。

```
RedBoot> fconfig ms104vga 640480 ←入力
.

RedBoot>
```

『fconfig ms104vga_output_ntsc true』を実行して、TV 出力を有効にします。

```
RedBoot> fconfig ms104vga_output_ntsc true ←入力
.

RedBoot>
```

VGA 以外の出力とその他の機能を無効にします。

```
RedBoot> fconfig ms104vga_custom_pc104 false ←入力
.
RedBoot> fconfig ms104vga_output_lcd false ←入力
.
RedBoot> fconfig ms104vga_output_lcd_sim false ←入力
.
RedBoot> fconfig ms104vga_output_svideo false ←入力
.
RedBoot> fconfig ms104vga_output_vga false ←入力
.
RedBoot>
```

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認します。

```
RedBoot> fconfig -l -n ←入力
boot_script: false
.
kernel_para: false
ms104vga: true ← MS104-VGA/LCD サポート
ms104vga_bpp: 8 ← 色深度設定 (8 or 15 or 16)
ms104vga_custom_pc104: false ← カスタム PC/104 バスの使用
ms104vga_output_lcd: false ← LCD 出力サポート
ms104vga_output_lcd_sim: false ← LCD 同時出力サポート
ms104vga_output_ntsc: true ← NTSC コンポジット出力サポート
ms104vga_output_svideo: false ← S-Video 出力サポート
ms104vga_output_vga: false ← VGA 出力サポート
ms104vga_res: 640480 ← 解像度設定 (800600 or 640480)
net_debug: false
RedBoot>
```

LCD、NTSC、S-Video、VGA 出力のサポート設定は同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

LCD 同時出力と LCD 出力を同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

MS104-VGA/LCD の VGA 出力に対応する RedBoot のコンフィグレーションデータの設定については「Table3.1.2 RedBoot コンフィグレーション設定」に一覧がありますのでご覧ください。

RedBoot の『reset』コマンドを実行し、再起動させます。

```
RedBoot> reset
+Monitor : NTSC output, 8bpp, 640 x 480
Ethernet eth0: MAC address 00:0c:7b:20:00:11
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot のコンフィグレーション設定が正しければ、TV モニタ上に「Fig 3.3-3 RedBoot ログ」が出力されます。

『liboot』コマンドでLinux カーネルを起動します。

```
RedBoot> liboot
Now booting Linux kernel:
Base address 0x8c001000 Entry 0x8c210000
MOUNT_RDONLY : 0x00000001
RAMDISK_FLAGS : 0x00000000
ORIG_ROOT_DEV : 0x00000100
LOADER_TYPE : 0x00000001
INITRD_START : 0x00360000
INITRD_SIZE : 0x00c00000
Cmdline : console=ttySC1,38400,N,8 mem=32M ether=8,0x1300,0,0,eth0 video=e1356fb:system:ms104sh4,
display:ntsc,tvfmt:0,tvfilt:7,bpp:8,.,tvfilt:7@62
Uncompressing Linux... Ok, booting the kernel.
.
.
```

Linux の起動を確認し、root 権限でログインします。

```
MS104SH4 login: root
```

Linux から NFS マウントを実行します。

NFS サーバの IP アドレスは「192.168.1.201」と仮定します。

```
[MS104SH4@root]# mount -t nfs -o nolock -o rsize=2048 192.168.1.201:/nfs /mnt/nfs
```

Microwindows のサンプルプログラムを起動します。

```
[MS104SH4@root]# /mnt/nfs/mdemo
```

TV モニタ上に「Fig 3.3-4 Microwindows デモプログラム」の画面が出力されます。

S-Video

解像度 640 × 480、色深度 8bpp 対応の Linux カーネルを起動し、Microwindowsのサンプルプログラムを動作させます。
以下に MS104-SH4、MS104-VGA/LCD および S-Video モニタの接続を示します。

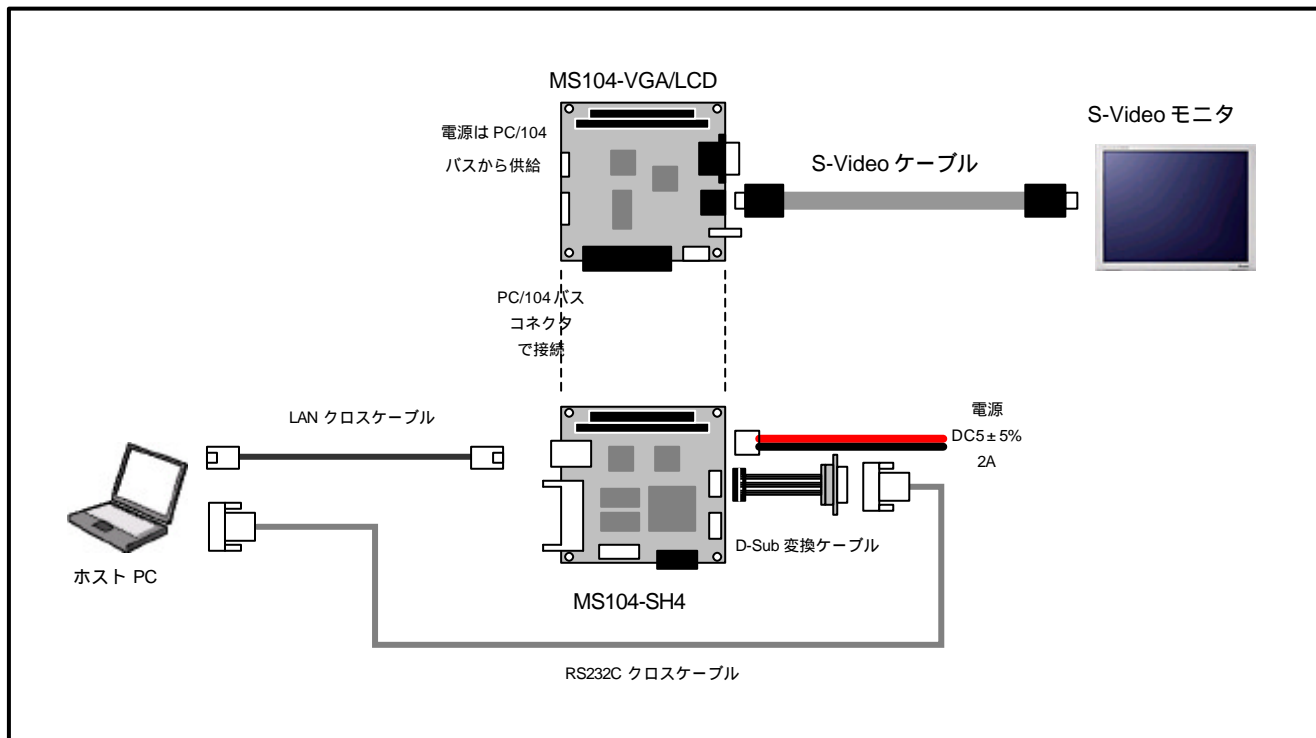


Fig 3.3-7 S-Video モニタとの接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROMから Microwindows デモプログラム『mdemo』を NFS サーバディレクトリ『/nfs』にコピーします。

NFS サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。

```
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/mdemo /nfs/. 
```

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-7 S-Video モニタとの接続」を参考に接続します。

MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF)と Ethernet ポートをそれぞれ、ホスト PC のシリアルポートと Ethernet ポートに接続してください。

MS104-SH4 の電源を ON し、RedBoot の起動ログが表示されることを確認します。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot の『fconfig』コマンドを使用して、MS104-VGA/LCD の VGA 出力に対応するよう RedBoot のコンフィグレーションデータを変更します。

『fconfig ms104vga true』を実行して、RedBoot の MS104-VGA/LCD サポート機能を有効にします。

```
RedBoot> fconfig ms104vga true
ms104vga: Setting to true
Update RedBoot non-volatile configuration - continue (y/n)? y
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .

RedBoot>
```

『fconfig ms104vga_bpp 8』を実行して、色深度を 8bpp に変更します。

```
RedBoot> fconfig ms104vga_bpp 8
.

RedBoot>
```

『fconfig ms104vga res 640480』を実行して、解像度を 640 × 480 に変更します。

```
RedBoot> fconfig ms104vga 640480
.

RedBoot>
```

『fconfig ms104vga_output_svideo true』を実行して、S-Video 出力を有効にします。

```
RedBoot> fconfig ms104vga_output_svideo true
.

RedBoot>
```


VGA 以外の出力とその他の機能を無効にします。

```
RedBoot> fconfig ms104vga_custom_pc104 false ←入力
.
RedBoot> fconfig ms104vga_output_lcd false ←入力
.
RedBoot> fconfig ms104vga_output_lcd_sim false ←入力
.
RedBoot> fconfig ms104vga_output_ntsc false ←入力
.
RedBoot> fconfig ms104vga_output_vga false ←入力
.
RedBoot>
```

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認します。


```
RedBoot> fconfig -l -n ←入力
boot_script: false
.
kernel_para: false
ms104vga: true ← MS104-VGA/LCD サポート
ms104vga_bpp: 8 ← 色深度設定 (8 or 15 or 16)
ms104vga_custom_pc104: false ← カスタム PC/104 バスの使用
ms104vga_output_lcd: false ← LCD 出力サポート
ms104vga_output_lcd_sim: false ← LCD 同時出力サポート
ms104vga_output_ntsc: false ← NTSC コンポジット出力サポート
ms104vga_output_svideo: true ← S-Video 出力サポート
ms104vga_output_vga: false ← VGA 出力サポート
ms104vga_res: 640480 ← 解像度設定 (800600 or 640480)
net_debug: false
RedBoot>
```

LCD、NTSC、S-Video、VGA 出力のサポート設定は同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

LCD 同時出力と LCD 出力を同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。


MS104-VGA/LCD の VGA 出力に対応する RedBoot のコンフィグレーションデータの設定については「Table 2.4-2 RedBoot コンフィグレーションデータ」に一覧がありますのでご覧ください。

RedBoot の『reset』コマンドを実行し、再起動させます。

```
RedBoot> reset   
+Monitor : S-Video output, 8bpp, 640 x 480  
Ethernet eth0: MAC address 00:0c:7b:20:00:11  
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0  
Default server: 0.0.0.0, DNS server IP: 0.0.0.0  
  
RedBoot(tm) bootstrap and debug environment [ROM]  
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003  
  
Platform: MS104-SH4 (SH7750R) Version x.x  
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.  
  
RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available  
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.  
RedBoot>
```

RedBoot のコンフィグレーション設定が正しければ、TV モニタ上に「Fig 3.4-3 RedBoot ログ」が出力されます。

『liboot』コマンドでLinux カーネルを起動します。

```
RedBoot> liboot   
Now booting Linux kernel:  
Base address 0x8c001000 Entry 0x8c210000  
MOUNT_RDONLY : 0x00000001  
RAMDISK_FLAGS : 0x00000000  
ORIG_ROOT_DEV : 0x00000100  
LOADER_TYPE : 0x00000001  
INITRD_START : 0x00360000  
INITRD_SIZE : 0x00c00000  
Cmdline : console=ttySC1,38400,N,8 mem=32M ether=8,0x1300,0,0,eth0 video=e1356fb:system:ms104sh4,  
display:svideo,tvfmt:0,tvflt:7,bpp:8,,tvflt:7@62  
Uncompressing Linux... Ok, booting the kernel.  
.  
.
```

Linux の起動を確認し、root 権限でログインします。

```
MS104SH4 login: root 
```

Linux から NFS マウントを実行します。

NFS サーバの IP アドレスは「192.168.1.201」と仮定します。

```
[MS104SH4@root]# mount -t nfs -o nolock -o rsize=2048 192.168.1.201:/nfs /mnt/nfs 
```

Microwindows のサンプルプログラムを起動します。

```
[MS104SH4@root]# /mnt/nfs/mdemo 
```

TV モニタ上に「Fig 3.3-4 Microwindows デモプログラム」の画面が出力されます。

VGA・LCD 同時出力

解像度 800 × 600、色深度 8bpp 対応の Linux カーネルを起動し、Microwindows のデモプログラムを動作させます。
以下に MS104-SH4、MS104-VGA/LCD、MS104-LVDS/DVI、LCD-KIT-A02 および VGA モニタの接続を示します。

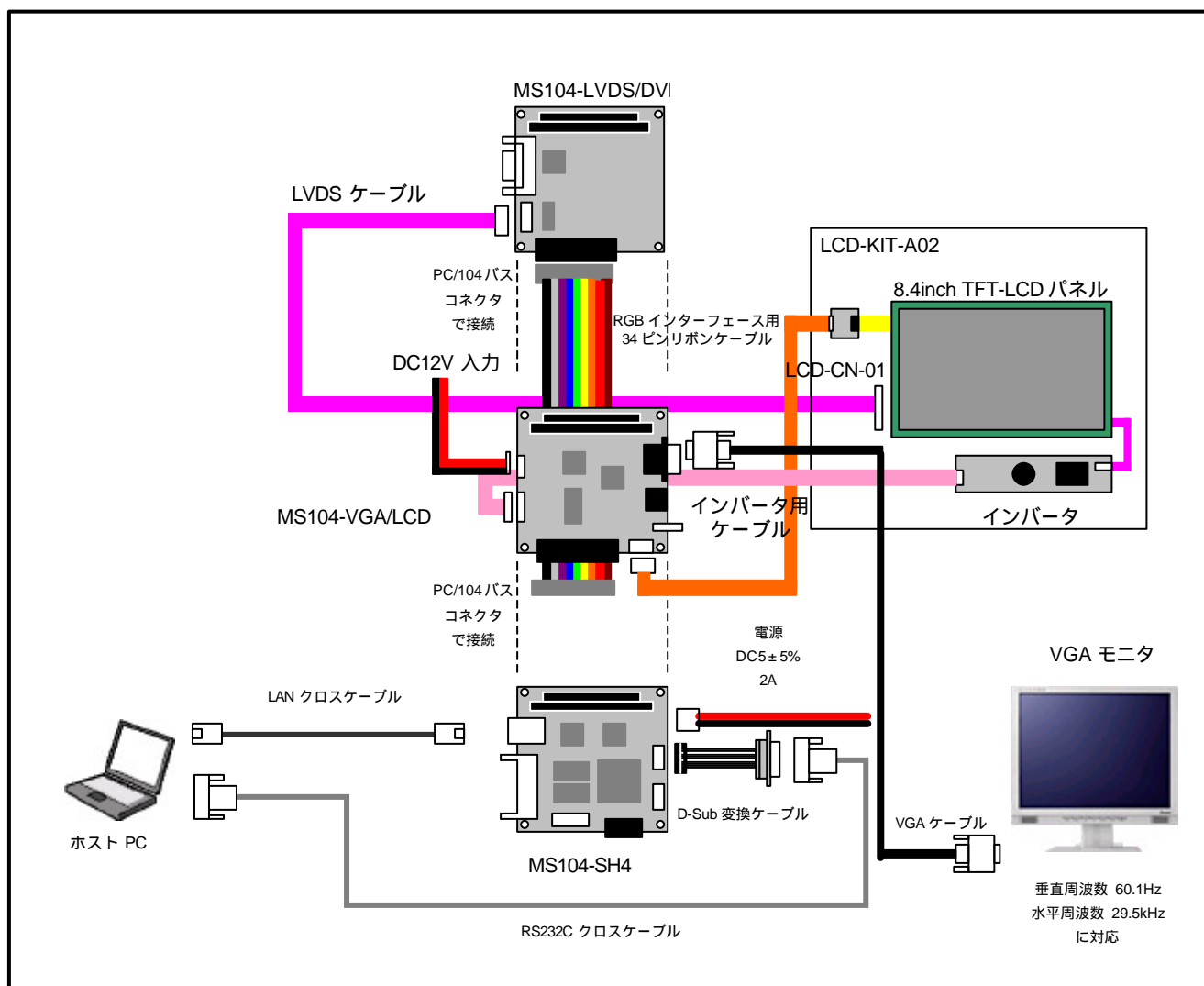


Fig 3.3-8 LCD-KIT-A02・VGA モニタとの接続

ホスト PC (LinuxPC) に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su - 
Password:*****  パスワードを入力してください
```

「MS104-SH4」の添付 CD-ROM から Microwindows デモプログラム『mdemo』を NFS サーバディレクトリ『/nfs』にコピーします。

NFS サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。

```
[alpha@ap_linux root]# cp /mnt/cdrom/linux/ms104vga/mdemo /nfs/ 
```

MS104-VGA/LCD と MS104-SH4 を「Fig 3.3-8 LCD-KIT-A02・VGA モニタとの接続」を参考に接続します。
MS104-SH4 の電源が OFF であることを確認し、MS104-SH4 ボードの COM2(SCIF)と Ethernet ポートをそれぞれ、
ホスト PC のシリアルポートと Ethernet ポートに接続してください。
MS104-SH4 の電源を ON し、RedBoot の起動ログが表示されることを確認します。

```
+Ethernet eth0: MAC address 00:0c:7b:xx:xx:xx
IP: 192.168.1.200/255.255.255.0, Gateway: 0.0.0.0
Default server: 0.0.0.0, DNS server IP: 0.0.0.0

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version v2_0 - built 20:25:22, Oct 3 2003

Platform: MS104-SH4 (SH7750R) Version x.x
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x8c000000-0x8e000000, 0x8c00f0b8-0x8dfdd000 available
FLASH: 0x80000000 - 0x81000000, 128 blocks of 0x00020000 bytes each.

RedBoot>
```

RedBoot の『fconfig』コマンドを使用して、MS104-VGA/LCD の VGA 出力に対応するよう RedBoot のコンフィグレーションデータを変更します。

『fconfig ms104vga true』を実行して、RedBoot の MS104-VGA/LCD サポート機能を有効にします。

```
RedBoot> fconfig ms104vga true
ms104vga: Setting to true
Update RedBoot non-volatile configuration - continue (y/n)? y
... Unlock from 0x80fc0000-0x80fd0000: .
... Erase from 0x80fc0000-0x80fd0000: .
... Program from 0x8dfcf000-0x8dfdf000 at 0x80fc0000: .
... Lock from 0x80fc0000-0x80fd0000: .

RedBoot>
```

『fconfig ms104vga_bpp 8』を実行して、色深度を 8bpp に変更します。

```
RedBoot> fconfig ms104vga_bpp 8

RedBoot>
```

『fconfig ms104vga_res 800600』を実行して、解像度を 800×600 に変更します。

```
RedBoot> fconfig ms104vga_res 800600

RedBoot>
```

『fconfig ms104vga_output_vga true』を実行して、VGA 出力を有効にします。

```
RedBoot> fconfig ms104vga_output_vga true

RedBoot>
```

『fconfig ms104vga_output_lcd_sim true』を実行して、LCD 同時出力を有効にします。

```
RedBoot> fconfig ms104vga_output_lcd_sim true

RedBoot>
```

VGA 以外の出力とその他の機能を無効にします。

```
RedBoot> fconfig ms104vga_custom_pc104 false
.
RedBoot> fconfig ms104vga_output_lcd false
.
RedBoot> fconfig ms104vga_output_ntsc false
.
RedBoot> fconfig ms104vga_output_svideo false
.
RedBoot>
```

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認します。

```
RedBoot> fconfig -l -n
boot_script: false
.
kernel_para: false
ms104vga: true
ms104vga_bpp: 8
ms104vga_custom_pc104: false
ms104vga_output_lcd: false
ms104vga_output_lcd_sim: true
ms104vga_output_ntsc: false
ms104vga_output_svideo: false
ms104vga_output_vga: true
ms104vga_res: 800600
net_debug: false
RedBoot>
```

←	MS104-VGA/LCD サポート
←	色深度設定 (8 or 15 or 16)
←	カスタム PC/104 バスの使用
←	LCD 出力サポート
←	LCD 同時出力サポート
←	NTSC コンポジット出力サポート
←	S-Video 出力サポート
←	VGA 出力サポート
←	解像度設定 (800600 or 640480)

LCD、NTSC、S-Video、VGA 出力のサポート設定は同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

LCD 同時出力と LCD 出力を同時に有効にすると正しく動作しない恐れがあります。必ずいずれか一つのみを有効にしてください。

MS104-VGA/LCD の VGA 出力に対応する RedBoot のコンフィグレーションデータの設定については「Table 2.4-2 RedBoot コンフィグレーションデータ」に一覧がありますのでご覧ください。

4. RedBoot

4.1 RedBoot の概要

RedBoot とは Red Hat 社により開発された組み込み用ブートローダです。RedBoot は専用のコンフィグレーションツールによって機能の追加や削除を行うことができます。この機能を利用し、LCD、VGA といった各出力先や解像度などを選択、コンパイルを行います。

各ターゲットに最適化した RedBoot はデバイスドライバ、HAL などソフトウェアやハードウェアの機能ごとに分類分けされたパッケージ（ソースコード）を組み合わせて生成します。

コンフィグレーションツールは、パッケージデータベースファイル「ecos.db」とコンポーネント記述ファイル「*.cdl」（CDL：Component Description Language）から、使用可能なパッケージと機能（コンポーネント、オプション）を確認し、コンフィグレーション項目を表示します。

コンフィグレーション項目から、ターゲットに必要な項目を設定し、「*.ecc」（ecc：eCos Configuration file）ファイルに設定値を保存し、build をおこないます。また、保存された ecc ファイルを呼び出すことにより再度コンフィグレーションをやり直すことなく再 build を行うことができます。

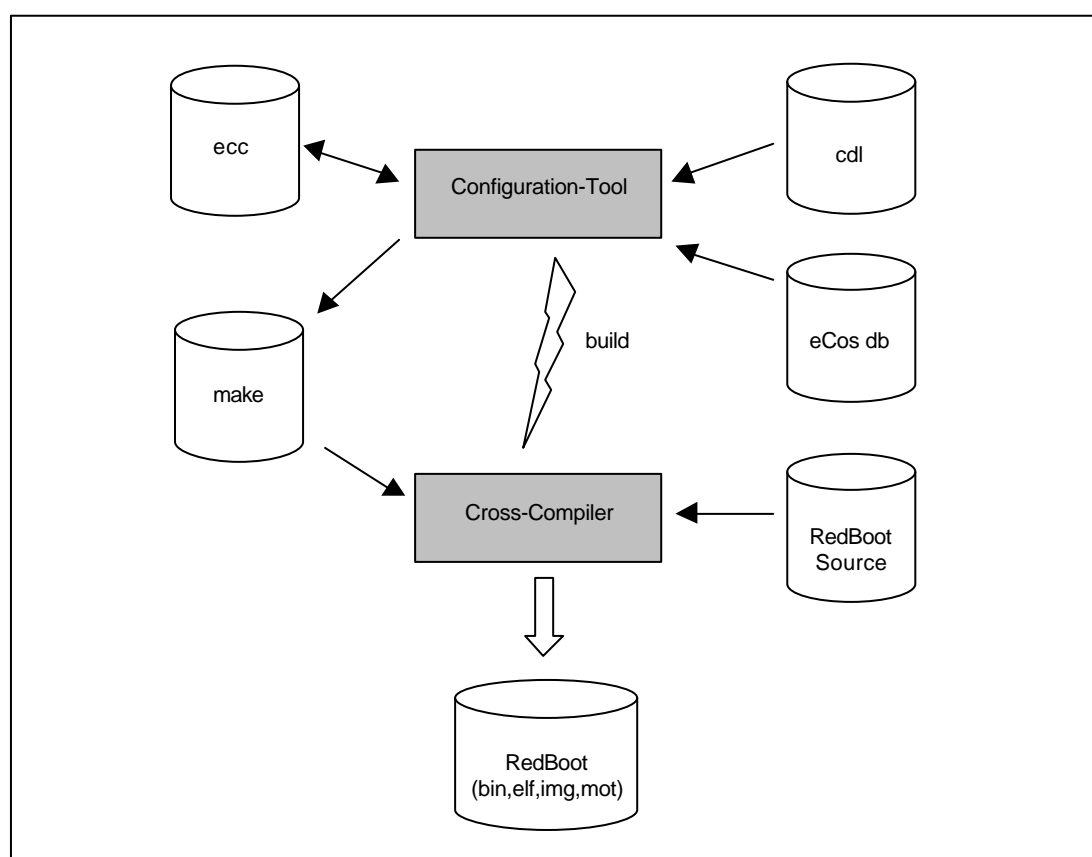


Fig 4.1-1 RedBoot の build イメージ

4.2 RedBoot のコンフィグレーション

RedBoot コンフィグレーションツールの起動

RedBoot のコンフィグレーションを行うにはコンフィグレーションツール「configtool」を使用します。コンフィグレーションツールは『/opt/alpha_project/linux-kit-a02/redboot/ecos-2.0/tools/bin』ディレクトリに格納されています。コンフィグレーションツールへの実行パスが通っている場合、「configtool」を実行し、コンフィグレーションツールを起動させます。

```
[root@ap_linux bin]# configtool
```

コンフィグレーションツールへの実行パスが通っていない場合は直接ディレクトリを指定し実行するか、「export」コマンドで実行パスを通してください。

```
[root@ap_linux root]# /opt/alpha_project/linux-kit-a02/redboot/ecos-2.0/tools/bin/configtool
```

```
[root@ap_linux root]# export PATH=$PATH:/opt/alpha_project/linux-kit-a02/redboot/ecos-2.0/tools/bin
[root@ap_linux root]# configtool
```

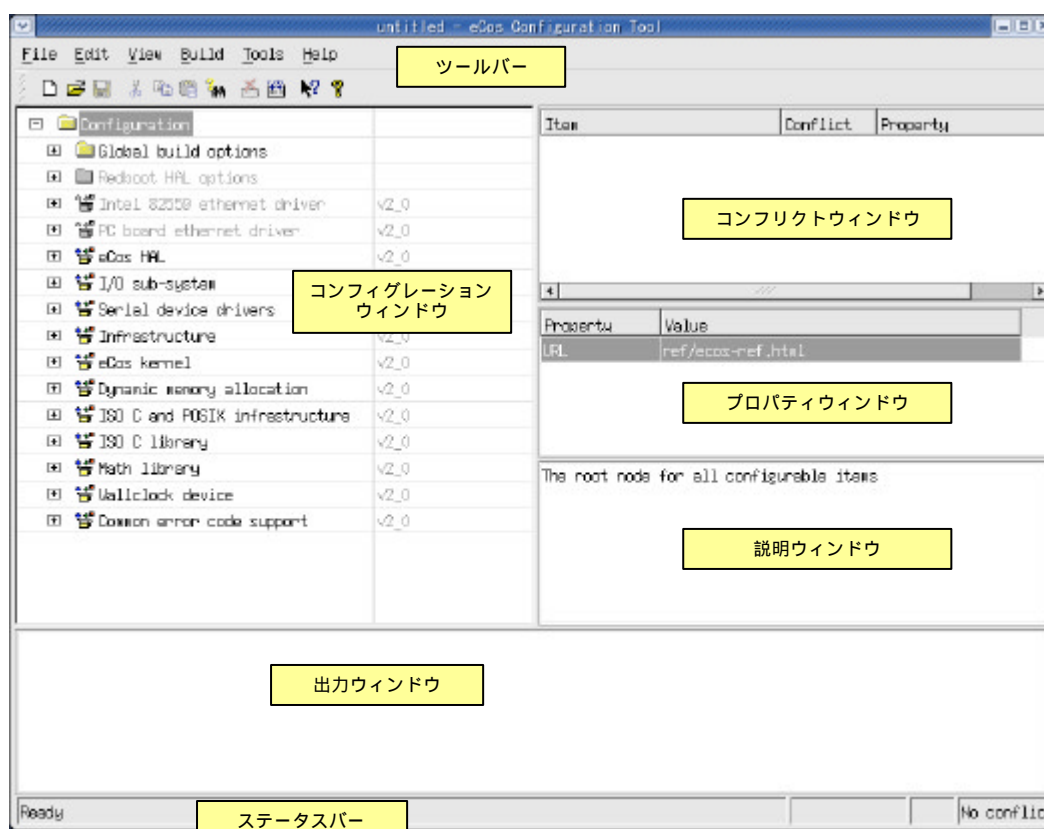


Fig 4.2-1 コンフィグレーションツールの起動画面

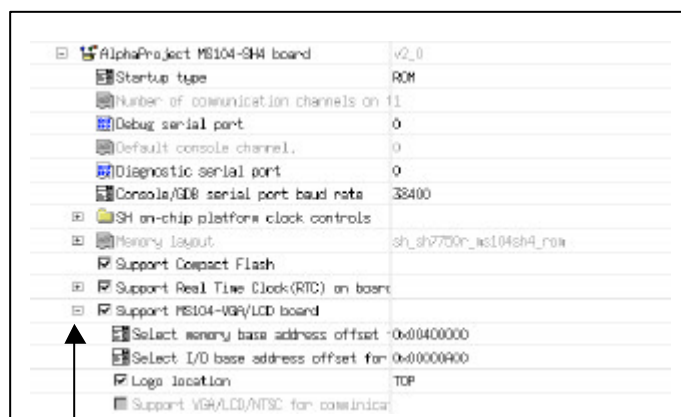
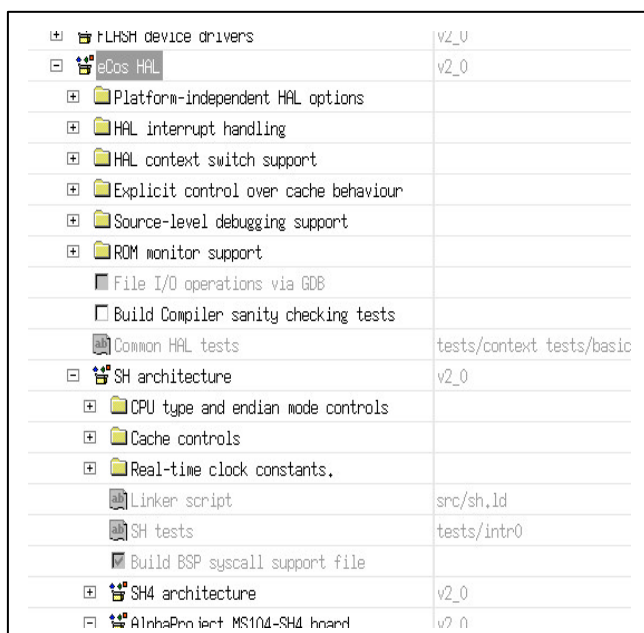
RedBoot コンフィグレーション項目について

MS104-VGA/LCD のために追加されたコンフィグレーションは以下のものがあります。

Table 4.2-2 MS104-VGA/LCD対応 RedBoot コンフィグレーション項目

設定項目	値	説明
Support MS104-VGA/LCD board	TRUE or FLASE	MS104-VGA/LCD ボードのサポート
Select memory base address offset for MS104-VGA/LCD board	0x000000 or 0x400000 or 0x800000 or 0xc00000	MS104-VGA/LCD ボードのメモリアドレスの選択 MS104-SH4 用 Linux 使用時は必ず 0x400000 を 選択してください。
Select I/O base address offset for MS104-VGA/LCD board	0x0000 ~ 0x0f00	MS104-VGA/LCD ボードの I/O アドレスの選択 MS104-SH4 用 Linux 使用時は必ず 0x0a00 を 選択してください。
Logo location	TOP or BOTTOM	ロゴの表示位置の選択
upport VGA/LCD/NTSC for communication channel	TRUE or FALSE	コンソールとして使用するか選択 現在サポートされておりません。

configtool の画面から MS104-VGA/LCD のコンフィグレーションは[eCos HAL] [SH architecture] [Alpha Project MS104-SH4 board] [Support MS104-VGA/LCD board]から行います。



MS104-VGA/LCD ボードのサポート

Fig 4.2-3 MS104-VGA/LCD対応 RedBoot コンフィグレーション

4.4 RedBoot の設定

RedBoot はフラッシュロム上に解像度、色深度、出力先などの設定データを格納しています。これらの設定データ（コンフィグレーションデータ）を変更することにより、MS104-VGA/LCD の解像度、色深度、出力先を変更することができます。フラッシュロムに格納されているコンフィグレーションデータの確認・変更は『fconfig』コマンドを用いて行います。各モニタへの出力方法については「3.3 MS104-VGA/LCD の動作」をご覧ください。

Table 4.4-1 MS104-VGA/LCD 関連 RedBoot コンフィグレーションデータ

コンフィグレーション	ニックネーム	デフォルト値	説明
グラフィックボード	ms104vga	false	グラフィックボード（MS104-VGA/LCD）を使用するか選択します。
色深度設定	ms104vga_bpp	8	グラフィックボードの色深度を指定します。（8 or 15 or 16）
カスタム PC/104 バス	ms104vga_custom_pc104	false	カスタム PC/104 バスを使用するか指定します。
LCD 出力	ms104vga_output_lcd	false	LCD 出力を指定します。
LCD 同時出力	ms104vga_output_lcd_sim	false	LCD 同時出力を指定します。
NTSC コンポジット出力	ms104vga_output_ntsc	false	NTSC コンポジット出力を指定します。
S-Video 出力	ms104vga_output_svideo	false	S-Video 出力を指定します。
VGA 出力	ms104vga_output_vga	false	VGA 出力を指定します。
解像度	ms104vga_res	800600	解像度を指定します。

RedBoot の MS104-VGA/LCD サポート機能

『fconfig -l -n』で MS104-VGA/LCD サポート機能の設定を確認することができます。

```

RedBoot> fconfig -l -n
boot_script: false
.
kernel_para: false
ms104vga: true      ← MS104-VGA/LCD サポート
ms104vga_bpp: 8    ← 色深度設定（8 or 15 or 16）
ms104vga_custom_pc104: false ← カスタム PC/104 バスの使用
ms104vga_output_lcd: false ← LCD 出力サポート
ms104vga_output_lcd_sim: false ← LCD 同時出力サポート
ms104vga_output_ntsc: false ← NTSC コンポジット出力サポート
ms104vga_output_svideo: false ← S-Video 出力サポート
ms104vga_output_vga: true ← VGA 出力サポート
ms104vga_res: 800600 ← 解像度設定（800600 or 640480）
net_debug: false
RedBoot>

```

5. Linux

5.1 Linux カーネルの概要

MS104-VGA/LCD に対応するために Linux デバイスドライバをフレームバッファ、タッチパネル、プザーデバイスドライバを追加する必要があります。

デバイスドライバの追加は X-Window もしくはテキストベースのコンフィグレータにより行います。また、Linux のコンフィグレータでは以前行ったコンフィグレーションを記録することができます。記録したコンフィグレーションファイル呼び出すことにより、再度コンフィグレーションをやり直すことができます。

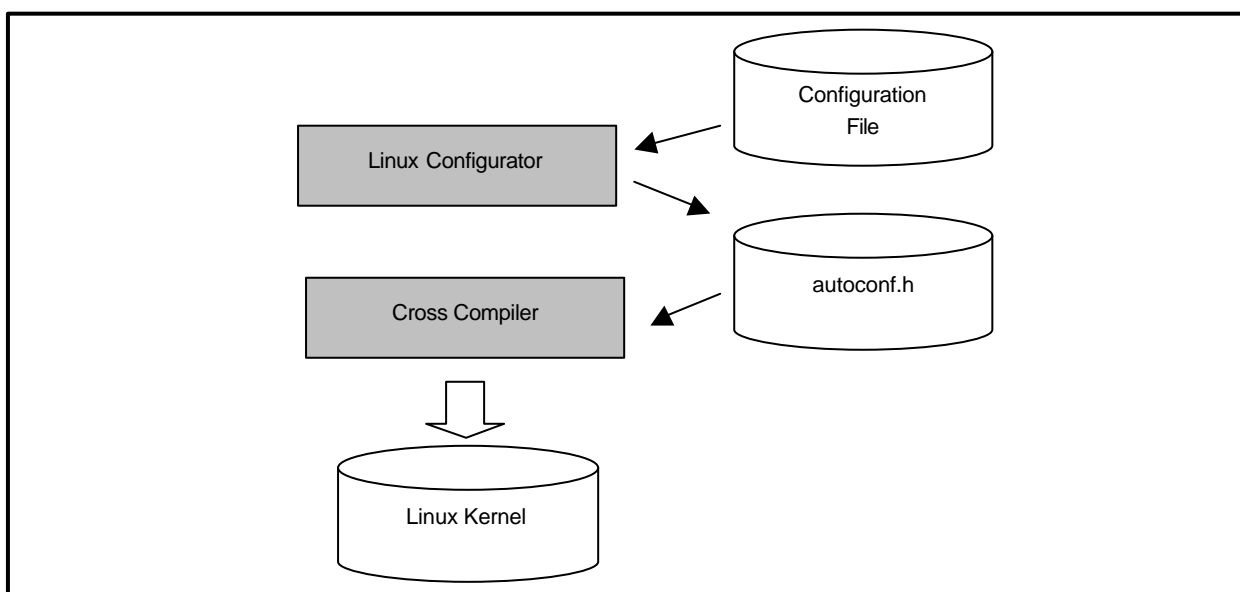


Fig 5.1-1 Linux の build イメージ

5.2 Linux カーネルのコンフィグレーション

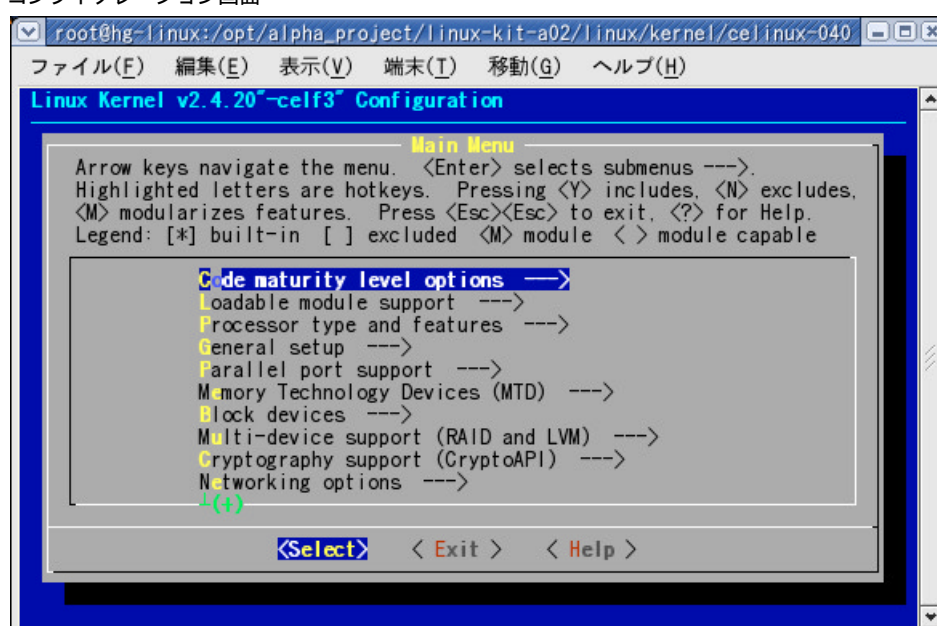
Linux コンフィグレータの起動

Linux カーネルは X-Window もしくはテキストベースのコンフィグレータにより、コンフィグレーションを行うことができます。

MS104-SH4 用 Linux のコンフィグレータを起動するには、『/opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp』ディレクトリに移動し、『make menuconfig』を実行してください。

```
[root@ap_linux root]# cd /opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp
[root@ap_linux celinux-040503-alp]# make menuconfig
```

Linux コンフィグレーション画面



Linux コンフィグレーション項目について

MS104-SH4 用 Linux カーネルのコンフィグレーションを下記に示します。

『/opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp/arch/sh/def-configs/ms104-sh4/ms104vga.config』

ディレクトリ下にコンフィグレーションファイルが保存されています。

Table 5.2-1 MS104-VGA/LCD用 Linux コンフィグレーション

設定項目	設定	説明
Character devices		シリアルポート、マウス等キャラクタ型デバイスに関するカテゴリ
Virtual terminal	* 1	ディスプレイ、キーボードデバイスをターミナルとしてサポート
Mice	-	マウスデバイスに関する設定
Mouse Support (not serial and bus mice)	*	マウスのサポート
MK712 touch screen support	*	タッチパネルのサポート
Buzzer Support	*	ブザーのサポート
Epson S1D13506/13806 framebuffer support	*	EPSON グラフィックコントローラのサポート
Use CRT display	*	VGA モニタ出力
Use Compisite NTSC (TV) display	*	TV モニタ出力
Use TFT Panel display	*	TFT 型 LCD モニタ出力
Advanced low level driver options	*	フレームバッファデバイスのハードウェアに関する設定
8 bpp packed pixels support	*	8 ビット色深度のサポート
16 bpp packed pixels support	*	16 ビット色深度のサポート
Select compiled-in fonts	*	フォントの組み込みのサポート
VGA 8x8 font	*	8x8 フォントサポート
VGA 8x16 font	*	8x16 フォントサポート

1 「*」はコンフィグレーションで選択した機能を Linux カーネルに静的に組み込むことを示しています。

6 . Microwindows

6.1 Microwindows の概要

Microwindowsは Century Software 社のオープンソースのプロジェクトで、組み込み機器向けに設計されたウィンドウシステムです。Microwindows は3階層で設計されており、最下層はモニターやマウス、キーボードなどの入出力デバイスのドライバです。中間層は線描、領域塗りつぶし、クリッピング、カラーモデルなどのグラフィックエンジン、最上位層は2つのAPIをサポートしています。

API は Microwindows と呼ばれる Win32 API を模した API と Nano-X と呼ばれる X Window System の API を模した API の2種類です。Nano-X は Linux で採用されています。

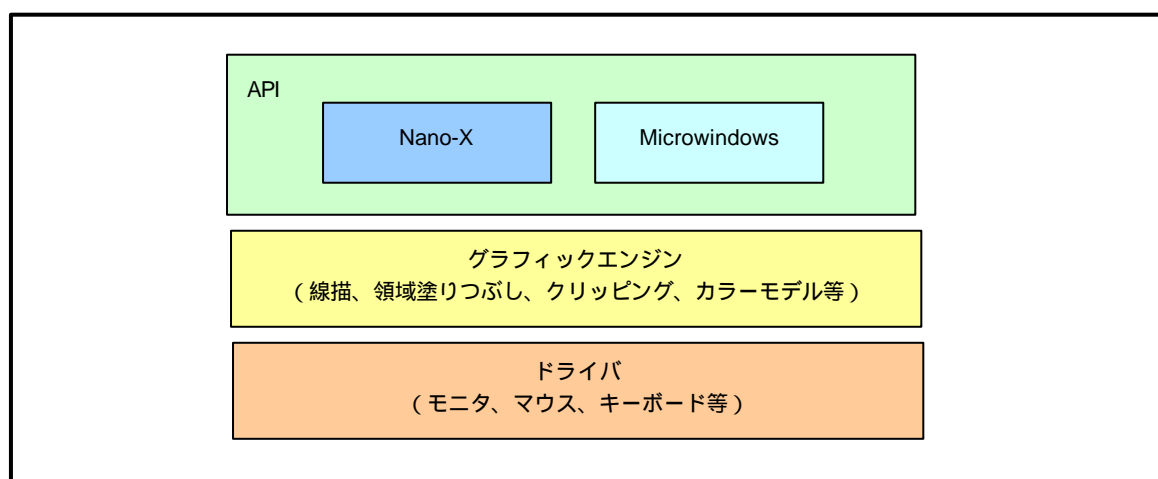


Fig 6.1-1 Microwindows アーキテクチャ

Nano-X は X Window System と同様にクライアント・サーバ型をとっています。そのため、アプリケーションプログラムを動作させる場合、Nano-Xサーバを立ち上げ、アプリケーションプログラムとして Nano-X クライアントを立ち上げます。

Nano-Xサーバはキーボードやモニター、マウスなどの入出力デバイスの処理を行い、クライアントは各アプリケーションの処理、つまり、ウィンドウの表示をするプログラムの処理を担当します。

Nano-X ではモニター上でウィンドウがどのように見えるかや、マウスがウィンドウをどのように動かすかなど、見栄えや操作方法については定義されていません。そのため、ウィンドウマネージャと呼ばれるアプリケーションプログラムが見栄えや操作方法を定義、管理します。Nano-Xには標準のウィンドウマネージャとして「nanowm」が添付されています。

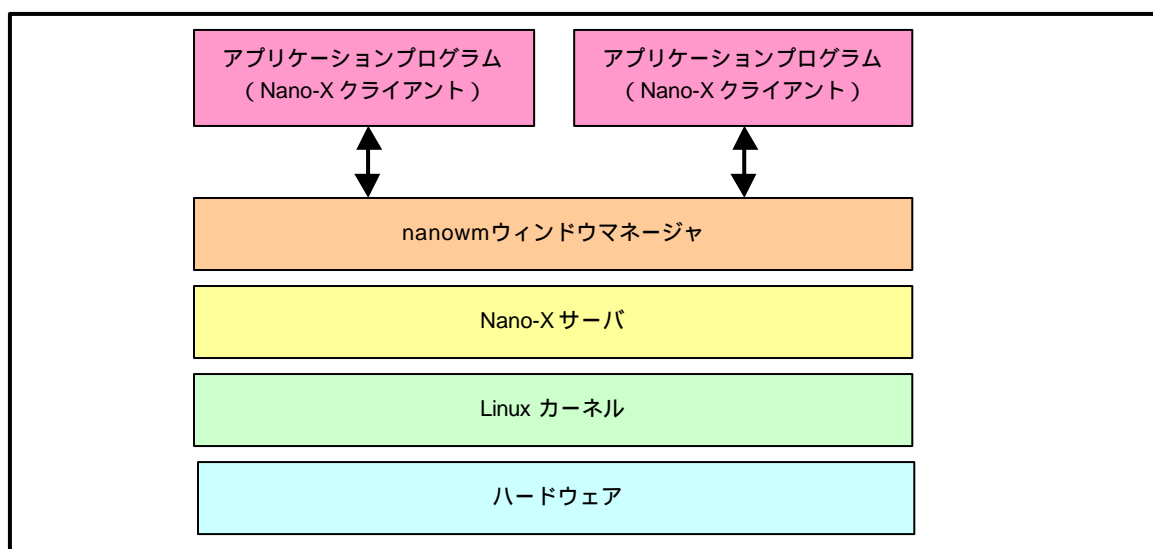


Fig 6.1-2 Microwindowsシステム構成

6.2 Microwindows の開発について

MicrowindowsはLinuxPC上でも動作するため、ある程度のデバッグをLinuxPC上で行うことができます。そのため、LinuxPC上でGUIの開発をある程度行い、動作を実機(MS104-SH4)上で確認することにより開発効率を上げることができます。以下にソースファイルのコンパイルから動作までの一連の流れを示します。

- A) ソースファイルを作成します。
- B) PC用コンパイラ『gcc』でコンパイルを実行し、実行ファイルを作成します。
- C) PC上で動作を確認します。
- D) SH4用クロスコンパイラ『sh4-linux-gcc』でコンパイルを実行し、SH4用実行ファイルを作成します。
- E) MS104-SH4ボード上で動作を確認します。

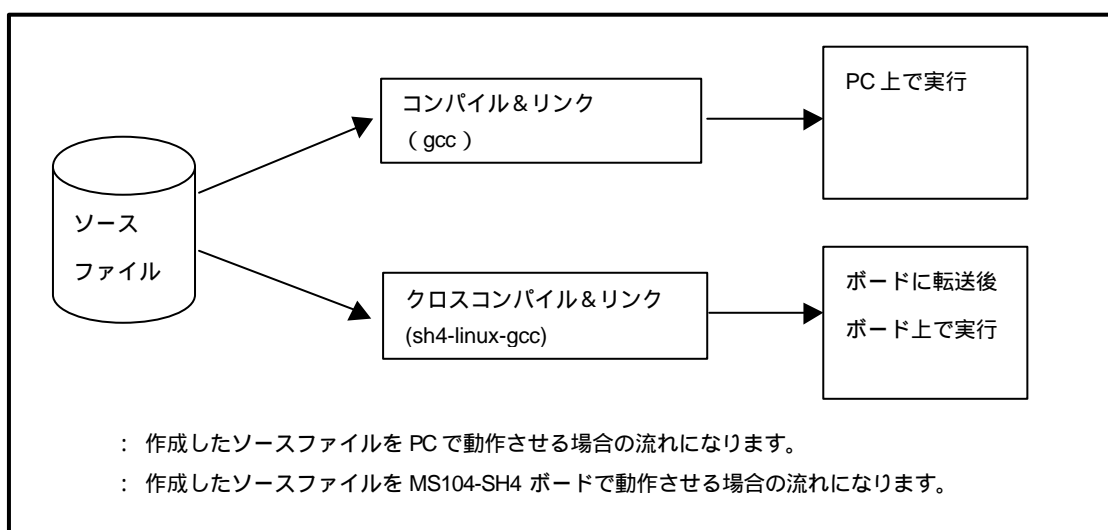


Fig 6.2-1 コンパイルの流れ

6.3 Microwindows のコンフィグレーション

Microwindowsはコンパイル時に、サポートデバイスや解像度・色深度を設定します。コンパイル後、これらの設定は変更することができません。

Microwindows コンフィグレーションファイル

Microwindowsの設定はコンフィグレーションファイルで行います。コンフィグレーションファイルは Microwindows ソースファイルディレクトリ『/opt/alpha_project/linux-kit-a02/linux/microwindows/microwindows-0.90-ms104』の『src/config』ファイルになります。

Microwindowsソースファイルディレクトリ下『src/Configs』ディレクトリには各プラットフォームに対応するコンフィグレーションファイルが格納されています。

MS104-SH4 用に 4 種類のコンフィグレーションファイルを用意しています。

Table 6.3-1 MS104-SH4 用 Microwindows コンフィグレーションファイル

サポートデバイス	MS104 シリーズ	色深度	コンフィグレーションファイル名
タッチパネル	MS104-SH4 MS104-VGA/LCD	8	config.ms104sh4_bpp8_vga
	MS104-LVDS/DVI LCD-KIT-A01	16	config.ms104sh4_bpp16_vga
USB キーボード USB マウス	MS104-SH4 MS104-VGA/LCD	8	config.ms104sh4_bpp8_usb
	MS104-USB H/S	16	config.ms104sh4_bpp16_usb

また、LinuxPC 用に 4 種類のコンフィグレーションファイルを用意しています。

Table 6.3-2 PC 用 Microwindows コンフィグレーションファイル

解像度	色深度	コンフィグレーションファイル名
640 × 480	8	config.ms104sh4_x11_bpp8_640x480
	16	config.ms104sh4_x11_bpp16_640x480
800 × 600	8	config.ms104sh4_x11_bpp8_800x600
	16	config.ms104sh4_x11_bpp16_800x600

6.6 タッチパネルのキャリブレーション

タッチパネル付 LCD モニタを使用する場合、タッチパネルで得られる値と LCD モニタの位置を関連付けなければなりません。タッチパネルで得られる値はアナログ値で、経年変化による劣化や使用環境による値の増減を考慮する必要があります。そのため、タッチパネル付 LCD モニタを使用する際はタッチパネルの位置合わせ（キャリブレーション）をする必要があります。MS104-VGA/LCD を弊社 LCD キット「LCD-KIT-A02」と接続した際に Microwindows 上からキャリブレーションを行う Nano-X クライアントアプリケーションプログラム『nxc calibration』が RAM ディスク（ルートファイルシステム）に含まれています。以下ではタッチパネルキャリブレーションとキャリブレーションの実行について説明します。

タッチパネルは 12 ビットの分解能、つまり、0~4095 までの値でタッチパネルのどの座標を押したかを示します。解像度 800×600 の LCD モニタを使用した場合、LCD モニタの (800、600) という座標はタッチパネルでは (4095、4095) という座標になります。実際には LCD モニタとタッチパネルの Y 座標の値は最大値と最小値が逆になるため、(800、600) は (4095、0) という値で表されます。

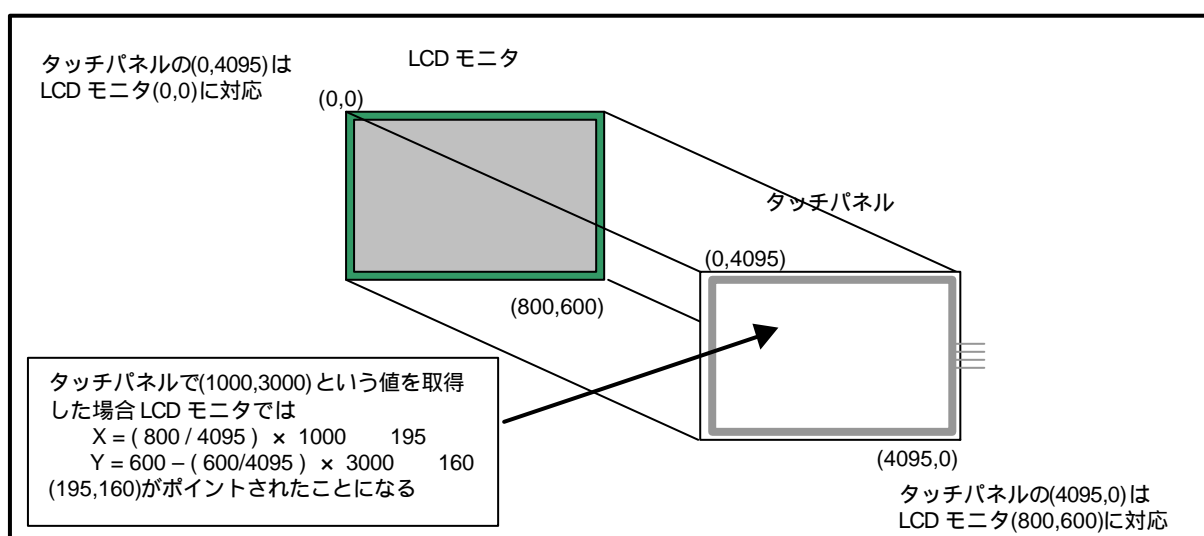


Fig 6.6-1 タッチパネルと LCD モニタの座標

しかし、上記は理想的な場合で実際には LCD モニタの (0、0) や (800、600) という座標がタッチパネルの (0、4095) や (4095、0) といった値にはなりません。タッチパネルの 0~4095 という値はあくまで分解能であり、LCD モニタの 0~800 という座標がそのまま 0~4095 に対応しません。したがって、LCD モニタの (0、0) や (800、600) という座標に対して、タッチパネルはどのような値をとるか測定し、その測定値を元に LCD モニタとタッチパネルの座標の変換を行う必要があります。

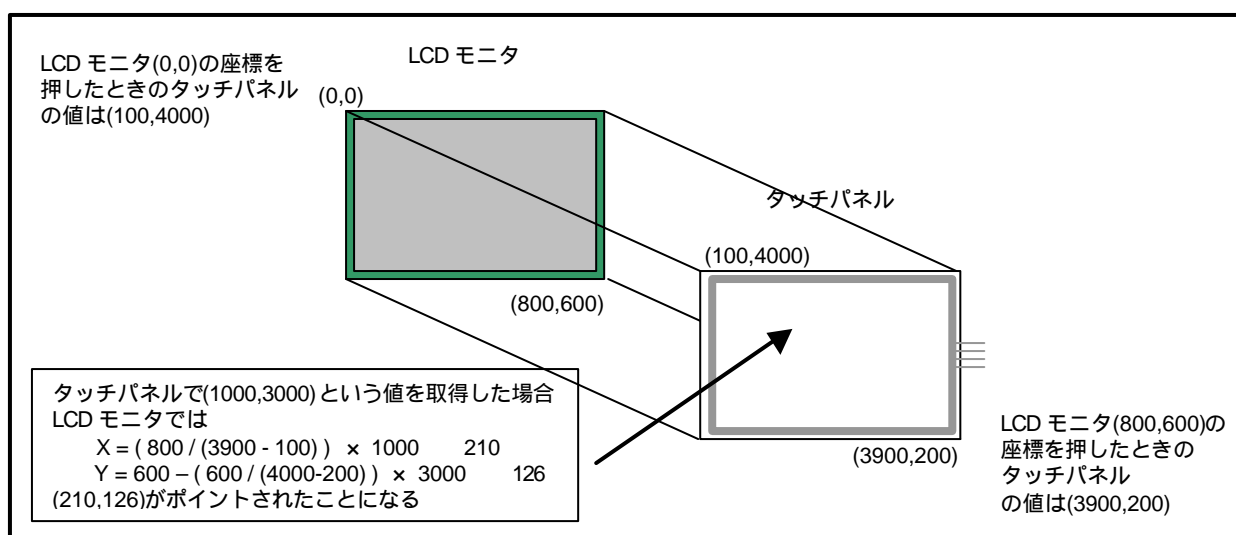


Fig 6.6-2 タッチパネルのキャリブレーション

7. 保証とサポート

弊社では最低限の動作確認をしておりますが、Linux および付属ソフトウェアの性能や動作を保証するものではありません。また、これらのソフトウェアについての個別のお問い合わせ及び技術的な質問は一切受け付けておりませんのでご了承ください。

個別サポートをご希望されるお客様には、別途有償サポートプログラムをご用意しておりますので、弊社営業までご連絡ください。

Linux など、付属する GPL ソフトウェアのソースコードは弊社ホームページより全てダウンロードすることができます。また、これらのソフトウェアは不定期にバージョンアップをおこない、ホームページ上で公開する予定です。

謝辞

Linux、SH-Linux、eCos/Redboot、Microwindows の開発に関わった多くの貢献者に深い敬意と感謝の意を示します。

著作権について

- ・ 本文書の著作権は（株）アルファプロジェクトが保有します。
- ・ 本文書の内容を無断で転載することは一切禁止します。
- ・ 本文書の内容は、将来予告なしに変更されることがあります。
- ・ 本文書の内容については、万全を期して作成いたしました。万が一不審な点、誤りなどお気付きの点がありましたら弊社までご連絡下さい。
- ・ 本文書の内容に基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社では一切責任を負いませんのでご了承下さい。

商標について

- ・ SuperH は、（株）ルネサステクノロジーの登録商標、商標または商品名称です。
- ・ Linux は、Linus Torvalds の米国およびその他の国における登録商標または商標です。
- ・ Microwindows は Century Software の登録商標、商標または商品名称です。
- ・ eCos™ および RedBoot™ は RedHat™ 社の商標です。
- ・ Windows® の正式名称は Microsoft® Windows® Operating System です。
Microsoft、Windows、Windows NT は、米国 Microsoft Corporation. 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。
Windows® XP、Windows® 2000 Professional、Windows® Millennium Edition、Windows® 98 は、米国 Microsoft Corporation. の商品名称です。
本文書では下記のように省略して記載している場合がございます。ご了承ください。
Windows® XP は Windows XP もしくは WinXP
Windows® 2000 Professional は Windows 2000 もしくは Win2000
Windows® Millennium Edition は Windows Me もしくは WinMe
Windows® 98 は Windows 98 もしくは Win98
- ・ その他の会社名、製品名は、各社の登録商標または商標です。



株式会社アルファプロジェクト
〒431-3114
静岡県浜松市東区積志町 834
<http://www.apnet.co.jp>
E-MAIL : sales@apnet.co.jp