

Open Source Conference 観光ガイド

オープンソースカンファレンス 東京春

NetBSD

高橋 徹(たかはし とおる)

1941年1月宇都宮市に生れ、45年に秋田県大曲市に疎開。宇都宮と大曲とで育つ。59年東北大学文学部入学。60年安保闘争を経験。64年美学科卒業。大阪で高校教師をつとめ、65年「オール関西」編集部勤務。67年に結婚ののち「日本読書新聞」に転じ上京。68年、冬樹社編集部入社、同人誌「艸」に参加。76年、泰流社入社。80年、現代企画室勤務となる。

論文：「矛盾を生きたイデオログ——橋樑」「共同体論序説」「政治的党派性の揚棄——吉本隆明と『政治と文学』論争」「アジア革命と民衆の暗部」「参謀の在」など。初期には熱田利明のペンネームを用いた。



日本 NetBSD ユーザーグループ

Japan NetBSD Users' Group

2023

NetBSD について

NetBSD とは？

NetBSD は、国際的なコミュニティによって開発された、完全に無料でオープンソースの UNIX ライクなオペレーティング システムです。これは「ディストリビューション」でもバリエーションでもありませんが、数十年にわたって進化し、BSD ファミリーの中で完全かつ独自のオペレーティング システムになりました。

NetBSD が最初にリリースされたのは 1993 年です。そのコードは、品質、清潔さ、および安定性の長い歴史に基づいて、時間の経過とともに多くの驚くべき環境に浸透してきました。NetBSD コードは、もともとカリフォルニア大学バークレー校の 4.4BSD Lite2 から派生したものです。

NetBSD は、完全に再現可能なバイナリーのセットとして配布されています:

- **リリース** は、一定期間のテストの後、安定したブランチから定期的に削除され、数年間サポートされます。
- **NetBSD-stable** は、最新のリリース ブランチのナイトリー ディストリビューションであり、次のポイントリリースに含まれる修正と改善が含まれています。同じブランチからのリリースのバイナリと互換性があります。
- **NetBSD-current** は、最新の開発ブランチの夜間配布であり、最新の機能が含まれていますが、実験的な変更やバグが含まれている可能性もあります。現在、-current 用の公式パッケージビルドは作成されていません。

なぜ NetBSD を使うのですか？

NetBSD ユーザーは、シンプルで十分に文書化された、完全に統合された UNIX ライクなシステムを楽しんでいます。このシステムは最小限で、多くの点で伝統的でありながら、多くの現代的で興味深い機能と最近のハードウェアのサポートを含んでいます。

コミュニティとして、NetBSD を作成する人々は幅広い関心を持っており、その結果、いくつかの多様な機能を備えたシステムが生まれました:

- セキュリティおよびメモリ強化機能- デフォルトでグローバルに適用される PaX MPROTECT (W^X) を含み、バイナリを除外するオプションなどがあります。ファイルの整合性保護は **verifexec** によって提供され、従来の BSD は、スーパーユーザーでさえ実行できる操作がさらに制限されます。NetBSD には独自のネイティブ ファイアウォールである **NPF** が含まれており、セキュリティ クリティカルなネットワーク デバイスで正常に使用されています。NetBSD のカーネルとユーザー空間は、コード サニタイザーと自動テストによる広範なチェックを受けています。
- 強力なパッケージ管理- NetBSD の **pkgsrc** には、四半期ごとの安定版ブランチと「ローリング リリース」ブランチの独自のリリース スケジュールがあり、NetBSD ベース システムと任意の方法で組み合わせることができます。 **pkgin** は、pkgsrc 用のユーザー フレンドリーなバイナリ パッケージ マネージャーですが、pkgsrc 自体を使用すると、パワー ユーザーは非常に柔軟に操作できます。 **pkgsrc** は、NASA を含む高性能科学計算コミュニティで広く採用されており、他のプラットフォームをサポートしていますが、NetBSD が優先されます。
- 最新のストレージ機能- **ZFS** ファイル システム、 **RAIDframe** ソフトウェア RAID システム、 **cgd** ディスク暗号化など。 **Logical Volume Manager** だけでなく、従来の BSD ファイルシステム (ロギング拡張機能付き) およびディスクラベル システムもサポートされて
- ARM ハードウェアは、強力な SBBA/SBBR サーバー、オープン ハードウェア ラップトップ、ポケットサイズの開発ボードなど、オープンで低コストのハイエンド デバイスを幅広くサポートし完全にメインライン カーネルにあり、単一のイメージでサポートされ、長期サポートを念頭に置いて NetBSD 開発者によって維持されます。
- 仮想化のサポート- **Xen** で確立されたエンタープライズ ソリューション、およびネイティブの NetBSD カーネル モジュールとライブラリを含み、 **NVMM** ハイパーバイザーを構成します。これにより、クリーンで安全な方法で **QEMU** のハードウェア アクセラレーションが提供されます
- NVMe、UEFI、高速化されたグラフィックス、さまざまな **ラップトップ** など、最新の x86 ハードウェアのサポート。
- 幅広い「レガシー」ハードウェアと ABI の安定したサポートを継続します。64 ビット時間などの機能に妥協することなく、初期の NetBSD リリースに対しても長期的な後方互換性があります。これらのシステムは、2038 年以降も稼働し続ける予定です。

NetBSD プロジェクトの目標

目標がなければ、プロジェクトは意味がありません。ありがたいことに、NetBSD プロジェクトには、かなり長い間忙しくしておくだけの十分な目標があります。一般的に言えば、NetBSD プロジェクトは:

- よく設計された、安定した、高速な BSD システムを提供します。
- ライセンスの煩わしさを回避し、
- 多くのハードウェア プラットフォームで動作するポータブル システムを提供します。
- 他のシステムとの相互運用性が高く、
- 可能な限りオープン システム標準に準拠します。

要約: NetBSD プロジェクトは、専門家、愛好家、研究者が好きなように使用できる、無料で再配布可能なシステムを提供します。

なぜ名前？

NetBSD は、ネットワーク接続されたバージョン管理システムを使用して OS を開発し、1993 年以来電子メールでプロジェクトを組織することで、完全にインターネット上で共同で組織された最初の主要なオープンソース プロジェクトの 1 つでした。インターネットは、NetBSD を可能にした実現技術でした。このように、私たちの名前の「ネット」は、インターネットへのオマージュとして選ばれました。

私たちの名前の「BSD」は、4.4BSDおよび386BSDの派生物としての私たちの伝統を明確に認識しています。

[NetBSD の歴史](#)についてもっと読んでください。

NetBSD を実現する人々

多くの人々が、システム自体の開発、開発のサポート、または単に使用することによって、NetBSD オペレーティング システムを今日の状態にするために多くの時間と労力を費やしてきました。これらの人々は、次のグループに分類できます。

- [NetBSD 開発者](#)
- [NetBSD コアグループ](#)
- [The NetBSD Foundation, Inc.](#)
- [NetBSD ポートメンテナー](#)
- [NetBSD プロジェクト内の他のグループ](#)
- [NetBSD に寄付または融資を行った組織および人々](#)

さらに、カリフォルニア大学バークレー校の [Computer Systems Research Group](#) と [Berkeley Software Distributions](#) への多くの貢献者がいなければ、NetBSD プロジェクトは確実に存在しませんでした。彼らの努力に感謝します。

[NetBSD Foundation](#) は、NetBSD プロジェクトの伝統的な目標と精神に専念する免税法人（米国内国歳入法のセクション 501(c)(3) に基づく）として米国で法人化されており、次の商標を所有しています。「NetBSD」という言葉。

目次:

- I. クイックスタート
 1. バイナリパッケージをインストール (NetBSD)
 2. バイナリ パッケージのインストール (illumos, macOS, Enterprise Linux)
 3. ソースパッケージをインストールする
- II. コミュニティとサポート
- III. 安全
- IV. 問題の解決
- V. pkgsrc が実行されるプラットフォーム
 1. 主な焦点
 2. その他のサポートされているプラットフォーム
- VI. 開発者向け情報
- VII. その他のリンク

クイックスタート

バイナリパッケージをインストール (NetBSD)

を設定し、まだインストールされていない場合はPKG_PATHインストールします。pkgin

```
# PKG_PATH="http://cdn.NetBSD.org/pub/pkgsrc/packages/NetBSD/${uname -p}/${uname -r|cut -f '1 2' -d.}/All/"
# export PKG_PATH
# pkg_add pkgin
```

URL の「リリース」フィールドは、9.2 STABLE ではなく、9.2 のようにする必要があることに注意してください。

pkgin は、pkgsrc バイナリ パッケージを管理するための主要な高レベル ツールであり、aptなどの他のパッケージ マネージャーのユーザーが使い慣れるように設計されています。

```
# pkgin search nginx
nginx-1.19.6      Lightweight HTTP server and mail proxy server
nginx-1.18.0nb8  Lightweight HTTP server and mail proxy server
# pkgin install zsh nginx-1.19.6 vim
# pkgin upgrade
```

NetBSD パッケージのブラウズ可能な Web インデックスが利用可能です。

バイナリ パッケージのインストール (illumos, macOS, Enterprise Linux)

SmartOS/illumos, macOS, NetBSD current, および Enterprise Linux のバイナリ パッケージ セットは、MNX から入手できます。

ソースパッケージをインストールする

pkgsrc を入手する

CVS で pkgsrc HEAD をチェックアウトできます。

```
$ CVS_RSH=ssh cvs -danoncv@anoncv.NetBSD.org:/cvsroot checkout -P pkgsrc
```

または pkgsrc の安定版ブランチ:

```
$ CVS_RSH=ssh cvs -danoncv@anoncv.NetBSD.org:/cvsroot checkout -r pkgsrc-2022Q4 -P pkgsrc
```

または、HTTP から pkgsrc をダウンロードします。

- 現在のブランチを tarball としてダウンロードする
- 安定版ブランチを tarball としてダウンロードします

ブートストラップ

NetBSD 以外のオペレーティング システム、およびデフォルト以外の設定 (異なるインストール PREFIX など) の NetBSD では、pkgsrc は最初にブートストラップ (ツールのビルドとインストール) を行う必要があります。

```
$ cd pkgsrc/bootstrap
$ ./bootstrap --prefix /opt/pkg-2022Q4 --prefer-pkgsrc yes --make-jobs 4
```

パッケージをビルドする

依存関係は自動的にダウンロード、ビルド、およびインストールされます。

```
$ cd pkgsrc/devel/memcached
$ bmake install clean
```

ブートストラップされていない NetBSD インストールでは、bmake単にmake.

ビルドされたパッケージは、バイナリ パッケージ (pkgin, pkg_info など) とまったく同じツールで管理できます。

pbulk, pkg_chk, pkg_rolling-replaceなど、pkgsrc パッケージのビルドとインストールを自動化するさまざまなツールが存在します。

自分でパッケージングを開始する

自分でソフトウェアをパッケージ化して貢献したい場合は、pkgsrc-wipを見てください。



The NetBSD Project

"Of course it runs NetBSD"

ホーム | [RSS](#) | [リリースエンジニアリング](#) | [開発](#) | [NetBSD 財団](#) | [ネットワークング](#) | [一般](#) | [ポート](#) | [セキュリティ](#) | [イベント](#) | [パッケージ](#) | [ログイン](#)

ブックマーク

[NetBSD プロジェクト](#)

[NetBSD ウィキ](#)

フィード

[全て](#)

[/リリースエンジニアリング](#)

[/発達](#)

[/NetBSD 財団](#)

[/ネットワークング](#)

[/全般的](#)

[/ポート](#)

[/安全](#)

[/イベント](#)

[/パッケージ](#)

[コメント](#)

NetBSD 10.0 ベータ版が利用可能

2022 年 12 月 20 日 投稿者 *Nia Alarie*

ほぼ3年間の開発（作業は 2019 年後半に NetBSD 10 で開始されました）を経て、興味のあるユーザーがテストできるように BETA スナップショットがいつに公開されました。最終リリースにタグを付ける前に、今後数か月にわたって開発ブランチからさらに多くの変更がバックポートされるため、ベータ イメージは更新され続けます。

何を期待します

NetBSD 10.0 は、特にマルチコア システムでのパフォーマンスの主要なマイルストーンであると予想されますが、現在、ベータ ビルドではいくつかの追加のカーネル診断が有効になっているため、パフォーマンスが多少低下する可能性があります。

NetBSD 10 で期待できる機能には、WireGuard® との互換性、自動スワップ暗号化、新しいディスク暗号化方式、およびカーネルでの CPU アクセラレーションを含む、再加工された暗号化があります。ハードウェア サポートでは、Linux 5.6 から更新された GPU ドライバー、より多くの ARM ハードウェア (Rockchip RK356X、NXP i.MX 8M、Amlogic G12、Apple M1 を含む) のサポートがあります。、および Raspberry Pi 4)、最新の ARM CPU に見られる新しいセキュリティ機能のサポート、Realtek 2.5 ギガビットおよび新しい Intel 10/25/40 ギガビット イーサネット アダプターのサポート、compat_linux は AArch64 に移植され、DTrace は MIPS に移植されました。レトロコンピューティング愛好家向けには、Alpha でのマルチプロセッサ サポートが改善され、iMac G5 のサポートが強化されています。Xen ハイパーバイザーのサポートが大幅に変更されました。手動でディスクを TRIM する blkdiscard(8)、オーディオ ボリュームを制御する aiomixer(1)、realpath(1)、fsck_udf(8) など、さまざまな新しいユーザー空間プログラムがあります。そして、さらにロード...

管理者が NetBSD 9 からアップグレードする際に関連する可能性のある小さな詳細が多数あるため、実験システムをアップグレードする前に、最終リリースの発表を待って読んでください。ブリッジ エンドポイントとして tap(4) を使用するネットワーク設定は、代わりに vether(4) を使用するように変更する必要があることに注意してください。また、セキュリティ上の理由から、compat_linux はカーネルに組み込まれなくなりました（代わりにモジュールとしてロードします）。DisplayPort と HDMI オーディオがデフォルトの x86 カーネルで有効になっているため、サウンド出力が得られない場合は、audiocfg(1) を使用してデフォルトのオーディオ デバイスを変更する必要がある場合があります。blacklistd(8) は、blocklistd(8) に名前が変更されました。

ダウンロード

- [インストール / 64 ビット x86 用の USB スティック イメージ \(ハイブリッド UEFI/BIOS イメージ\)](#)。Gunzip と dd を使用して、Unix では .img.gz ファイルを、Windows では Rawrite32 を書き込みます。
- [Installation / USB stick image for 64-bit x86 \(legacy BIOS only\)](#)
- [Installation / USB stick image for 32-bit x86](#)
- [For images for various ARM devices, see armbsd.org](#). All Arm boards without UFEI or Raspberry Pi firmware also require U-Boot to be written the SD card separately when using a generic image.
- [Live image for Raspberry Pi 0 or 1](#)
- [Live image for Cavium Octeon \(also needs U-Boot\)](#)
- [CD/DVD image for 64-bit x86 \(for actual CD/DVDs and virtual machines only\)](#)
- [CD/DVD image for 32-bit x86](#)
- [CD/DVD image for 64-bit Arm virtual machines](#)
- [CD/DVD image for SPARC64](#)
- [CD/DVD image for Alpha](#)
- [More? \(e.g. Apple PowerPC/68K, Amiga, Dreamcast, PA-RISC, SGI MIPS, SPARC32, VAX...\)](#)

Packages

サードパーティのソフトウェアは、これまでどおり `pkgsrc` から入手できます。amd64 用のバイナリパッケージを利用できます。i386 用のバイナリも近日中に公開したいと考えています。

バグの報告

[ここでバグを報告できます!](#)

[7 コメント]

◀ [Oracle 上の NetBSD Arm...](#) | [メイン](#) | [再現可能なビルド...](#) ▶

コメント:

pkgsrc-2022Q4 ブランチがリリースされました

- 宛先: pkgsrc-users%NetBSD.org@localhost
- 件名: pkgsrc-2022Q4 ブランチがリリースされました
- From: Thomas Klausner <wiz%NetBSD.org@localhost>
- 日付: 2022 年 12 月 29 日 (木) 02:04:12 +0100

pkgsrc 開発者は、77 回目の四半期リリースを発表できることを誇りに思います
クロスプラットフォームのパッケージングシステムである pkgsrc の pkgsrc が利用可能
27,000 以上のパッケージを備え、23 のプラットフォームをサポートし、そのうち 10 のプラットフォームをサポートします。
現在動作していることが知られています。pkgsrc 自体の詳細情報<https://www.pkgsrc.org/>
で入手できます。

合計で、215 個のパッケージが追加され、63 個のパッケージが削除され、2326 個のパッケージが削除されました。
パッケージの更新 (1687 個の固有のパッケージまで) が処理されました。
pkgsrc-2022Q3 リリース。更新には、56 個の perl5 パッケージ、570 個の Python が含まれます
パッケージ、184 個の Ruby パッケージ、および 39 個の TeX パッケージ。

この発表では、関連するパッケージをグループ化して概要を説明します
単一の名前で、アップストリームの複数のバージョンを扱う
論理ユニットとしてパッケージ化します。したがって、「gcc 12」、新しいエントリ
「lang/gcc12」は「lang/gcc*」のアップデートです。

2022Q4 リリースでは、次の重要なパッケージを歓迎します
追加:

- suse15 エミュレーション パッケージ
- 意味する
- パイソン 3.11.1
- PostgreSQL 15.1
- Qt 6
- PHP 8.2
- クボ 0.17.0

次の重要な更新を歓迎します。

- 証明書ボット 2.1.0
- CMake 3.25.1
- Emacs 28.2
- FFmpeg 5.1.2
- Firefox 107.0.1
- GIMP 2.99.14 (GTK3 バージョン)
- 1.19.4に行く
- ハーフバズ 5.3.1
- LXQt 1.2.0
- マトリックスシナプス 1.73.0
- メイト1.26
- NodeJS 18.12.1
- qgis 3.28.2
- さび 1.64.0
- wxGTK 3.2.1
- yt-dlp 2022.11.11

いつものように、多くのパッケージが
アップストリームのマイクロ リリース。例は次のとおりです。

- Firefox ESR 102.6.0
- 1.18.9に行く
- マリアDB 10.6.11、10.5.18
- NodeJS 16.18.1、14.21.1
- PostgreSQL 14.6、13.9、12.13、11.18、10.23
- パイソン 3.10.9、3.9.16、3.8.16
- Qt および PyQt5 5.15.7
- Xfce4 [一部のコアおよびプラグイン コンポーネントが更新されました]

Port-arm archive

[\[Date Prev\]](#)[\[Date Next\]](#)[\[Thread Prev\]](#)[\[Thread Next\]](#)[\[Date Index\]](#)[\[Thread Index\]](#)[\[Old Index\]](#)

2023-03-04-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)

- To: port-arm%NetBSD.org@localhost
- Subject: 2023-03-04-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)
- From: Jun Ebihara <jun@soum.co.jp@localhost>
- Date: Sat, 04 Mar 2023 20:12:55 +0900 (JST)

I've updated 2023-03-04-netbsd-raspi-earmv6hf.img.gz for RPI.

<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2023-03-04-earmv6hf/2023-03-04-netbsd-raspi-earmv6hf.img.gz>
<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2023-03-04-earmv6hf/MD5>

Update:

- NetBSD 10.99.2 evbarm-earmv6hf 2202303030400Z rpi.img from nyftp.

- armv6 on QEMU (was Re: armv6 support for rust)
<http://mail-index.netbsd.org/tech-pkg/2022/10/21/msg026888.html>
- armv6 support for rust?
<http://mail-index.netbsd.org/tech-pkg/2022/10/19/msg026864.html>

- Re: Raspberry Pi camera under NetBSD current
<http://mail-index.netbsd.org/current-users/2021/11/16/msg041683.html>

1. firmware copy start* and fix*
2. sync dtb

- Firmware update
#commit 2595b815bf0e3c2af4b9a6be2151661df1ed4710 (HEAD -> master, origin/master, origin/HEAD)
#Author: Dom Cobley <popcornmix@gmail.com@localhost>
#Date: Fri Apr 1 11:41:42 2022 +0100
kernel: drm/vc4: Force trigger of dlist update on margins change
See: <https://github.com/raspberrypi/linux/pull/4970>

- Raspberry Pi zero 2 W
<http://mail-index.netbsd.org/port-arm/2022/02/14/msg007593.html>
earmv7hf works well.earmv6hf not yet.

- Raspberry Pi [0-3] have been supported in big-endian mode (Rin Okuyama)
XXX: should test big-endian image and pkgsrc

<http://mail-index.netbsd.org/port-arm/2021/06/17/msg007310.html>
"earmv7hfeb works fine so far, pkgsrc included. aarc64eb doesn't boot on a Raspberry Pi 3,"

- Fw: gstreamer1 updated, raspberry help needed
<http://mail-index.netbsd.org/port-arm/2021/11/17/msg007498.html>
- multimedia/gst-plugins1-egl-glib
- multimedia/gst-plugins1-egl-opengl
- multimedia/gst-plugins1-omx

PR

#55505 RaspberryPi3A+ can't find Wi-Fi module
<http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=55505>

#54941
Raspberry Pi Zero W serial console corrupted when CPU frequency changed
<http://gnats.netbsd.org/54941>

- RPI4: testing on NetBSD/aarch64
<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/README>

Changes from 2021-09-18 version
<https://github.com/ebijun/NetBSD/commit/4b4566c7b33b38a9aa9b3ae7326d069d431a1d63>

- pkgin support
check /usr/pkg/etc/pkgin/repositories.conf.
I add
<http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2021-1>
for testing, with pkg_summery.gz .

sysinfo:
sh-20220122-20230302054510Z

pkgsrc:
curl-7.88.1 dbus-1.14.6 glib2-2.74.6 gnutls-3.8.0nb1 libproxy-0.4.18
nghttp2-1.52.0 ruby31-atk-4.1.2 ruby31-cairo-gobject-4.1.2 ruby31-gdk3-4.1.2
ruby31-gdk-pixbuf2-4.1.2 ruby31-gio2-4.1.2 ruby31-glib2-4.1.2
ruby31-gobject-introspection-4.1.2 ruby31-gtk3-4.1.2 ruby31-pango-4.1.2
sudo-1.9.13p2 uim-1.8.9

Need Checking:

- bluetooth keyboard/mouse setting
- RPI camera module
<http://mail-index.netbsd.org/current-users/2021/11/16/msg041683.html>

Got Reports:

- HDMI output works very well however I wanted to change the resolution into 800x600 both tty console and X Window graphical modes. Where should I change it? As config.txt with hdmi_group=1, hdmi_mode=1 or hdmi_safe=1

didn't work.

- USB input devices seem to work fine as well with my USB keyboard, mouse and barcode reader. However for my USB output device such as my Xprinter printer didn't work with device driver of ulpt(4). It is detected and working in OpenBSD.[9.0 release also occurs same error]

```
[ 195.114857] ulpt0 at uhub1 port 5 configuration 1 interface 0
[ 195.114857] ulpt0: Xprinter (0x0483) USB Printer P (0x5743), rev 2.00/1.00, addr 5, iclass 7/1
[ 195.114857] ulpt0: using bi-directional mode
```

```
rpi# cat myfile.txt > /dev/ulpt0
-sh: cannot create /dev/ulpt0: error 16
```

- with my Raspberry Pi 3B because every time I invoked this command "shutdown -h now" a kernel panic occurs relating to usb transfer.
- After shutting-down leaving the system unplugged, this will turn it's processor very very hot and seems harmful to the system.

pre-installed packages:
<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/pkgsrc/pkginfo>

Keyboard layout checkpoint:
<http://www.netbsd.org/docs/guide/en/chap-cons.html>

```
/etc/wscons.conf
#encoding sv
#encoding us.swapctrlcaps
encoding jp
```

System Update:
<http://cvsweb.netbsd.org/bsdweb.cgi/src/distrib/sets/lists/base/shl.mi>

mikutter support :
 I make sample API key,pre-installed.
<https://github.com/ebijun/NetBSD/tree/master/RPI/RPIimage/root/.mikutter/plugin>
<https://github.com/Akkiesoft/how-to-make-mikutter-work-again>
 cd /root/.mikutter
 git submodule add https://github.com/toshia/twitter_api_keys.git twitter_api_keys

RPI Wifi:
<http://mail-index.netbsd.org/port-arm/2019/08/31/msg006102.html>

Overview:
http://wiki.NetBSD.org/ports/evbarm/raspberry_pi/

QEMU,with GENERIC kernel : vexpress to GENERIC: testing
<https://github.com/ebijun/NetBSD/tree/master/vexpress/Boot>
<https://github.com/ebijun/NetBSD/tree/master/vexpress/vexpress-v2p-ca15-tc1.dtb>
<http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2023-03-04-earmv6hf/QEMU/netbsd-GENERIC.ub.gz>
<http://mail-index.netbsd.org/port-arm/2017/06/02/msg004154.html>

dmesg:
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI0>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI0W>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI2>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI2-1.2>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3A+>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/earmv6hf/RPI3B+>

Problem:
 - CPUFLAGS=-mfpu=neon-vfpv4 breaks some programs
<http://gnats.netbsd.org/52528>
 - webkit24-gtk-2.4.11nb9 compile failed.
 - dillo-3.0.5nb2 works with -O0
 thanx Jared McNeill.

security.pax.mprotect.enabled
http://netbsd.org/gallery/presentations/msaitoh/2016_BSDCan/BSDCan2016-NetBSD.pdf
 man security
 man paxctl
 sysctl -a |grep pax
 If application failed, such as omxplayer.
 try to test
 sysctl -w security.pax.mprotect.enabled=0

Automatic resize partition: see /etc/rc.conf and /etc/fstab
 1. copy image to SD/MicroSD
 2. Boot
 3. Calculate and resize ld0 partition and automatic reboot
 4. after the reboot,root partition fit for your card.
<http://movapic.com/pic/20150416115108552fa22c4f225>
 In this image, ld0a re-created with newfs -b 4096.

pkgsrc:
 # cd /usr
 # ftp <http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz>
 # ls /usr/pkgsrc ... check if exists.
 # tar tzvf pkgsrc.tar.gz |head ... check the archive
 # tar xzvf pkgsrc.tar.gz ... extract
 # ls /usr/pkgsrc ... check what extracted
 # pkg_chk -g ... List to/usr/pkgsrc/pkgchk.conf
 # (cd /usr/pkgsrc; cvs update -PA) ... update
 # pkg_chk -un ... Update (listup)
 # pkg_chk -u ... Update

I use /usr/pkgsrc with USB SSD disk.

Pre-compiled packages:
 - Pre-compiled packages path setting: man 5 pkg_install.conf

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

2023-03-04-netbsd-raspi-aarch64.img

- To: port-arm%NetBSD.org@localhost
- Subject: [2023-03-04-netbsd-raspi-aarch64.img](#)
- From: Jun Ebihara <jun%soum.co.jp@localhost>
- Date: Sat, 04 Mar 2023 14:18:03 +0900 (JST)

I've updated 2023-03-04-netbsd-raspi-aarch64.img.gz for RPI4.

<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2023-03-04-aarch64/2023-03-04-netbsd-raspi-aarch64.img.gz>
<https://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2023-03-04-aarch64/MD5>

Topics:

- NetBSD-10.99.2
- Raspberry Pi 4 UEFI Firmware v1.34
Fix PCI DT node address in SyncPcie() [tianocore/edk2-platforms@ad00518]
- SD/MMC Configuration
Switch Default uSD/eMMC Routing: Arasan SDHCI -> EMMC2
<http://mail-index.netbsd.org/port-arm/2021/08/08/msg007393.html>
the Arasan SDHCI is used to connect bwfm(4).

In this image, a symbolic link added

```
/libdata/firmware/if_bwfm:
ln -s brcmfmac43455-sdio.raspberrypi,4-model-b.txt "brcmfmac43455-sdio.Raspberry Pi 4 Model B.txt"
to avoid
bwfm0: autoconfiguration error: NVRAM file not available
```

- only for RPI4
(fill dtb for RPI3?)
- build script
<https://github.com/ebijun/NetBSD/tree/master/RPI/RPIimage/Image/aarch64>
- [Issue] Can't recognize DOS partition from Windows
Should I Use MBR version image?
- rpi cm4 + waveshare mini base board B seems stop after kernel load
fixed: To use USB, set UEFI+FDT
<http://mail-index.netbsd.org/port-arm/2021/06/30/msg007343.html>
<http://mail-index.netbsd.org/port-arm/2022/10/03/msg007875.html>
- pkgin support
edit /usr/pkg/etc/pkgin/repositories.conf
cf.
<http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/aarch64/2023>
and pkgin update

sysinfo:

```
bind-9.16.37 openssl-1.1.1t sh-20220122-20230215063151Z
```

pkgsrc:

```
brotil-1.0.9 cairo-1.16.0nb9 cairo-gobject-1.16.0nb7
dillo-3.0.5nb12 fltk-1.3.8nb2 fontconfig-2.14.2
freetype2-2.13.0 gd-2.3.3nb6 gdk-pixbuf2-2.42.10nb2
git-base-2.39.1 glib2-2.74.5 graphite2-1.3.14nb1
gtk2+-2.24.33nb9 gtk3+-3.24.36nb2 harfbuzz-6.0.0
libXft-2.3.7nb1 libotf-0.9.16nb3 libxml2-2.10.3nb2
m17n-lib-1.8.0nb10 medit-1.2.0nb23 mlterm-3.9.2nb10
pango-1.50.12nb1 py310-cElementTree-3.10.10
py310-expat-3.10.10nb1 python310-3.10.10 qt5-qtbase-5.15.8nb1
qt5-qt5xmlpatterns-5.15.8nb1 qt5-qt5xmlpatterns-5.15.8nb1
ruby31-atk-4.1.1 ruby31-cairo-1.17.8nb1 ruby31-cairo-gobject-4.1.1
ruby31-gdk3-4.1.1 ruby31-gdk-pixbuf2-4.1.1 ruby31-gio2-4.1.1
ruby31-glib2-4.1.1 ruby31-gobject-introspection-4.1.1
ruby31-gtk3-4.1.1 ruby31-mikutter-5.0.4nb3 ruby31-pango-4.1.1
sudo-1.9.12p2 tiff-4.5.0nb1 zstd-1.5.4
```

RPI4 status:

- boot via UEFI_VER=v1.34
<https://github.com/pftf/RPi4/releases>

In this image, pre-setted /boot/RPI_EFI.fd

1. Advanced Configuration
-> Limit RAM to 3GB Disabled
2. Boot order, from microSD

- HDMI: works
- audio: vcaudio
<http://mail-index.netbsd.org/source-changes-d/2021/01/22/msg013133.html>
- USB device: works
- bwfm0 Wireless network?:
(works well on RPI3, not works on RPI4?)
- Pi 4 Bootloader Configuration
https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2711_bootloader_config.md

Update:

- NetBSD-current 10.99.2 evbarm-aarch64 202303030400Z arm64.img from nyftp.
- pkginfo
<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/pkginfo>
- sysinfo
<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIimage/Image/aarch64/sysinfo>

Update from 20221-06:
<https://github.com/ebijun/NetBSD/commit/1b3ef330ecf023aa73d58ccb8b4029c520f33cdf>

```
- mate/firefox/inkscape/libreoffice works well
  pkg_add mate
  pkg_add gvfs
  pkg_add firefox
  pkg_add firefox-l10n
  pkg_add libreoffice
  pkg_add inkscape
  pkg_add gimp
  pkg_add scribus
  pkg_add py310-sphinx
  pkg_add wxGTK30
  pkg_add gnome-keyring

- Sphinx with latexpdf
  https://github.com/ebijun/NetBSD/blob/master/Guide/latexpdf.rst

- pkgin: upgrade failed
- raspberry-pi userland: compile failed
```

Pre-installed packages:

```
bash
tcsh
vlgothic-ttf
icewm
xli
ruby27-mikutter
uim
fossil
sudo
git-base
zsh
mlterm
pkg_chk
lintpkgsrc
mozilla-rootcerts
dillo
medit
lrzsz
```

```
- pre-build packages:
  See /etc/pkg_install.conf
  PKG_PATH=http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/aarch64/2023/
```

```
pkgsrc:
# cd /usr
# ftp http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz
# ls /usr/pkgsrc ... check if exists.
# tar tzvf pkgsrc.tar.gz |head ... check the archive
# tar xzvf pkgsrc.tar.gz ... extract
# ls /usr/pkgsrc ... check what extracted
# pkg_chk -g ... List to/usr/pkgsrc/pkgchk.conf
# (cd /usr/pkgsrc; cvs update -PA) ... update
# pkg_chk -un ... Update (listup)
# pkg_chk -u ... Update
```

EEPROM version:

```
Check&update with Raspberry Pi OS (pi:raspberrypi)
pi@raspberrypi:~$ cat /etc/default/rpi-eeeprom-update
FIRMWARE_RELEASE_STATUS="stable"
pi@raspberrypi:~$ sudo rpi-eeeprom-update
BCM2711 detected
Dedicated VL805 EEPROM detected
BOOTLOADER: up-to-date
CURRENT: Thu 16 Jul 15:15:46 UTC 2020 (1594912546)
LATEST: Thu 16 Jul 15:15:46 UTC 2020 (1594912546)
FW DIR: /lib/firmware/raspberrypi/bootloader/stable
VL805: up-to-date
CURRENT: 000138a1
LATEST: 000138a1
```

--
Jun Ebihara

-
- Prev by Date: [install binary and pkgsrc](#)
 - Previous by Thread: [install binary and pkgsrc](#)
 - Indexes:
 - [reverse Date](#)
 - [reverse Thread](#)
 - [Old Index](#)

[Home](#) | [Main Index](#) | [Thread Index](#) | [Old Index](#)

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

pinebook status update (20230305)

- To: port-arm%netbsd.org@localhost
- Subject: pinebook status update (20230305)
- From: Jun Ebihara <jun%soum.co.jp@localhost>
- Date: Sun, 05 Mar 2023 16:44:07 +0900 (JST)

NetBSD Arm Bootable Images: NetBSD9/NetBSD-current
<https://nycdn.netbsd.org/pub/arm/>

Update:
bind-9.16.37
openssl-1.1.1t

Problems:

- pinebook:
Using audio interface, rebooting.
PR port-arm/57111
- Pinebook Pro graphics noticeably slower since DRM update
fixed: PR port-arm/56596
<http://gnats.netbsd.org/cgi-bin/query-pr-single.pl?number=56596>
- gimp with -python option: don't depends on python27 packages
- seamonkey 2.53.10 can't start
- gnucash build failed
- USB Audio 2.0 Support
<http://mail-index.netbsd.org/current-users/2022/06/03/msg042467.html>
<https://raw.githubusercontent.com/openbsd/src/master/sys/dev/usb/uaudio.c>

Topics:

- Tow-Boot on Pinebook Pro
<https://tow-boot.org/devices/pine64-pinebookPro.html>
need more testing on NetBSD.
- Pinebook GPU
"I think I am making a bit of progress with the DRM driver for the Pinebook."
<http://mail-index.netbsd.org/port-arm/2022/05/12/msg007650.html>
- pkgsrc update as after 2022Q1.
firefox 98.0.2 and libreoffice 7.3.1.3
<https://twitter.com/ebijun/status/1513773516402790403>
- graphical mixer for NetBSD audio.

on pinebook:

```
Select a mixer device:
[*] /dev/mixer0: sun50i-a64-audio ausoc
[ ] /dev/mixer1: hdmi-audio ausoc
```

on pinebook pro:

```
XXX: [inputs] tab controls audio output level.
https://twitter.com/ebijun/status/1395184397599469568
```

- Audio CD with wavpack

Encode:

```
On Windows: Exact Audio Copy & foobar2000
On pkgsrc:
1. pkg_add abcde; pkg_add glyr
   abcde -d /dev/rcd0a -B -o wav -l -a default,cue
   -> wav,cue,jpg file created.
2. pkg_add wavpack
   wavpack -h .wav --write-binary-tag "Cover Art (Front)=@.jpg" -w "cuesheet=@.cue"
   -> wv file created.
```

Play:

```
qmp: can play wavpack file with cue. How can I show Cover Art?
Setting->Output->OSS plugin(liboss.so)
```

- audio0 input issue

```
http://mail-index.netbsd.org/port-arm/2021/02/22/msg007185.html
"Is audio input from the inbuilt microphone(s) working? I get "read failed:
Resource temporarily unavailable" from audiorecord"
audio0(ausoc1): trigger_input failed: errno=5
audio0(audoc1): device timeout
```

pkgsrc:

- works : inkscape, scribes, seamonkey, minitube, mikutter, xournalpp, libreoffice

- firefox 90.0 and libreoffice 7.1.4.2 on NetBSD 9.99.85/aarch64 on pinebook.

zoom meeting with firefox on NetBSD/aarch64 on pinebook pro
with UserAgent switcher addon as Linux.

Send: Share Screen, camera/USB Camera
Problem: Can't show others screen/in-camera.

- pkgin support

```
make pkg_summary. and pkgin works.
cd /usr/pkgsrc/packages/All
find . -name '*.tgz' -exec pkg_info -X {} \; > pkg_summary
```

```

gzip -f pkg_summary

pkgsrc packages:
http://cdn.netbsd.org/pub/NetBSD/misc/jun/aarch64/aarch64/2021-1/

- nono: luna68k/luna88k/x68k emulator
  add /etc/mk.conf
  ACCEPTABLE_LICENSES= nono-license
  cd /usr/pkgsrc/emulators/nono;make ;make package-install
- search sound mixer GUI to manage mixerctl

ToDo:
- 3D acceleration

- Pinebook Pro
  mate/xfce4/camera/WiFi
  ToDo:
  - HDMI output via USB-C
  - Audio output: can't switch speaker & headphone out

- Pinebook
  mate/xfce4/camera/HDMI
  ToDo:
  - internal WiFi

System Updates:

sysinfo:
bind-9.16.37 openssl-1.1.1t sh-20220122-20230304134236Z

pkgsrc:
cairo-1.16.0nb9 cairo-gobject-1.16.0nb7 fontconfig-2.14.2
gd-2.3.3nb6 graphviz-7.1.0nb1 gspell-1.8.3nb1 gtk2+-2.24.33nb9
gtk3+-3.24.36nb2 gtksourceview4-4.8.0nb9 harfbuzz-7.0.0
libXft-2.3.7nb1 libcanberra-0.30nb9 libwnck3-3.36.0nb4
libxfce4ui-4.18.2 nono-0.5.0 pango-1.50.12nb1 poppler-23.02.0
poppler-glib-23.02.0 qt5-qtbase-5.15.8nb1 qt5-qtxmlextras-5.15.8nb1
qt5-qtxmlpatterns-5.15.8nb1 talloc-2.4.0 tevent-0.14.1
tiff-4.5.0nb1 tk-8.6.12nb1 ttmkfdir2-20021109nb7 uim-1.8.9
unicode-character-database-15.0.0 unicode-emoji-15.0 vala-0.56.3nb1
vte-0.28.1nb30 vte3-0.68.0nb11 web2c-2022nb1 wxGTK30-3.0.5.1nb10
xapian-1.4.22 xdvik-22.87.06nb1 xfce4-appfinder-4.18.0nb2
xfce4-conf-4.18.0nb2 xfce4-desktop-4.18.0nb2 xfce4-exo-4.18.0nb2
xfce4-garcon-4.18.0nb3 xfce4-mousepad-0.6.0 xfce4-orage-4.18.0
xfce4-panel-4.18.0nb2 xfce4-session-4.18.1 xfce4-settings-4.18.2
xfce4-terminal-1.0.4nb3 xfce4-thunar-4.18.4 xfce4-tumbler-4.18.0nb2
xfce4-wm-4.18.0nb2 xfce4-wm-themes-4.10.0nb20 xkeyboard-config-2.38
xnp2-0.86nb18 xorg-cf-files-1.0.8 xorg-util-macros-1.20.0
xscreensaver-6.05.1nb3 zenity-3.41.0nb5 zstd-1.5.4

System Updates:
https://github.com/ebijun/NetBSD/commit/2e07db8ce05b7f8dc77b266b9ba98a7cc8dfd170

Tips:
- System Clocks
% sysctl -a |grep freq

Install pkgsrc/sysutils/estd from pkgsrc and start it on bootup
for automatic up/downscaling.

pinebook:
machdep.cpufreq.cpu0.available = 1152 1104 1008 816 648 408
# sysctl -w machdep.cpufreq.cpu0.target=1152

pinebook-pro:
machdep.cpufreq.cpu0.available = 1416 1200 1008 816 600 408
machdep.cpufreq.cpu4.available = 2000 1800 1608 1416 1200 1008 816 600 408
# sysctl -w machdep.cpufreq.cpu0.target=1416
# sysctl -w machdep.cpufreq.cpu4.target=2000

- Battery Status
  envstat
  xbattbar
  XXX: link from dbus or mate-power-manager?

- HDMI port works on pinebook.
  Connect display to pinebook HDMI port
  # pkg_add arandr
  % arandr

- mate: pkgsrc/meta-pkgs/mate
https://twitter.com/ebijun/status/1086814579647102976
  pkg_add mate
  echo "mate-session" >> .xinitrc
  startx

- ibus
  export XMODIFIERS=@im=ibus

- Add eDP transmitter and enable display pipeline
http://mail-index.netbsd.org/source-changes/2019/02/03/msg103047.html
  arandr works well: display up-side down
https://twitter.com/ebijun/status/1092885154127724544

- keymapping: [Shift][<>] on llinch
https://wiki.netbsd.org/ports/evbarm/allwinner/#index3h2

- replace AC Adapter for heavy load
  Anker PowerPort10 A2133+USB DC Cable with 3.5mm/1.35mm connector
  USB-C laptop charger from HP

- USB Wifi: urtwn0/run0 works

```

urtnw0: I-0 DATA DEVICE, INC. (0x4bb) WN-G150UM (0x94c)
pinebook: internal Wi-Fi not supported
pinebook pro: internal bwfm0 works,sometimes hangs

Version:

NetBSD 10.99.2 (GENERIC64) #0: Sat Mar 4 13:42:36 UTC 2023 mkrepro%mkrepro.NetBSD.org@localhost:/usr/src/sys/arch/evbarm/compile/GENERIC64

dmesg:

<https://github.com/ebijun/NetBSD/blob/master/dmesg/aarch64/pinebook>
<https://github.com/ebijun/NetBSD/blob/master/dmesg/aarch64/pinebook-pro>

System image: dd to eMMC.

<http://www.invisible.ca/arm/>

pinebook-pro:

1. Insert microSD, boot (Linux from eMMC comes up),
 2. hit "Restart",
 3. It will boot from microSD.
 4. dd'ing to eMMC for installation, the eMMC is faster
- eMMC can disable with switch inside,and boot MicroSD.

Getting Started with NetBSD on the Pinebook Pro by Benny Siegert

<https://bentsukun.ch/posts/pinebook-pro-netbsd/>

sysinfo:

<https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/sysinfo>

pkginfo:

<https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/pkginfo>

pkgchk.conf:

<https://github.com/ebijun/NetBSD/blob/master/pinebook/Update/pkgchk.conf>
cp pkgchk.conf /usr/pkgsrc;pkg_add pkg_chk; pkg_chk -au

--

Jun Ebihara

- Prev by Date: **Re: install binary and pkgsrc**
- Previous by Thread: **2023-03-04-netbsd-raspi-earmv6hf.img (Re: Raspberry Pi update please.)**
- Indexes:
 - **reverse Date**
 - **reverse Thread**
 - **Old Index**

Home | **Main Index** | **Thread Index** | **Old Index**

Port-arm archive

[[Date Prev](#)][[Date Next](#)][[Thread Prev](#)][[Thread Next](#)][[Date Index](#)][[Thread Index](#)][[Old Index](#)]

apple m1 status update (20230303)

- **To:** port-arm%netbsd.org@localhost
- **Subject:** apple m1 status update (20230303)
- **From:** Jun Ebihara <jun%soum.co.jp@localhost>
- **Date:** Fri, 03 Mar 2023 12:40:13 +0900 (JST)

Install:

<https://wiki.netbsd.org/ports/evbarm/apple/>

Topics:

- X
startx failed with signal 11,
till NetBSD-10.99.2-evbarm-aarch64-202302150640Z

X.Org X Server 1.21.1.7

```
[ 40.591] (EE) Backtrace:
[ 40.592] (EE) 0: /usr/X11R7/bin/X (xorg_backtrace+0x3c) [0x9761b8c]
[ 40.592] (EE) 1: /usr/X11R7/bin/X (os_move_fd+0x78) [0x975d4f8]
[ 40.592] (EE) 2: /usr/lib/libc.so.12 (__sigtramp_siginfo_2+0x0) [0xf7d48807e1e8]
[ 40.592] (EE) 3: /usr/X11R7/bin/X (CreateColormap+0x418) [0x9644a68]
[ 40.592] (EE) 4: /usr/X11R7/bin/X (miCreateDefColormap+0x84) [0x96b7744]
[ 40.592] (EE) 5: /usr/X11R7/lib/modules/drivers/wsfb_drv.so (+0x147c) [0xf7d4874e322c]
[ 40.592] (EE) 6: /usr/X11R7/bin/X (AddScreen+0xac) [0x9639c6c]
[ 40.592] (EE) 7: /usr/X11R7/bin/X (InitOutput+0x26c) [0x967616c]
[ 40.592] (EE) 8: /usr/X11R7/bin/X (dix_main+0x1a0) [0x9608fd0]
[ 40.592] (EE)
[ 40.592] (EE) Segmentation fault at address 0xf7d487aa07f0
[ 40.592] (EE)
```

Fatal server error:

[40.592] (EE) Caught signal 11 (Segmentation fault). Server aborting

startx but colormap

<https://twitter.com/ebijun/status/1583622834990956545>

startx -- -depth 30

<http://mail-index.netbsd.org/port-arm/2021/10/21/msg007458.html>

<https://twitter.com/ebijun/status/1480042814704873473>

application checked via tigervnc.

<https://twitter.com/ebijun/status/1615544069161357314>

- pkgsrc/net/tigervnc build failed depends:

x11/libXfont2:

ERROR: This package has set PKG_SKIP_REASON:
ERROR: Package set is using native X11 component
works with comment out
#.include "../x11/libXfont2/buildlink3.mk"

- pkgsrc/x11/xterm failed

ld: fontutils.o: undefined reference to symbol 'FT_Load_Char'
ld: /usr/X11R7/lib/libfreetype.so.19: error adding symbols: DSO missing from command line

Hardware:

Apple Mac mini M1 2020 A2348
USB SSD 512GB BUFFALO SSD-PGM2.0U3-BC

Filesystem	1M-blocks	Used	Avail	%Cap	Mounted on
/dev/dk6	480587	401267	55290	87%	/
/dev/dk5	79	34	44	43%	/boot
ptyfs	0	0	0	100%	/dev/pts
procfs	0	0	0	100%	/proc
tmpfs	1913	0	1913	0%	/var/shm

ToDo:

- audio
- Wireless LAN

pkgsrc:

- pkgsrc packages:

<http://cdn.netbsd.org/pub/NetBSD/misc/jun/aarch64/aarch64/2022/pkgin-support>

System Updates:

sysinfo:

sh-20220122-20230301200841Z

pkgsrc:

```
ImageMagick-7.1.0.62 curl-7.88.1 dav1d-1.1.0 firefox-110.0
firefox-l10n-110.0 font-util-1.4.0 gdbus-codegen-2.74.6
glib-networking-2.74.0 glib2-2.74.6 glib2-tools-2.74.6
gnupp2-2.4.0 gnutls-3.8.0nb1 libass-0.17.1 libproxy-0.4.18
lmbd-0.9.30 meson-1.0.1 nghttp2-1.52.0 openal-soft-1.23.0
p5-HTTP-Daemon-6.15 poppler-23.02.0 poppler-cpp-23.02.0
poppler-glib-23.02.0 postgresql14-client-14.7 py310-build-0.10.0
py310-pip-23.0.1 py310-setuptools-67.3.3
py310-typing-extensions-4.5.0 py310-zipp-3.15.0 ruby31-atk-4.1.2
ruby31-cairo-gobject-4.1.2 ruby31-gdk3-4.1.2 ruby31-gdk-pixbuf2-4.1.2
ruby31-gio2-4.1.2 ruby31-glib2-4.1.2 ruby31-gobject-introspection-4.1.2
```

nono 0.5.1 (2023/02/12)

nono は NetBSD とかで動作する OMRON LUNA-I/LUNA-88K (と SHARP X68030) のエミュレータです。 [nono is OMRON LUNA-I/LUNA-88K (and SHARP X68030) emulator runs on NetBSD and etc.]

Index of this page:

- [1. ビルド方法 \[How to build\]](#)
- [2. 実行方法 \[How to execute\]](#)
- [3. 設定 \[Configuration\]](#)
- [4. VM について \[About VM\]](#)
- [5. 実行してみる \[Try it\]](#)
- [6. ホストネットワーク設定例 \[Example of host network setup\]](#)
- [7. 既知の問題 \[Known Issues\]](#)
- [8. 過去のバージョンからの移行方法 \[How to migrate from old versions\]](#)
- [9. 変更履歴 \[Changes\]](#)
- [10. 連絡先、ライセンス等 \[Contact, License, etc\]](#)

緑背景は新規または目立った更新のあった箇所です。 [Green Background is new or updated paragraph.]

▼ 1. ビルド方法 [How to build]

ビルドには以下が必要です。 [The followings are required for build.]

- make (BSD make, not GNU make)
- C/C++ compiler which supports `-std=c++14`.
(For gcc, 7.4 or newer works at least. For clang, 7.0 - 13.0 works at least.)
- wxWidgets 3.0.x "stable" branch or 3.2.x "new stable" branch
- gettext

wxWidgets は NetBSD(pkgsrc) なら pkgsrc/x11/wxGTK30 (OPTIONS:gtk3 で動作確認) です。2022Q3 の wxGTK32 ではビルドできないようです。 [If you use NetBSD(pkgsrc), wxWidgets is pkgsrc/x11/wxGTK30 (OPTIONS:gtk3 is tested). wxGTK32 in 2022Q3 doesn't seem to build for now.]

(NetBSD 以外でのビルドはサポートしていませんが) Ubuntu 20.04 ではたぶん以下のパッケージが必要です。 [You may need the following packages on Ubuntu 20.04 (though we won't support non-NetBSD platform).]

- `bmake build-essential gettext libbsd-dev libkqueue-dev libwxgtk3.0-gtk-dev zlib1g-dev`

nono のソースアーカイブを展開したら以下のようにビルドします。 `configure` には環境変数 `CC`、`CXX` でコンパイラを指定することができます。また `wx-config` が標準的な名前では提供されていないために見付けられない場合には環境変数 `WX_CONFIG` にパスを指定することができます。 [Extract the nono's source archive and build as following. You can specify C/C++ compiler using environment variable CC and CXX if configure cannot find standard name suitable compiler. Also, you can specify wx-config path using environment variable WX_CONFIG if configure cannot find wx-config.]

`-DNOTEST` は開発者用のテストツールをビルドしないためのものです。 [With `-DNOTEST`, the build excludes test tools which is needed for developers only.]

```
% ./configure
% make -DNOTEST depend
% make -DNOTEST
% su
# make install
```

2つの実行ファイルがインストールされます。 `nono` が GUI 版実行ファイル、 `nono-cli` がコマンドライン版です。 [Two executables will be installed. `nono` is the GUI executable and `nono-cli` is the command line executable.]

▼ 2. 実行方法 [How to execute]

`nono` は複数機種に対応しているため設定なしでは起動できません。設定は設定ファイルかコマンドラインオプションで指定します。`nono` はまず設定ファイル `~/nono.cfg` があれば読み込み、続いてカレントディレクトリの設定ファイル `nono.cfg` があれば読みます (`-c` オプションで設定ファイル (VM ディレクトリ) の位置を指定することも出来ます)。詳細は以下の設定の章を参照してください。 [nono supports multiple architectures so that it needs configuration. First, nono reads a configuration file names `~/nono.cfg` if exists, then reads a configuration file names `nono.cfg` in the current directory if exists (You can specify the configuration file path (or VM directory) using `-c` option). See the following Configuration section for details.]

▼ 2.1. コマンドラインオプション [Command Line Option]

`-c vmpath`

VM ディレクトリ/設定ファイルを指定します。 `vmpath` がディレクトリならそのディレクトリの中の `nono.cfg` を設定ファイルとします。 `vmpath` がファイルならそれを設定ファイルとします。そしていずれの場合も設定ファイルがあるディレクトリを VM ディレクトリとします。 `-c` オプションを省略すると `vmpath` をカレントディレクトリとします。 [Specifies the VM directory/configuration file. If `vmpath` is a directory, make `nono.cfg` in that directory a configuration file. Or if `vmpath` is a file, make the specified file a configuration file. And in both cases, make the directory where that file is located a VM directory. If `-c` option is omitted, `vmpath` is considered as the current directory.]

`--create-sram`

(X68030 Only) VM ディレクトリに X68030 用の SRAM.DAT がなければ初期状態で作成して終了します。 [nono will create the initial SRAM.DAT for X68030 and exit, if there is no SRAM.DAT in the VM directory.]

`-f`

高速モードで起動します。GUI なら起動後にもメニューから変更できますが、その初期値を変えるだけです。設定の `fast-mode=1` と等価です。[Boot as the fast mode. You can change this mode on GUI menu after boot, and the option only changes its initial state. This option is equivalent to `fast-mode=1` in configuration.]

--fontsize *height*

GUI 版のみ。全サブウィンドウの起動時のフォントサイズを指定します。起動後にメニューから変更できます。設定の `monitor-fontsize` と等価です。[GUI Only. Specifies the initial fontsize of all sub windows. You can change this value on GUI menu after boot. This option is equivalent to `monitor-fontsize` in configuration.]

-s *scale*

GUI 版のみ。メインウィンドウの起動時のスケールを実数で指定します。設定の `mainview-scale` と等価です。起動後はメニューからプリセットされた倍率と `mainview-scale` で指定された倍率には変更可能です。[GUI Only. Specifies the initial main window scale in real number. This is equivalent to `mainview-scale` in configuration. You can change this scale on GUI menu after boot.]

--show-config

設定ファイルと `-v` オプションを読み込んだ結果を表示します。[Shows the result of reading configuration file and parsing `-v` options.]

-v

バージョンを表示します。[Shows the version.]

-V *name=value*

設定ファイルで指定した `name=configvalue` の代わりにこのオプションの `name=value` を適用します。 `name` が正しくない場合はエラー終了します。[Use this `name=value` instead of `name=configvalue` specified in configuration file. If `name` is not correct, it will exit on error.]

-X *file*

ホストの `file` をロードして実行します。 `file` が相対パスの場合カレントディレクトリからのパスになります。ファイルが `gzip` 圧縮されていれば自動的に展開します。(展開後の) ファイル形式は以下の通りですが、実際にはブートローダとカーネル程度しか想定していません。また、いずれも起動元デバイスが取得できないなどの問題はあるかも知れません。[Loads and executes host's `file`. If `file` is relative path, it is path from the current directory. If the file is `gzip`'d, it is automatically extracted. The supported file format (after extracting) is the following. Actually, it only assumes bootloaders or kernels. And note that it may not obtain some information that where did I boot from, for example.]

- `a.out` (OMAGIC) 実行ファイル (おそらくブートローダのみサポート) [`a.out` (OMAGIC) executable (It probably supports bootloaders only)]
- ELF 実行ファイル (おそらくカーネルのみサポート。NetBSD の実行ファイルのようであれば、カーネルだと思ってシンボルテーブルも読み込みます) [ELF executable (It probably supports the kernel only. If the file is assumed to be NetBSD executable, it will also load symbol tables as the bootloader does to the kernel.)]
- ELF object (Experimental)
- Human68k `.x` executable (Experimental)

LUNA では、設定の `prom-image`、`luna-dipsw1` の内容(後述)によらず内蔵の互換 PROM で起動し、直ちにホストファイルを実行します。X68030 では、IPLROM 起動の後の起動デバイスに細工がしてあり、そこからホストファイルを実行します。[On LUNA, regardless of `prom-image` and `luna-dipsw1` configuration (see below), the internal emulated PROM loads and executes the host file immediately. On X68030, nono hacks the boot device after normal boot from IPLROM, and loads and executes the host file.]

以下開発用。[For developers:]

-b *hexaddr[,skipcount]*

デバッグのブレークポイントを 16進数で指定します。

-C

ログをコンソールにも出力します。通常はログウィンドウにだけ出力されます。

-d

起動時にデバッグプロンプトで停止します。

-D

コンソールをデバッグとして使用します。過去との互換性のために存在していますが、`-V debugger-driver=stdio` と等価です。

-L *name1=level1[,name2=level2,...]*

ログレベルを指定します。カンマで区切って複数指定することも出来ます。 `-Lhelp` で `name` の一覧を表示します。

--load-only *file*

`-x` オプションと同様ですが、PROM のプロンプトで停止します。

-M *name[,name2,...]*

起動時に表示するモニタウィンドウを指定します。カンマで区切って複数指定することも出来ます。 `-Mhelp` で `name` の一覧を表示します。

▼ 3. 設定 [Configuration]

`nono` の設定はいずれも以下の順序で適用されます。[nono's configurations are always applied in the following order.]

1. デフォルト値 [Default value]
2. `~/nono.cfg` があればその内容 [Contents of `~/nono.cfg` if exists]
3. VM ディレクトリ内の `nono.cfg` (または `-c` で指定したファイル) があればその内容 [Contents of `nono.cfg` in the VM directory (or the file specified by `-c` option) if exists]
4. コマンドラインオプション [Command line option]

ファイルの書式はどちらも `key = value` 形式で1行1項目ずつです。 `key` と `value` の前後の空白は取り除かれます。また空行と `"#"` で始まる行は無視します。知らないキーは警告を出した上で無視します。同じキーが複数回現れた場合、上に列挙した順に後から書いたほうで上書きし、同じファイル内でも同様に後に書いたほうで上書きします。コマンドラインオプション `-v` 等はこれをさらに上書きします。コマンドラインオプションで同じキーが複数回現れた場合も後に書いたほうが上書きします。[The syntax of both files is `key = value` format, one per line. White spaces before and after `key` and `value` are ignored. Blank lines, lines beginning with `"#"`

are also ignored. The lines with unrecognized key are ignored with a warning. If the same key appears more than once, the latter overwrites the former in the above order. If the same key appears in a file, the latter overwrites the former in the same manner. Then, command line option `-v` etc. overwrites it. If the same key appears more than once in the command line option, do in the same manner.]

`vmtpe` を除くすべての設定項目はそれぞれデフォルト値を持っています。つまり少なくとも `vmtpe` だけは設定ファイルかコマンドラインオプションで指定する必要があります。[All configuration items except `vmtpe` have default value. It means, you must specify at least only `vmtpe` by configuration file or commandline option.]

設定項目は次の通りです。[The configuration items are:]

`vmtpe = string`

VM 種別を以下のいずれかから指定します。省略不可です。[Specifies the VM type from the following. This field is mandatory.]

```
luna ... LUNA-I
luna88k... LUNA-88K
x68030 ... X68030 (Experimental)
news ... NWS-1750 (Just a joke)
```

`cgrom-image = path`

(X68030 Only) X68030 の外部 CGROM イメージファイルのパスを指定します。CGROM は 768KB です。 `path` がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。 `path` が相対パスなら VM ディレクトリからの相対パスになります (現在のディレクトリからではありません)。空にすると nono 内蔵の互換 CGROM を使用します。デフォルトは空です。[Specifies the X68030's external CGROM image file path. This CGROM is 768KB. If the `path` does not have any path delimiters, the VM directory and then its parent directory will be searched. If the `path` is a relative path, it will be path from the VM directory, not from the current directory. If the `path` is empty, nono's builtin compatible CGROM will be used. The default value is empty.]

`clock-sync = value`

仮想マシン内の時刻の同期方法を指定します。 `real` なら実時間に同期、 `virtual` なら仮想時間に同期します。デフォルトは `real` です。この機能は実験中のため将来予告なく仕様変更になる可能性があります。[Specifies how to synchronize the time in virtual machine. If `real`, synchronize with the real time; if `virtual`, synchronize with the virtual time. The default is `real`. This feature is under experimentation and may be changed in the future without notice.]

`debugger-driver = string`

デバッガのコンソールドライバを指定します。 `stdio`、 `tcp`、 `none` が選択可能です。 `stdio` は標準入出力を使用します。 `tcp` は TCP ポートで TELNET プロトコルで待ち受けます。 `none` ならホスト側とは一切通信を行いません。デフォルトは `none` です。[Specifies console driver of the debugger. `stdio`, `tcp`, and `none` can be specified. `stdio` uses the standard input/output. `tcp` listens on TCP port using TELNET protocol. `none` doesn't make any communication with the host. The default is `none`.]

`debugger-tcp-port = integer`

デバッガのコンソールドライバが `tcp` の時の TCP 待ち受けポート番号を指定します。[Specifies the TCP port number that debugger console driver listens.]

`dipsw-autoboot = yesno`

(LUNA and NEWS Only) 機種に依存せず DIPSW を自動起動に設定するかどうか指定します。 "yes" なら自動起動するように、 "no" なら自動起動しないように DIPSW 設定を上書きします。 "" (空) なら何もしません。デフォルトは "" です。このオプションは `luna-dipsw1` や `news-dipsw` の状態が設定ファイルやコマンドラインオプションによって確定した後に該当のスイッチだけを変更します。[Specifies whether to configure DIPSW to boot automatically, regardless of the models. "yes" overwrites DIPSW, to boot automatically. "no" overwrites DIPSW, not to boot automatically. "" (Empty) does nothing. The default value is "". This option only changes the appropriate switch(es) after the `luna-dipsw1` OR `news-dipsw` configuration is determined by the configuration file or command line options.]

`dipsw-serial = yesno`

(LUNA and NEWS Only) 機種に依存せず DIPSW をシリアルコンソールを使うかどうか指定します。 "yes" ならシリアルコンソールを使うように、 "no" ならシリアルコンソールを使わないように DIPSW 設定を上書きします。 "" (空) なら何もしません。デフォルトは "" です。 `luna-dipsw1` もしくは `news-dipsw` (機種による) の状態が設定ファイルやコマンドラインオプションによって確定した後にこのオプションによって該当のスイッチだけを変更します。 NEWS でシリアルコンソールを使わないを選択した場合、 SW1,2,3 は OFF, OFF, ON にセットされ、 NWB-512 モノクロコンソール(未実装) が選択されます。[Specifies whether to configure DIPSW to use serial console, regardless of the models. "yes" overwrites DIPSW, to use serial console. "no" overwrites DIPSW, not to use serial console. "" (Empty) does nothing. The default value is "". This option only changes the appropriate switch(es) after the `luna-dipsw1` OR `news-dipsw` configuration is determined by the configuration file or command line options. If you choose not to use serial console on NEWS, SW1,2,3 will be set OFF, OFF, ON respectively, and NWB-512 monochrome console (not implemented) will be chosen.]

`ethernet-macaddr = string`

(LUNA and NEWS only) イーサネットデバイスの仮想マシン側の MAC アドレスを指定します。 `xx:xx:xx:xx:xx:xx` 形式で指定します。 `auto` なら自動的に決定します。デフォルトは `auto` です。

`extram-size = integer`

(X68030 Only) 拡張メモリのサイズを MB 単位で指定します。今の所以下だけが指定できます。[Specifies the extended RAM size in MB. For now, only the following can be specified.]

- 0 ... 拡張メモリを使用しません。デフォルトです。[No extended memory. It's default.]
- 16 ... TS-6BE16 互換モードで、アドレス \$0100'0000~\$01ff'ffff の 16MB です。[TS-6BE16 compatible mode. Its address is \$0100'0000~\$01ff'ffff and the size is 16MB.]
- 128/256 ... 060turbo 互換モードで、アドレス \$1000'0000~\$17ff'ffff/\$1fff'ffff の 128/256MB です。[060turbo compatible mode. Its address is \$1000'0000~\$17ff'ffff/\$1fff'ffff and the size is 128 or 256MB.]

`fast-mode = integer`

起動時の動作モードを指定します。0 なら通常モード、1 なら高速モードです。デフォルトは 0 です。コマンドラインオプション `-f` でも高速モードへのみ指定可能です。

`fd-drive = integer`

(X68030 Only) フロッピードライブの数を 0 から 4 で指定します。デフォルトは 2 です。 [Specifies the number of floppy drives from 0 to 4. The default is 2.]

`fdN-image = path`

(X68030 Only) フロッピードライブ *N* に起動時に挿入するディスクイメージを指定します。イメージパスが相対パスなら VM ディレクトリからの相対パスになります。起動後はメニューから操作できます。デフォルトは空です。 [Specifies a diskimage to be inserted to floppy drive *N* at startup. If the *path* is relative path, it is from the VM directory. You can also operate them from menu after startup. The default is empty.]

`fpu-type = value`

(X68030 Only) FPU を装着するかどうか指定しています。none なら FPU を装着しません。68881 なら FPU として 68881 を装着します。現状 68882 はサポートしていません。また none の代わりに 0、68881 の代わりに 1 と書くこともできます。デフォルトは 68881 です。LUNA-I はたぶん 68881 搭載モデルのみのため設定できません。 [Specifies whether to install FPU or not. If none, FPU is not installed. If 68881, 68881 FPU is installed. 68882 is not supported yet. And, none can also be written as 0, 68881 can also be written as 1. The default is 68881. This item cannot be specified on LUNA-I because (probably) all LUNA-I has 68881.]

`hostcom-driver = string`

シリアルポートのホスト側ドライバを指定します。stdio、tcp、none が選択可能です。stdio は標準入出力を使用します。tcp は TCP ポートで TELNET プロトコルで待ち受けします。none ならホスト側とは一切通信を行いません。デフォルトは none です。 [Specifies the host driver of serial port. stdio, tcp, and none can be specified. stdio uses the standard input/output. tcp listens on TCP port using TELNET protocol. none doesn't make any communication with the host. The default is none.]

`hostcom-tcp-port = integer`

ホストドライバが tcp の時の TCP 待ち受けポート番号を指定します。 [Specifies the TCP port number that host driver listens.]

`hostcom-fallback = integer`

`hostcom-driver` で指定したドライバが使用可能でなかった時、0 ならプロセスを終了します。1 なら none を選択して実行を続けます。デフォルトは 0 です。 [Specifies the behavior when the driver which is selected by `hostcom-driver` is unusable; terminate the process if 0, or continue to run using the none driver if 1. The default is 0.]

`hostkbd-input = string`

(LUNA and X68030 only) ホストキーボードの入力モードを指定します。char ならキャラクタ入力モード、jp なら日本語キーボードモードです。デフォルトは char です。詳細は [VM について](#) の章を参照してください。 [Specifies the input mode of the host keyboard. char means the character mode, jp means the Japanese keyboard mode. The default is char. See [About VM](#) for details.]

`hostnet-driver = string`

(LUNA and NEWS Only) イーサネットデバイスのホスト側ドライバを指定します。afpacket、bpf、tap のうちホスト OS がサポートしているものと auto、none が選択可能です。afpacket は Linux の AF_PACKET ソケットを使用します。bpf は bpf(4) デバイスを使用します。tap は tap(4) インタフェースを使用します。none ならホスト側とは一切通信を行いません。auto ならホスト OS がサポートしているもののうち tap → afpacket → bpf を順に試します。デフォルトは auto です。 [Specifies the host driver of ethernet device. afpacket, bpf, and tap can be specified only if the host OS supports them. auto and none can always be specified. afpacket uses Linux's AF_PACKET socket, bpf uses bpf(4) device, and tap uses tap(4) interface. none doesn't make any communication with the host. If auto is specified, it tries tap, afpacket, and bpf in that order (if the host OS supports them). The default is auto.]

`hostnet-afpacket-iframe = iframe`

(LUNA and NEWS Only) ホストドライバが afpacket の時にバインドするインタフェースを1つ指定します。iframe が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ません。 [Specifies an interface name to bind to, when the host driver is afpacket. If auto is specified as *iframe*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

`hostnet-bpf-iframe = iframe`

(LUNA and NEWS Only) ホストドライバが bpf の時にバインドするインタフェースを1つ指定します。iframe が auto なら使用可能なインタフェースを1つ自動的に選択します。デフォルトは auto です。なお、このホストドライバではローカルホストとの通信は出来ません。 [Specifies an interface name to bind to, when the host driver is bpf. If auto is specified as *iframe*, it selects an usable interface automatically. The default is auto. Note that this host driver cannot communicate with the localhost.]

`hostnet-tap-devpath = path`

(LUNA and NEWS Only) ホストドライバが tap の時に使用するデバイスを1つフルパスで指定します。デフォルトは auto で、この場合は自動的にデバイスを選択します。この時の探し方はホスト OS によって異なります。Linux なら /dev/net/tun を指定したのと同じです。OpenBSD なら /dev/tap0 から /dev/tap9 まで順番にオープンできるまで試します。NetBSD (と FreeBSD) ならまず /dev/tap でクローニングを試み、それが失敗すれば /dev/tap0 から /dev/tap9 までを順番にオープンできるまで試します。いずれの場合も VM ディレクトリかその親ディレクトリに nono-ifup、nono-ifdown という名前のシェルスクリプトが必要です。何もすることがない場合でも正常終了する空のシェルスクリプトを用意してください。 [Specifies a device pathname, when the host driver is tap. The default is auto. The behavior in this case depends on the host OS. On Linux, it's the same as /dev/net/tun. On OpenBSD, it will try from /dev/tap0 to /dev/tap9 until successful. On NetBSD (and FreeBSD), it will try cloning by /dev/tap first. If that fails, then try from /dev/tap0 to /dev/tap9 until successful. In all cases, you need to prepare two shell script files which names are nono-ifup and nono-ifdown in the VM directory or its parent directory. Even if you don't have anything to do in these scripts, you need to prepare empty scripts that will terminate

successfully.]

hostnet-fallback = *integer*

(LUNA and NEWS Only) hostnet-driver で指定したドライバが使用可能でなかった時、0 ならプロセスを終了します。1 なら none を選択して実行を継続します。デフォルトは 0 です。[Specifies the behavior when the driver which is selected by hostnet-driver is unusable; terminate the process if 0, or continue to run using the none driver if 1. The default is 0.]

iplrom1-image = *path*

(X68030 Only) X68030 の外部 IPLROM イメージファイルのパスを指定します。こちらは 0xfc0000..0xfffff の 128KB の部分で、IPLROM30 と呼ばれているほうです。path がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。path が相対パスなら VM ディレクトリからの相対パスになります (現在のディレクトリからではありません)。空にすると無償配布されている IPLROM30 を使用します。デフォルトは空です。[Specifies the X68030's external IPLROM image file path. This one contains 128KB at 0xfc0000..0xfffff and is known as IPLROM30. If the path does not have any path delimiters, the VM directory and then its parent directory will be searched. If the path is a relative path, it will be path from the VM directory, not from the current directory. If the path is empty, the public released IPLROM30 will be used. The default value is empty.]

iplrom2-image = *path*

(X68030 Only) X68030 の外部 IPLROM イメージファイルのパスを指定します。こちらは 0xfc0000..0xfdffff の 128KB の部分で、ROM30 と呼ばれているほうです。path がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。path が相対パスなら VM ディレクトリからの相対パスになります (現在のディレクトリからではありません)。空にすると内蔵 ROM を使用します。内蔵 ROM は NetBSD/x68k が起動する程度の SCSI IOCS のみサポートしています。デフォルトは空です。[Specifies the X68030's external IPLROM image file path. This one contains 128KB at 0xfc0000..0xfdffff and is known as ROM30. If the path does not have any path delimiters, the VM directory and then its parent directory will be searched. If the path is a relative path, it will be path from the VM directory, not from the current directory. If the path is empty, internal emulated ROM will be used. The ROM supports only enough SCSI IOCS to boot NetBSD/x68k. The default value is empty.]

keyboard-connect = *integer*

(LUNA and X68030 Only) 起動時にキーボードを本体に接続するかどうかを指定します。1 なら接続し、0 なら接続しません。デフォルトは 1 です。起動後はメニューから変更可能です。[Specifies whether to connect keyboard on boot. If 1, it is connected; if 0, it isn't connected. The default is 1. You can change it on GUI menu after boot.]

luna-adjust-misused-epoch = *integer*

(LUNA Only) LUNA で誤った RTC epoch を採用している OS 向けに RTC エミュレーションを補正するかどうかを指定します。0 なら補正をしません(実機と同じ動作)、1 なら補正します(現実世界と同じ動作)。デフォルトは 1 で、通常 1 のままで使用して問題ありません。[Specifies whether nono corrects RTC emulation for OSes that adopts wrong RTC epoch on LUNA. 0 means making no correction (this is the same behavior as the actual machine). 1 means making correction (this is the same behavior as the real world). The default is 1. Normally, leave it 1.]

LUNA で採用している RTC (MK48T02) は2桁で保持している年の値が 4 で割り切れる年をうるう年とする仕様です。ところが NetBSD/luna68k、OpenBSD/luna88k などはこの RTC の年の値を 1970年からの経過年として使用しています。例えば1970年のはうるう年ではないため 2月28日の翌日は 3月1日ですが、MK48T02 的には 00年であるためうるう年と認識し 2月28日の翌日が2月29日になります。このように、これらの OS を使っている場合実機の RTC は4年のうち約2年間、1日ずれた日付を指しているようです。しかしながら、RTC の時刻は OS 起動時に一度読んだ後は基本的に参照しない上、今時必要な NTP で時間を合わせるため、実機でも問題が顕在化することはまずないと思います。nono の場合は実機と異なり、アプリケーション実行中しか RTC が進まないため、補正がない場合の動作が問題になるのは nono を起動したまま偶数年の2月末日から日付をまたいで、かつ nono を起動したまま OS を再起動して NTP など時刻修正を行わなかった時だけだと思います。このオプションはほぼ開発者向けの動作確認用です。

luna-dipsw1 = *string*

(LUNA Only) 本体前面 DIPSW#1-1..#1-8 の内容を指定します。"0" を DOWN、"1" を UP として、これを8つ並べた形式で、前から順に #1..#8 に対応します。[Specifies status of the front panel DIPSW#1-1..#1-8 using 8 digits. "0" means DOWN and "1" means UP. The first character corresponds to #1 and the eighth character corresponds to #8.]

LUNA-I でのデフォルトは 11110111 です。各スイッチの内容は以下のリンクを参照してください。[On LUNA-I, the default value is 11110111. See the following link about DIPSW.]

→ [NetBSD/luna68k: Information](#)

LUNA-88K でのデフォルトは 11111111 です。各スイッチの内容は以下のリンクを参照してください。[On LUNA-88K, the default value is 11111111. See the following link about DIPSW.]

→ [OpenBSD manual pages: boot_luna88k\(8\)](#)

自動起動するかどうかとシリアルコンソールを使うかどうかを指定したい場合はこちらではなく [dipsw-autoboot](#)、[dipsw-serial](#) 設定を使うほうが便利です。[If you only want to specify whether to autoboot and/or to use serial console, [dipsw-autoboot](#), [dipsw-serial](#) configurations are useful.]

luna-dipsw2 = *string*

(LUNA Only) 本体前面 DIPSW#2-1..#2-8 の内容を指定します。書式は luna-dipsw1 と同じです。デフォルトは 11111111 です。[Specifies status of the front panel DIPSW#2-1..#2-8. The same syntax as luna-dipsw1 is used. The default value is 11111111.]

NetBSD/luna68k のブートローダは、DIPSW#2 が "11111111" なら自動的にカーネルをロードして実行し、どれかでも "0" にするとプロンプトで停止するようです。(本当は #8 だけで制御するつもりだったんじゃないかという気がします)

[NetBSD/luna68k bootloader will automatically load and execute the kernel, if the DIPSW#2 is "11111111".

Otherwise, the bootloader will enter interactive mode. (I doubt that they actually wanted to switch with only #8)]

luna-video-plane = *integer*

(LUNA Only) LUNA のビデオボードのプレーン数を 1、4、8 から指定します。1 ならモノクロビデオボード、4 なら16色 (4bpp) ビデオボード、8 なら 256色 (8bpp) ビデオボードです。デフォルトは 4 です。[Specifies number of planes on LUNA video board. The valid values are 1, 4 or 8. 1 means a monochrome video board, 4 means 16-color (4bpp)

video board, 8 means 256-color (8bpp) video board. The default value is 4.]

mainview-scale = *double*

メインウィンドウの起動時のスケールを実数で指定します。コマンドラインオプション `-s` で上書き可能です。起動後はメニューから、プリセットされた倍率とここで指定した倍率には変更可能です。デフォルトは 1.0 です。

monitor-fontsize = *integer*

テキスト系モニタとステータスパネルのフォントサイズを 12, 16, 24 から指定します。コマンドラインオプション `--fontsize` で上書き可能です。起動後はメニューから変更することができます。デフォルトは 12 です。

monitor-rate = *integer*

テキスト系モニタウィンドウの更新頻度を Hz 単位で指定します。1 から 60 までの間で指定でき、デフォルトは 20Hz です。起動後にメニューから、プリセットされた頻度とここで指定した頻度には変更可能です。[Specifies refresh rate of all text monitor windows in Hz. It ranges from 1 to 60. The default is 20Hz. You can choose this value on GUI menu after boot, from preset rate or rate specified here.]

mpu-clock = *value*

MPU のクロック数を MHz 単位で指定します。デフォルトは LUNA-I なら 20MHz、LUNA-88K と X68030 なら 25MHz です。[Specifies the MPU clock in MHz. The default value is 20MHz on LUNA-I, or 25MHz on LUNA-88K and X68030.]

mpu-pseudo-stop = *integer*

(LUNA-88K Only) m88100 にて疑似 STOP 状態を有効にするかどうかを指定します。0 なら無効(実機と同じ動作)、1 なら有効で、デフォルトは 1 です。m88100 には、m68k の STOP 命令(割り込みが上がるまで何もせず待つ)に相当する命令がなく、大抵ビジーウェイトループで割り込みが上がるのを待つこととなります。これは実機では(消費電力を減らす手段がないという些細な問題以外には)何のデメリットもないのですが、エミュレータで特に高速動作させている時には割り込みが上がるまで(例えば人間がキーを入力するまで)ホスト CPU パワーを使い潰してビジーウェイトループを実行し続けることになり、ホスト CPU があつあつになります。それを防ぐための機能です。特徴的な命令列を検出して実現しているので、すべての状況で動作するわけではありません。

news-dipsw = *string*

(NEWS Only) DIPSW の内容を指定します。"0" を OFF、"1" を ON として、これを 8 つ並べた形式で、前から順に SW1..SW8 に対応します。デフォルトは 00001000 です。[Specifies status of the DIPSW using 8 digits. "0" means OFF and "1" is ON. The first character corresponds to SW1 and the eighth character corresponds to SW8. The default value is 00001000.]

各スイッチの内容は以下のリンクを参照してください。[See the following link about DIPSW.]

→ [NetBSD/news68k Frequently Asked Questions](https://www.netbsd.org/docs/news/68k-frequently-asked-questions/)

自動起動するかどうかとシリアルコンソールを使うかどうかを指定したい場合はこちらではなく [dipsw-autoboot](#)、[dipsw-serial](#) 設定を使うほうが便利です。[If you only want to specify whether to autoboot and/or to use serial console, [dipsw-autoboot](#), [dipsw-serial](#) configurations are useful.]

prom-image = *path*

(LUNA Only) LUNA-I/LUNA-88K の外部 ROM イメージファイルのパスを指定します。path がファイル名のみなら VM ディレクトリとその親ディレクトリからこのファイル名を検索します。path が相対パスなら VM ディレクトリからの相対パスになります(現在のディレクトリからではありません)。空にすると内蔵 ROM を使用します。デフォルトは空です。[Specifies the LUNA-I/LUNA-88K's external ROM image file path. If the path does not have any path delimiters, the VM directory and then its parent directory will be searched. If the path is a relative path, it will be path from the VM directory, not from the current directory. If the path is empty, internal emulated ROM will be used. The default value is empty.]

実機を持っていない場合はこの値を空に(= デフォルトのままに)しておくと、nono 内蔵のなんちゃって下位互換 ROM で起動します。[If you does not have the real LUNA machines, you can boot with nono's internal downward compatible emulated ROM if you set this field empty (or leave it as the default).]

LUNA-I 実機を持っている場合は ROM ファイルを指定することで実機 ROM で起動できます。ROM ファイルは実機の 0x41000000-0x4101ffff (128KB) を保存したものです。今のところ ROM は V4.22 (Thu Jul 27 11:45:42 1989) のみサポートしています。それ以外については何も分かりません。[If you have the real LUNA-I machine, you can boot with the real ROM specifying the ROM file path. The ROM file is extracted from 0x41000000-0x4101ffff (128KB) of the real LUNA-I machine. For now, only V4.22 (Thu Jul 27 11:45:42 1989) is supported. I have no idea about other ROMs.]

LUNA-88K 実機の場合は 0x41000000-0x4103ffff (256KB) を保存したものです。今のところ ROM は version 1.20 のみサポートしています。[For LUNA-88K, the ROM file is extracted from 0x41000000-0x4103ffff (256KB). For now, only version 1.20 is supported.]

ram-size = *integer*

搭載する RAM サイズを MB 単位で指定します。[Specifies the RAM size in MB.]

- LUNA-I のデフォルトは 16MB です。16MB 未満は 4MB 単位で、16MB 以上は 255MB まで 1MB 単位で指定できます(ちなみに NetBSD/luna68k の起動には最低でも 8MB 必要です)。[On LUNA-I, the default is 16MB. If the size is less than 16MB, you can specify in 4MB unit. If larger, you can specify up to 255MB in 1MB unit. By the way, NetBSD/luna68k needs at least 8MB to boot.]
- LUNA-88K のデフォルトは 64MB です。64MB 未満は 16MB 単位で、64MB 以上は暫定で 255MB まで 1MB 単位で指定できます。[On LUNA-88K, the default is 64MB. If the size is less than 64MB, you can specify in 16MB unit. If larger, you can specify up to tentative 255MB in 1MB unit.]
- X68030 ではメイン RAM 容量を示し、デフォルトは 12MB です。4MB から 12MB まで 1MB 単位で指定できます。拡張メモリについては `extram-size` を参照してください。[On X68030, the default is 12MB. You can specify it in 1MB unit from 4MB to 12MB. See `extram-size` for extended memory.]
- NWS-1750 では現状 16MB で変更できません。[On NWS-1750, This is 16MB fixed for now.]

`rtc-epoch-year = integer`

(LUNA Only) RTC (MK48T02) の基準年を指定します。デフォルトは 1970 年です。NetBSD/luna68k、OpenBSD/luna88k はいずれも基準年を 1970 年としていますので、デフォルトのまま構いません。ゲスト OS に 4.4BSD を使用する際は 1900 を指定します。NEWS では今の所 1900 固定で変更出来ません。[Specifies the RTC (MK48T02) epoch year. The default is 1970. Since both NetBSD/luna68k and OpenBSD/luna88k use 1970 as epoch year, there is no need to touch this value. If you boot 4.4BSD on LUNA-I as the guest OS, specify 1900. On NEWS, it is fixed at 1900.]

`show-statuspanel = integer`

ステータスパネルを表示するかどうかを指定します。0 なら非表示、1 なら表示です。デフォルトは 1(表示) です。起動後はメニューから変更可能です。[Specifies whether to display the status panel or not. If 0, it is hidden; if 1, it is shown. The default is 1 (show). You can change it on GUI menu after boot.]

`spc0-idN-image = devtype[,path]`

(LUNA and X68030 Only) SCSI デバイスとイメージを指定します。キーの *N* には 0 から 7 が入ります。ただし ID 7 は本体が使用しますので指定しないでください。値はデバイス種別 *devtype* とディスクイメージパス *path* を ","(カンマ) で区切って並べた形式です。デバイス種別 *devtype* は以下のいずれかです。[Specifies SCSI device and image. *N* in the key is 0 to 7. But don't specify ID 7 because the host uses it. The value is in a form of device type *devtype* and the disk image path *path* separated by ","(comma). *devtype* can be one of the following:]

- hd ... HD drive
- cd ... CD-ROM drive
- mo ... MO drive

devtype が hd なら *path* は省略できません。 *devtype* が cd か mo なら *path* は省略可能です。イメージパスが相対パスなら VM ディレクトリからの相対パスになります。[If *devtype* is hd, *path* cannot be omitted. If *devtype* is cd or mo, *path* can be omitted. If the *path* is relative path, it is from the VM directory.]

例えば、nono.cfg と同じディレクトリに置いた sd0.img を起動 HDD ディスクイメージとして使い (LUNA では通常 ID 6 をプライマリ HDD に割り当てます)、ID 5 に同じディレクトリの install.iso をセットした CD ドライブを、ID 4 に起動時メディアなしの MO ドライブを接続する場合は次のようになります。[For example, if you use a harddisk image sd0.img placed in the same directory as nono.cfg (LUNA usually assigns ID 6 to the primary HDD), ID 5 for CD-ROM drive that loads install.iso in the same directory, and ID 4 for MO drive without media on boot, write as following:]

```
spc0-id6-image = hd,sd0.img
spc0-id5-image = cd,install.iso
spc0-id4-image = mo
```

`spc0-idN-seektime = integer`

(LUNA and X68030 Only) 指定の SCSI HDD の平均シークタイムを msec 単位で指定します。現在のデフォルトは 0 です (S・S・D!! S・S・D!!)。16 程度を指定すると幾分往時に思いを馳せることが出来るかもしれませんが、今の所あまり安定していません。[Specifies the average seek time of specified SCSI HDD in msec. Currently, the default value is 0 (This may be something like SSD :-). If you specify about 16 or so, you can feel nostalgic, but this feature is still unstable.]

`spc0-idN-writeignore = integer`

(LUNA and X68030 Only) 指定の SCSI HD デバイスへの書き込みをイメージに書き戻すかどうか指定します。0 なら通常動作 (書き込みを行う) です。1 ならディスクイメージに一切書き戻しません。デバイスへ書き込んだはずのデータは Copy-On-Write の要領で、VM ディレクトリ内の一時ファイルに書き込みます。この一時ファイルはアプリケーション終了時に削除します。fsck を気にせずカーネルのデバッグとかを行いたい場合にはどうぞ。何が起きるか意味が分からない人は指定しないでください。デフォルトは 0 です。[Specifies whether nono writes a writing to SCSI HD devices back to the diskimage. 0 means normal operation (writes to the diskimage). 1 means that nono will not write to the diskimage back. Data that the guest wrote to the device will be written to a temporary file in the VM directory, like Copy-On-Write. The temporary file will be removed when the application exits. This is useful for kernel debugging because it does not require fsck after the kernel hangs or reboots. Don't use this flag if you don't understand this paragraph. The default value is 0.]

ちなみに、メディアを書き込み禁止にしたい場合はこれではなく、イメージファイルの書き込み権を落としてください。[By the way, if you want to make the media write-protected, clear the write permission from the image file (instead of this setting).]

`spc0-idN-writeprotect = integer`

(LUNA and X68030 Only) 古いオプションです。代わりに `spc0-idN-writeignore` を使ってください。このオプションは開発用です。[Obsolete. Use `spc0-idN-writeignore` instead. This option is for developers.]

`sram-sync-ramsize = integer`

(X68030 Only) SRAM の RAM 容量欄を実際の RAM サイズに合わせるかどうかを指定します。0 なら何もしません (実機と同じ)。1 なら、アプリケーション起動時に SRAM の \$ed0008.L (RAM 容量) を ram-size の設定値で更新します。この更新は、アプリケーション起動時に SRAM マジックが正しく書かれている場合のみ行われます。デフォルトは 1 です。[Specifies whether to synchronize a RAM size field in SRAM with real RAM size or not. If 0, the application will do nothing (as same as the real). If 1, the application updates(synchronizes) \$ed0008.L (RAM size) in SRAM with configuration value of ram-size. This update is performed once only if the SRAM magic string is correct at the application startup. The default is 1.]

`xxx-news-sic-ignore = integer`

(NEWS Only) 未実装の SCSI コントローラおよび DMA コントローラ領域へのアクセスを無視するか指定します。現状 (2022/08)、NetBSD/m68k はこの領域へのバスエラーを処理できず、uvm_fault のパニックを起こしてしまいます。1 にするととりあえず si(4) へのアクセスが不調に終わって (他に問題がなければ) NetBSD/news68k が起動できます。デフォルトは 0 です。このオプションは暫定的なもので将来廃止される予定です。[Specifies whether to ignore accesses to unimplemented SCSI controller and DMA controller areas. As of Aug. 2022, NetBSD/m68k can not handle bus errors to this area, and it happens uvm_fault panic. Specifying 1 avoids it and NetBSD/news68k can boot. The default value is 0. This option is temporary and will be discontinued in the future.]

▼ 4. VM について [About VM]

▼ 4.1. 実装状況 [Implementation Status]

- ◎ : ほぼ実装済み [Mostly implemented]
- : 困らない程度には実装済み [Works]
- △ : とりあえず動く程度 [Works somehow]
- × : 未実装 [Not implemented]
- : 実装予定なし [Not planned]

LUNA-I

MPU (68030)	○	キャッシュ未実装 [Cache not implemented]
FPU (68881)	◎	
BT454	◎	
CRTC2	○	標準で使用する値のみサポート
Frame buffer	◎	
Keyboard/Mouse	◎	
SIO (uPD7201)	◎	
PIO (8255)	○	
Ethernet (AM7990)	◎	LARF を除く [Except LARF]
NVRAM&RTC (MK48T02)	◎	
Front LCD	◎	
SPC (MB89352)	◎	
SCSI HD/CD/MO Devices	○	
XP(HD647180) CPU	△	
XP(HD647180) Timer0/1	○	
XP(HD647180) その他の内蔵デバイス [Other internal devices]	×	
XP 周辺デバイス [Peripheral devices]	×	
電源 [Power Circuit]	○	

LUNA-88K specific

MPU (88100 part)	○	マルチプロセッサ未実装、バグも未実装? [MP (and bugs?) not implemented]
MPU (88200 part)	◎	
MPU (88110)	—	実装予定なし [Not planned]
FUSEROM	○	

X68030

MPU (68030)	○	キャッシュ未実装、ファンクションコード未実装 [Cache and FC not implemented]
MPU (68EC030)	—	実装予定なし [Not planned]
FPU (68881)	◎	68882 は未実装 [68882 not implemented]
DMAC	○	
CRTC/VC	△	768x512 のテキスト画面のみサポート [Only 768x512 text screen supported]
FDC/FDD	○	
標準・拡張エリアセット [Areaset]	×	
ADPCM/OPM	×	
グラフィック機能全般 [All other graphics]	×	当面予定なし [Not scheduled]
Keyboard	◎	TVコントロールを除く [Except TV Control]
Mouse	◎	本体接続のみ
MFP	△	
PPI (8255)	△	
Printer	×	
RTC	◎	アラームを除く [Except Alarm]
SPC (MB89352)	◎	
SCSI HD/CD/MO Devices	○	
SCC (Z8530)	◎	
SRAM	◎	
システムポート[System port]	△	
テキスト画面[Text screen]	○	
電源 [Power circuit]	○	
Nereid Ethernet	×	
Nereid Bank memory	×	
Nereid USB	—	実装予定なし [Not planned]
拡張メモリ [Extended memory]	○	

NWS-1750

MPU (68030)	○	キャッシュ未実装 [Cache not implemented]
FPU (68881)	◎	68882 は未実装 [68882 not implemented]
PROM	×	
SCC (Z8530)	◎	チャンネル A のみホストに接続 [Only channel A is connected to the host]
Ethernet (AM7990)	◎	LARF を除く [Except LARF]
NVRAM&RTC (MK48T02)	◎	
Video Board	×	
Keyboard/Mouse	×	
SCSI Controller	×	

IDROM, DIP-SW, 100Hz Timer, 診断用 LED	○
AST, Lv2Intr	×
電源 [Power Circuit]	×

▼ 4.2. MPU (68030)

68030 はキャッシュが未実装です。これに伴い命令の所要クロック数もまだ正しくなく、そのためクロック測定などでは正確な値ができません。また ATC (アドレス変換キャッシュ) は実際の仕様とは異なり高速化してあります。 [68030 cache is not implemented. Therefore, number of clock cycle is not correct. ATC (Address Translation Cache) is differ from and faster than the real 68030.]

▼ 4.3. LUNA の ROM エミュレーション

LUNA-I, LUNA-88K で PROM の外部イメージを指定しない場合 (prom-image が空の場合)、内蔵の独自 ROM が動作し、コンソールモニタが起動します。このコンソールモニタの動作は実機のそれとよく似ていますが、これは実機確認の際に操作方法が違いすぎると (自分)困るため仕方なく似せているもので、忠実なコピーを作る目的ではありません。

▼ 4.4. X68030 の ROM (ROM30) エミュレーション

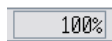
X68030 で ROM30 の外部イメージを指定しない場合 (iplrom2-image が空の場合)、内蔵の独自 ROM が組み込まれます。本来の ROM30 には SCSIIOCS の他に ROM デバッグなどが含まれていますが、内蔵 ROM はこのうち SCSIIOCS を NetBSD/x68k が起動するのに十分なだけしか実装していません。Human68k を起動する場合はお手元の実機から ROM イメージを作成してください。

▼ 4.5. NEWS エミュレーション

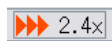
NEWS エミュレーションは未完成です。今の所 -X でカーネルを直接起動することしか出来ません。

▼ 4.6. ステータスパネル [Status Panel]

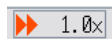
ステータスパネル中央にあるパフォーマンスメータのアイコンは高速モードの状態を表示しています。ダブルクリックすると高速/等速モードの指定を切り替えることができます。 [The performance meter's icon at the center of the status panel shows the VM speed status. You can switch full speed / synchronized mode by double-clicking on this icon.]



... マークなしの場合、ユーザが等速モードを指定していて、等速モードで実行中です。 [When no icons are displayed, the user has specified synchronized mode and the VM is running in synchronized mode.]



... 三角3つの場合、ユーザが高速モードを指定していて、高速モードで実行中です。高速モード中はパーセントではなく何倍速で動作しているかを表します。 [When an icon (three triangles) is displayed, the user has specified full speed mode and the VM is running in full speed mode.]



... 三角2つの場合、ユーザが高速モードを指定していて、等速モードで実行中です。以下のいずれかの状態で起きます。 [When an icon (two triangles) is displayed, the user has specified full speed mode and the VM is running in synchronized mode. This will happen when any of the following occurs:]

- キー押下中(後述)あるいはキーボードブザー発声中 [When any keys are pressed (see below), or the keyboard buzzer is sounding.]
- STOP instruction on m68k.
- pseudo STOP state on m88k.

▼ 4.7. DIPSW 設定ウィンドウ

DIPSW 設定ウィンドウでの変更はアプリケーション起動中のみ有効です。またここでの設定は即 VM に反映されますが、LUNA の場合 DIPSW の状態を PIO(8255) がラッチするのが概ね起動時および再起動時のみとなっています (厳密には PROM と XP が云々ですが)。

▼ 4.8. メモリダンプモニタ [Memory Dump Monitor]

メモリダンプモニタの出力形式を、バイト、ワード、ロングワード (m88k ではそれぞれバイト、ハーフワード、ワードと表現します) から選択できます。また 68030/88200 の MMU ディスクリプタとして表示させることもできます (この位置が正しいディスクリプタかどうかは関係なく表示します)。逆アセンブルも、逆アセンブルウィンドウからこちらにマージしました。LUNA-I なら m68k、HD64180 の逆アセンブルが、LUNA-88K なら m88k、HD64180 の逆アセンブルが、X68030 と NEWS なら m68k の逆アセンブルが選択できます。 [On memory dump monitor, you can choose the output format from byte, word and longword (In m88k, it is represented as byte, halfword and word respectively). Also you can choose 68030/88200 MMU descriptor format (It displays regardless of whether this address is used as descriptor or not). Disassemble is now merged from disassemble monitor. On LUNA-I, you can choose m68k and HD64180 disassemble. On LUNA-88K, you can choose m88k and HD64180 disassemble. On X68030 and NEWS, you can choose m68k disassemble.]

表示されているダンプ値をダブルクリックすることでメモリの値を編集できます。編集できるのはメモリと以下の部分のみです。 [You can edit the memory value by double click. Only memory and the following areas are editable.]

- NVRAM (LUNA and NEWS Only) ... 編集するとただちにファイルも更新します。後ろの8バイトは時計レジスタを兼ねているため編集不可です。 [The backend file is also updated immediately when edited. The last 8 bytes are not editable since there are the clock register.]
- Bitmap framebuffer (LUNA Only) ... 共通プレーンは書き込み不可です。 [The common plane is not editable.]
- SRAM (X68030 Only) ... 編集するとただちにファイルも更新します。 [The backend file is also updated immediately when edited.]
- TVRAM とテキストパレット (X68030 Only) [Text VRAM and text palette]

▼ 4.9. デバイスマップモニタ [Device Map Monitor]

m68k/m88k バスに接続されているデバイスを表示します。32ビットアドレスのうち上位 8 ビット、256個分を表示しています。1枠が 16MB になります。また、X68030 ではこのうち 24ビット空間を別モニタで表示します。こちらは 1枠 8KB x 2048個です。どちらも内部構造を表示しているだけなので名前に馴染みがない場合があるかもしれません。

▼ 4.10. キー入力モード [Key input mode]

LUNA も X68030 もキーボードは通常日本語キーボードです (LUNA には英語キーボードも存在するようですが...)。そのため、ホストが日本語キーボードの場合とそうでない場合のために 2つの入力モードを用意しています。どちらも一長一短があります。 [LUNA

and X68030 has the Japanese keyboard (though I ever heard there are US keyboard for LUNA...). So nono provides two input modes for Japanese keyboard users and non Japanese keyboard users. Both have advantages and disadvantages.]

JP キーボードモードは、ホストが日本語キーボードであることを前提に対応するキーの押下、開放をその都度 VM に入力します。メリットとしては動作が実機に近くなり、特にキーリピート(をハードウェアが行わないこと)も忠実に再現可能です。デメリットは ALT + TAB やアクセラレータキーでメニューを開く場合などにキー入力が残ることがあることです(キーを押した後でフォーカスが外れるとキーを離れたことがアプリケーションに通知されないため)。またホストが日本語キーボードでなければ使いづらいです。[JP keyboard mode assumes that the host uses the Japanese keyboard. Each time you presses and releases a key, nono sends the corresponding key-pressed or key-released to the VM. One advantage is that it's close to the real machine behavior, and another is that it can reproduce the LUNA keyboard itself does not perform a key repeat. On the other hand, one disadvantage is that key sometimes remains pressed, especially when you enter ALT + TAB to switch the application or enter accelerator key to open the menu (If the application loses the focus after the key is pressed, the application will not receive the subsequent key release). And another disadvantage is that it will be hard to use for non Japanese keyboard users.]

一方、キャラクタ入力モードは、ホストから入力された「文字」を再び LUNA/X68030 のキーコードに解釈し直して VM に入力するモードです。例えばゲスト OS に "@" を入力したい場合、US キーボードなら [SHIFT] + [2] を、JP キーボードなら [@] キーを押しますが、どちらも VM には [@] キーの押下、[@] キーの開放が連続して送られます。US キーボードで [SHIFT] を押したことは通知されません。逆に "=" を入力したい場合、US キーボードなら [=] キーを、JP キーボードなら [SHIFT] + [-] を押しますが、どちらも VM には [SHIFT]押下 → [-]押下 → [-]開放 → [SHIFT]開放のシーケンスが送られます。このようにホストキーボードのレイアウトによらず、入力したい文字が入力できるようになり、通常のタイピングで困ることはないと思います。デメリットは、通常の文字入力以外のタイピングはほぼ出来ないことです。[SHIFT] キーを押したままにすることや、文字が入力できないキーの組み合わせは入力出来ません。もう一つのデメリットはキーリピート(をハードウェアが行わないこと)が再現できず、ホスト側のキーリピートが効いてしまうことです(本来キーリピートが動作しない ROM やブートローダでもキーリピートが効くことは一般ユーザからはメリットに感じられるかも知れませんが)。[In contrast, in the character mode, when you type a character, nono re-converts it into the LUNA/X68030's key code and sends it to the VM. For example if you want to type "@" in the guest OS, you would press [SHIFT] + [2] on the US keyboard or [@] on the JP keyboard. In both cases, the sequence of [@]-pressed and [@]-released will be sent to the VM. Note that the fact that you pressed [SHIFT] key on the US keyboard will not be sent to the VM. Another example, if you want to type "=", you would press [=] on the US keyboard or [SHIFT] + [-] on the JP keyboard. In both cases, the sequence of [SHIFT]-pressed, [-]-pressed, [-]-released, and [SHIFT]-released will be sent to the VM. Thus, you can type what character you want to type, regardless of your keyboard layout. Of course there are some disadvantages. Firstly, it's hard to type any special characters, for example, you can not type any key combinations that don't generate a character. Or you can not hold [SHIFT] key pressed. Secondly, the key repeat works (by the host), even though it should not work as a correct emulation (Although some of you may feel it an advantage).]

▼ 4.11. キャラクタ入力モードのキーコード対応表 [Keycode table in the character mode]

キャラクタ入力モードでの、ホストから入力した文字と VM に送信されるキーの対応は次の通りです。ファンクションキーとカーソルキーは文字ではありませんが特別に対応しています。ここに記載のないゲストキーはソフトウェアキーボードからのみ入力できます。

[The following table shows the correspondence between the character that typed from the host and the key that will be sent to the VM. nono can recognize the function keys and arrow keys. The keys that not listed in this table can only be typed from the software keyboard window.]

HostChar	GuestKey
a ~ z	[A] ~ [Z]
0 ~ 9	[0] ~ [9]
SPACE	[SPACE]
-	[-]
^	[^]
\	[\]
@	[@]
[[[]]
;	[;]
:	[:]
]	[[]]
,	[,]
.	[.]
/	[/]

HostChar	GuestKey
A ~ Z	[SHIFT]+[A] ~ [SHIFT]+[Z]
!	[SHIFT]+[1]
"	[SHIFT]+[2]
#	[SHIFT]+[3]
\$	[SHIFT]+[4]
%	[SHIFT]+[5]
&	[SHIFT]+[6]
'	[SHIFT]+[7]
([SHIFT]+[8]
)	[SHIFT]+[9]
=	[SHIFT]+[-]
~	[SHIFT]+[^]
	[SHIFT]+[\]
`	[SHIFT]+[@]
{	[SHIFT]+[[]]
+	[SHIFT]+[;]
*	[SHIFT]+[:]
]	[SHIFT]+[[]]
<	[SHIFT]+[,]
>	[SHIFT]+[.]
?	[SHIFT]+[/]
_	[SHIFT]+[_]

HostChar	GuestKey
F1 ~ F10	[PF1] ~ [PF10]
F11	[PF10]
↑	[↑]
←	[←]
→	[→]
↓	[↓]
^@	[CTRL]+[@]
^A ~ ^Z	[CTRL]+[A] ~ [CTRL]+[Z]
^I	[ESC]
^\	[CTRL]+[\]
^]	[CTRL]+[[]]
^^	[CTRL]+[^]
^_	[CTRL]+[_]
^H	[BS]
^I	[TAB]
^M	[Enter]

▼ 4.12. JP キーボードモードのキーコード対応表 [Keycode table in the JP keyboard mode]

JP キーボードモードでの、ホストキーと VM に送信されるキーの対応は次の通りです。ここに記載のないキーはソフトウェアキーボードからのみ入力出来ます。[The following table shows the correspondence between the host key and the guest key. The keys that not listed in this table can only be enter from the software keyboard window.]

HostKey	GuestKey
[0] ~ [9]	[0] ~ [9]
[A] ~ [Z]	[A] ~ [Z]
[-]	[-]
[^]	[^]
[\]	[\\]
[@]	[@]
[I]	[I]
[;]	[;]
[:]	[:]
[I]	[I]
[,]	[,]
[.]	[.]
[/]	[/]
[_]	[_]

HostKey	GuestKey
[ESC]	[ESC]
[TAB]	[TAB]
[CTRL]	[CTRL]
[BS]	[BS]
[Enter]	[Enter]
[F1] ~ [F10]	[PF1] ~ [PF10]
[F11]	[PF10]
[SHIFT-L]	[SHIFT-L]
[SHIFT-R]	[SHIFT-R]
[↑]	[↑]
[←]	[←]
[→]	[→]
[↓]	[↓]

HostKey	GuestKey
[DEL]	Numpad[DEL]
Numpad[0]~[9]	Numpad[0]~[9]
Numpad[+]	Numpad[+]
Numpad[-]	Numpad[-]
Numpad[*]	Numpad[*]
Numpad[/]	Numpad[/]
Numpad[=]	Numpad[=]
Numpad[.]	Numpad[.]
Numpad[Enter]	Numpad[Enter]

▼ 4.13. キーリピート **[Key Repeat]**

LUNA では、キーボードがハードウェア側でキーリピートを行わず、ソフトウェア(OS など)がキーリピートの処理を行います。そのため、キーリピートを実装していない LUNA の PROM や NetBSD/luna68k のブートローダではキーリピートは起きず、キーリピートを実装している NetBSD/luna68k カーネルではキーリピートが起こせます。キーリピートを起こす間隔をゲストのソフトウェアが測定しているということは、VM が高速動作するとキーリピートもそれに合わせて発生することになり、これをホスト側から防ぐ手段はありません。そこで nono ではキー入力が発生している間 (キーが一つでも押されてからキーが全て離されるまでの間)、高速モードが指示されていても VM を一時的に等速モードに落として実行します。上記のアイコンがそれを区別しているのはこのためです。そのため、何らかの理由でキーが入りっぱなしになった場合 (ALT + TAB やアクセラレータキーでメニューを開くと起きがちです) 高速モードが抑制されたままということが起きえます。その場合はソフトウェアキーボードからキー入力を解除するなどしてください。[On LUNA, key repeat is done by software(OS), not by the keyboard hardware. For this reason, key repeat doesn't occur on LUNA's PROM or NetBSD/luna68k's bootloder that don't implement it, and key repeat occurs on NetBSD/luna68k kernel that implements it. Since the timing of key repeat is measured by the guest software, if the VM is running faster than the real, the key repeat will occur faster, too. The host application doesn't have the way to avoid it. Therefore, nono will temporarily suppress the full speed mode while any keys are pressed. That is why the above-mentioned icon distinguishes them. If keys continue to be pressed for some reasons, the VM also continues to run synchronized mode. In this case, you can resolve it by using the software keyboard window.]

▼ 5. 実行してみる **[Try it]**

▼ 5.1. NetBSD/luna68k を実行してみる **[Try NetBSD/luna68k]**

つついさんが NetBSD/luna68k 9.2 の liveimage を用意されています。[Tsutsui-san has provided a liveimage of NetBSD/luna68k 9.2.]

<https://twitter.com/tsutsui/status/1436381589211017217>

ここではこれを起動してみます。[Let's try it.]

- どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します (例えば ~/nono/luna/)。[Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna/).]
- 以下のリンクからイメージファイルをダウンロードして展開し、VM ディレクトリ ~/nono/luna/ に置きます。[Download imagefile from the following link, extract it and place it in the VM directory, ~/nono/luna/.]

<http://teokurebsd.org/netbsd/liveimage/20210910-luna68k/>

- 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。[Create a configuration file nono.cfg in the VM directory, ~/nono/luna/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna
spc0-id6-image = hd,liveimage-luna68k-with-packages-20210910.img
hostnet-driver = none
```

- nono -c ~/nono/luna で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。[Run as nono -c ~/nono/luna. (NVRAM.DAT will be created automatically in the VM directory)]
- Emulated ROM Monitor が起動するので、初回は以下のように入力すると NetBSD が起動します。[The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot NetBSD.]

```
k#
#
#
d#
boot#
g#
x#
```

画面はこんな感じのはずです (太字が入力部分)。[You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.5.1 Emulated ROM Monitor for LUNA-I

** NVRAM Initialized.

>k#
controller: dk ?#
drive unit: 0 ?#
partition : c ?d#
```

```
filename : vmunix ?boot#
>g#
Loaded.  Entry point = $00700000
>x#
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 NetBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it will boot NetBSD automatically.]

- 初回起動時、Updating fontconfig cache はあほみたいに時間がかかりますが、nono がハングアップしてるわけではありません(>_<)。また初回は起動後数時間めっちゃくちゃ重たいですが、これはバックグラウンドで makemandb が動いたためで nono のせいではありません(>_<)。 [At the first boot, you will see the console stops after printing "Updating fontconfig cache". This is because the infamous fontconfig takes very loooong time. nono would not have hang-up. In addition, at the first boot, you will feel it's too heavy for several hours. This is because the infamous makemandb(8) runs heavily in the background for a loooong time. It's very sad to me that these two accidents which are far from the ideal are the first experiences of newcomers.]
- 終了する時は root ユーザーで "shutdown -p now" を実行してください。LUNA はソフトウェアから電源オフでき、VM の電源オフで nono も終了します。 [To quit, type "shutdown -p now" as the root user. LUNA can be powered off by software, and nono will terminate when the VM is powered off.]

▼ 5.2. OpenBSD/luna88k を実行してみる [Try OpenBSD/luna88k]

あやまさんが OpenBSD/luna88k 7.0 の liveimage を用意されています。 [Aoyama-san has provided a liveimage of OpenBSD/luna88k 7.0.]

https://twitter.com/ao_kenji/status/1473611893386661888

ここではこれを起動してみます。 [Let's try it.]

- どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します(例えば ~/nono/luna88k/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/luna88k/).]
- 以下のリンクから liveimage-luna88k-raw-20211221.img.gz をダウンロードして展開し、VM ディレクトリ ~/nono/luna88k/ に置きます。 [Download liveimage-luna88k-raw-20211221.img.gz from the following link, extract it and place it in the VM directory, ~/nono/luna88k/.]

<http://www.nk-home.net/~aoyama/liveimage/>

- 以下の内容の設定ファイル nono.cfg を VM ディレクトリ ~/nono/luna88k/ に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file nono.cfg in the VM directory, ~/nono/luna88k/, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = luna88k
spc0-id6-image = hd,liveimage-luna88k-raw-20211221.img
hostnet-driver = none
```

- nono -c ~/nono/luna88k で起動します (VM ディレクトリに自動的に NVRAM.DAT が作られます)。 [Run as nono -c ~/nono/luna88k. (NVRAM.DAT will be created automatically in the VM directory)]
- Emulated ROM Monitor が起動するので、初回は以下のように入力すると OpenBSD が起動します。 [The emulated ROM Monitor will be executed. Then, only for the first time, entering the following can boot OpenBSD.]

```
nvramp boot_filename boot#
y
b#
```

画面はこんな感じのはずです (太字が入力部分)。 [You will see a screen like this. The bold text indicates the characters you need to enter.]

```
NONO 0.5.1 Emulated ROM Monitor for LUNA-88K

** NVRAM Initialized.

N>nvramp boot_filename boot#
Update boot_filename : "vmunix" -> "boot" (Y/[N]):y
Updated
N>b#
```

この内容は NVRAM.DAT に記録されているので次回以降は直接 OpenBSD が起動します。 [The information you have just entered is recorded in the NVRAM, so next time it will boot OpenBSD automatically.]

- 終了する時は root ユーザーで "shutdown -p now" を実行してください。LUNA-88K はソフトウェアから電源オフでき、VM の電源オフで nono も終了します。 [To quit, type "shutdown -p now" as the root user. LUNA-88K can be powered off by software, and nono will terminate when the VM is powered off.]

▼ 5.3. NetBSD/x68k を実行してみる [Try NetBSD/x68k]

つっいさんが NetBSD/x68k 9.2 の liveimage を用意されています。ここではこれを起動してみます。 [Tsutsui-san has provided a liveimage of NetBSD/x68k 9.2. Let's try it.]

- どこかに nono 用のディレクトリを用意し(例えば ~/nono/)、その中に VM ディレクトリを用意します(例えば ~/nono/x68k/)。 [Create a directory somewhere for nono (for example ~/nono/), and create subdirectories for individual VMs in it (for example ~/nono/x68k/).]
- 以下のリンクから plain-install か with-packages のお好みのほうをダウンロードして展開し、VM ディレクトリ ~/nono/x68k/ に置きます。 [Download your favorite one of plain-install or with-packages from the following link, extract it and place it in the VM directory, ~/nono/x68k/.]

<http://teokurebsd.org/netbsd/liveimage/20210613-x68k/>

- 以下の内容の設定ファイル `nono.cfg` を VM ディレクトリ `~/nono/x68k/` に作成します。ここでは説明を簡単にするためネットワークなしにしていますが、ネットワーク設定は必要に応じて行ってください。 [Create a configuration file `nono.cfg` in the VM directory, `~/nono/x68k/`, with following contents. By the way, to simplify the explanation, we assume there is no network here. However, please configure the network if necessary.]

```
vmtype = x68030
spc0-id0-image = hd,liveimage-x68k-with-packages-20210613.hds
hostnet-driver = none
```

- 初回のみ `nono -c ~/nono/x68k --create-sram` を実行します。これで VM ディレクトリに初期状態の `SRAM.DAT` が作られます。 [Only the first time, run as `nono -c ~/nono/x68k --create-sram`. This will create initial `SRAM.DAT` in the VM directory.]
- `nono -c ~/nono/x68k` で起動すると、HDD から `NetBSD/x68k` が起動します。次回以降はこれだけで直接 `NetBSD` が起動します。 [Again, run as `nono -c ~/nono/x68k`. `NetBSD/x68k` will be booted from HDD. Next time, it will boot `NetBSD` automatically.]
- 初回起動後数時間はめちゃくちゃ重たいですが、これはバックグラウンドで `makemandb` が動くため `nono` のせいではありません(>_<)。 [At the first boot, you will feel it's too heavy for several hours. This is because the infamous `makemandb(8)` runs heavily in the background for a loooong time. It's very sad to me that an accident which is far from the ideal is the first experiences of newcomers.]
- `NetBSD` が起動した後なら (正確には `powerd(8)` が起動していれば)、終了する時はメニューの「デバイス > 操作 > 電源ボタンを押す」でシャットダウンして電源オフになり、VM の電源オフで `nono` も終了します。 [After `NetBSD` boots up (more precisely, if `powerd(8)` is running), you can shutdown and power off by "Device > Operation > Push Power button" on menu, and `nono` will terminate when the VM is powered off.]

▼ 6. ホストネットワーク設定例 [Example of host network setup]

`wm0` を持つ `NetBSD` ホストに `tap(4)` デバイスを用いて `nono` のゲスト OS を接続する場合の設定例です。

- 設定ファイル `nono.cfg` に以下の行を追加します (と言いつつ `NetBSD` では書かなくてもデフォルトでこの動作になりますが) [Add the following line to configuration file, `nono.cfg`. (Although you don't need to write it since these are default behavior on `NetBSD`)]

```
hostnet-driver = tap
hostnet-tap-devpath = auto
```

- デフォルトでは `/dev/tap` は一般ユーザからアクセスできないので、`chmod` で適当にパーミッションを与えます。番号の付いていないほうの `/dev/tap` だけでいいです。 `sysinst` 等で OS をアップグレードするとパーミッションが `600` に戻るのがハマリポイントです。 [By default, `/dev/tap` is only accessible to privileged user. You need to `chmod /dev/tap` (without unit number) appropriately. Note that upgrading using `sysinst` always reset the permission to `600`.]
- `bridge(4)` インタフェースを作成し、ホストの外部(物理)インタフェースをブリッジに追加しておきます。 [Create a `bridge(4)` interface, and add your physical interface to the bridge.]

```
# ifconfig bridge0 create
# brconfig bridge0 add wm0
```

常用するなら `/etc` の設定ファイルに書いておきましょう。 [If you use it regularly, you can put configuration file into `/etc`.]

```
/etc/ifconfig.bridge0
```

```
create
up
!/sbin/brconfig $int add wm0
# /etc/rc.d/network restart
```

- 一般ユーザに戻って、VM ディレクトリかその親ディレクトリに次のような 2つのスクリプトを用意します。 `nono` は `tap(4)` をオープンし、そのデバイス名を引数にこれらのスクリプトを呼びます。 `sudo` の設定は別途行ってください。 [Return to non-privileged user, and create following two scripts in the VM directory or its parent directory. `nono` will open `tap(4)` and invoke these scripts with the name of the device as an argument. In addition, you need to set up `sudo` separately.]

```
nono-ifup
```

```
#!/bin/sh
sudo /sbin/ifconfig $1 up
sudo /sbin/brconfig bridge0 add $1
```

```
nono-ifdown
```

```
#!/bin/sh
sudo /sbin/brconfig bridge0 delete $1
sudo /sbin/ifconfig $1 down
```

```
% chmod +x nono-ifup nono-ifdown
```

- `nono` を起動し、メニューの「モニタ > ホスト > ホストネットワーク」を開いて `HostNet Driver: tap` になっていれば動いているはずですが。 [Run `nono`, and open "Monitor > Host > Host Network" window from menu. It's OK if you can see "HostNet Driver: tap".]

▼ 7. 既知の問題 [Known Issues]

- デバッガのコンソール入力で、カーソルキーなどの特殊文字入力に対する表示が動作していません。 [In debugger console, some special characters like cursor keys aren't printed correctly.] (since ver 0.3.0)
- ホストが `VirtualBox` (on Windows?) で、`VirtualBox` の「マウス統合」が有効になっている場合、`nono` の「マウスモード」でマウスポインタが正しく移動しません。 [If the host OS is running on `VirtualBox` (on Windows?) and "Mouse

integration" on VirtualBox is enabled, the mouse pointer will not move as expected in nono's Mouse Mode.] (2021/12/30)

- Wayland の場合にウィンドウサイズが正しく設定されない場合があります。 wxWidgets (3.0.5) のバグと思われます。 [On Wayland, window size may be incorrect. This may be due to wxWidgets (3.0.5) bug.] (2021/12/30)

▼ 8. 過去のバージョンからの移行方法 **[How to migrate from old versions]**

バージョンアップに伴い設定ファイル等に非互換が発生する場合があります。その場合は以下の移行方法を参照して設定ファイル等を更新してください。 [Some versions may have incompatibilities in the configuration files, etc. In such case, you may need to upgrade it by referring the following link.]

- [From ver 0.2.x to ver 0.3.0](#)
- [From ver 0.1.x to ver 0.2.0](#)

▼ 9. 変更履歴 **[Changes]**

See [changes.html](#).

▼ 10. 連絡先、ライセンス等 **[Contact, License, etc]**

▼ 10.1. ライセンス **[License]**

- nono 自身については [nono-license.txt](#) を読んでください。 [See nono-license.txt about nono itself.]
- nono は無償公開されている X68030 の IPLROM を使用しています。 [許諾条件](#) を読んでください。 [nono uses X68030 IPLROM image distributed without charge. You must read [the license agreement](#) (even though it's Japanese original text only).]
- nono は SIL Open Font License version 1.1 で配布されているフォントの一部を改変して組み込んでいます。 [nono modifies and embeds the font distributed as SIL Open Font License version 1.1.]

▼ 10.2. 連絡先 **[Contact us]**

バグ報告などは以下をお願いします。日本語でおk。 [If you find any problems, please let me know. You may write in English.] <https://github.com/isaki68k/nono-issue/issues>

▼ 10.3. パッチの提供について **[About contributes]**

パッチを提供して下さる場合は以下に同意したものとします。 [If you provide a patch to nono, we assume that you agree the following conditions:]

- 成果物が nono のライセンスに従って運用あるいは配布されること。 [All your work will be operated or distributed under the nono license.]
- ライセンスが将来変わる可能性があること。 [The license may be changed in the future.]
- 著作部分に関して著作権者人格権を行使しないこと。 [Do not exercise your author's rights.]

▼ 10.4. Acknowledgements

nono は以下の広告条項を含むソースコードを利用しています。 [nono uses source code with the following advertising clause.]

This product includes software developed by Gordon Ross

This product includes software developed by the University of California, Lawrence Berkeley Laboratory.

1. 東京

Maps:

<https://maps.google.co.jp/maps/ms?msa=0&msid=208676479199435389545.0004c2d1822250759aa9f>

このドキュメント:

<https://github.com/ebijun/NetBSD/blob/master/Guide/Place/tokyo.rst>

2. 東京でのオープンソースカンファレンス

開催年	開催日	参加者	参加グループ	会場
2004	9/4	500	20	日本電子専門学校
2005	3-25-26	1100	33	日本電子専門学校
2005 Fall	9/17	800	35	日本電子専門学校
2006 Spring	3/17-18	1100	47	日本電子専門学校
2006 Fall	10/28	800	46	日本電子専門学校
2007 Spring	3/16-3/17	1200	42	日本電子専門学校
2007 Fall	10/5-10/6	1410	45	大田区産業プラザPiO
2008 Spring	2/29	1500	46	日本電子専門学校
2008 Fall	10/3-10/4	1300	58	大田区産業プラザPiO
2009 Spring	2/20-2/21	1450	77	日本電子専門学校
2009 Fall	10/30-10/31	1600	78	日本工学院専門学校
2010 Spring	2/26-2/27	1300	76	明星大学
2010 Fall	9/9-9/10	1400	77	明星大学
2011 Spring	3/4-3/5	2100	84	早稲田大学
2011 Fall	11月19日	1500	75	明星大学
2012 Spring	3/16-3/17	1700	74	明星大学
2012 Fall	9/6-9/7	1450	82	明星大学
2013 Spring	2/22-2/23	1700	78	明星大学
2013 Fall	10/19-20	1300	80	明星大学
2014 Spring	2/28-3/1	1900	82	明星大学
2014 Fall	10/18-19	1550	72	明星大学
2015 Spring	2/27-28	1700	79	明星大学
2015 Fall	10/24-25	1550	77	明星大学
2016 Spring	2/26-27	1550	71	明星大学
2016 Fall	11/5-6	1350	77	明星大学
2017 Spring	3/10-11	1300	72	明星大学
2017 Fall	9/9-10	1100	57	明星大学
2018 Spring	2/23-24	1200	62	明星大学
2018 Fall	10/27-28	1050	62	明星大学
2019 Spring	2/22-23	1010	68	明星大学
2019 Fall	11/23-24	830	63	明星大学
2020 Spring	4/24-25	500	19	オンライン
2020 Fall	10/23-24	500	15	オンライン
2021 Spring	3/6-7	570	26	オンライン
2021 Fall	10/22-23	350	18	オンライン
2022 Spring	3/11-12	420	19	オンライン
2022 Fall	10/28-29	270	15	オンライン
2023 Spring	3/10-11			オンライン
2023 Tokyo	4/1			東京都立産業貿易センター台東館

2.1. 観光ガイドバックナンバー

これまですべてのバックナンバーは

<https://github.com/ebijun/osc-demo> にあります。

No	イベント	URL
185	OSC2023東京春	http://www.re.soum.co.jp/~jun/OSC2023tokyospring.pdf
181	OSC2022東京秋	http://www.re.soum.co.jp/~jun/OSC2022tokyofall.pdf
175	OSC2022東京春	http://www.re.soum.co.jp/~jun/OSC2022tokyospring.pdf
171	OSC2021東京秋	http://www.re.soum.co.jp/~jun/OSC2021tokyofall.pdf
164	OSC2021東京春	http://www.re.soum.co.jp/~jun/OSC2021tokyospring.pdf
159	OSC2020東京秋	http://www.re.soum.co.jp/~jun/OSC2020tokyofall.pdf
152	OSC2019東京秋	http://www.re.soum.co.jp/~jun/OSC2019tokyofall.pdf
139	OSC2019東京春	http://www.re.soum.co.jp/~jun/OSC2019tokyospring.pdf
132	OSC2018東京秋	http://www.re.soum.co.jp/~jun/OSC2018tokyofall.pdf
121	OSC2018東京春	http://www.re.soum.co.jp/~jun/OSC2018tokyospring.pdf
113	OSC2017東京秋	http://www.re.soum.co.jp/~jun/OSC2017tokyofall.pdf
104	OSC2017東京春	http://www.re.soum.co.jp/~jun/OSC2017tokyospring.pdf
97	OSC2016東京秋	http://www.re.soum.co.jp/~jun/OSC2016tokyofall.pdf
85	OSC2016東京春	http://www.re.soum.co.jp/~jun/OSC2016tokyospring.pdf
80	OSC2015東京秋	http://www.re.soum.co.jp/~jun/OSC2015tokyofall.pdf

69	OSC2015東京春	http://www.re.soum.co.jp/~jun/OSC2015tokyospring.pdf
64	OSC2014東京秋	http://www.re.soum.co.jp/~jun/OSC2014tokyofall.pdf
53	OSC2014東京春	http://www.re.soum.co.jp/~jun/OSC2014tokyospring.pdf
49	OSC2013東京秋	http://www.re.soum.co.jp/~jun/OSC2013tokyofall.pdf
39	OSC2013東京春	http://www.re.soum.co.jp/~jun/OSC2013tokyospring.pdf
31	OSC2012東京秋	http://www.re.soum.co.jp/~jun/OSC2012tokyofall.pdf
22	OSC2012東京春	http://www.re.soum.co.jp/~jun/OSC2012tokyospring.pdf
19	OSC2011東京秋	http://www.re.soum.co.jp/~jun/OSC2011tokyofall.pdf
6	OSC2011東京春	http://www.re.soum.co.jp/~jun/OSC2011tokyospring.pdf

2.2. OSC東京まとめ

OSC2023 Online/Spring NetBSDのご紹介 の記録	https://togetter.com/li/2094211
OSC2022 Online/Fall BSDなひとときの記録	https://togetter.com/li/1964480
OSC2022 Online/Spring NetBSDのご紹介 の記録	https://togetter.com/li/1856809
OSC2021 Online/Fall NetBSDのご紹介 の記録	https://togetter.com/li/1790501
OSC2021 Online/Spring NetBSDのご紹介 の記録	https://togetter.com/li/1675363
OSC2020東京秋 BSDなひとときの記録	https://togetter.com/li/1611536
OSC2019東京秋 NetBSDブース展示の記録	https://togetter.com/li/1433822
OSC2019東京春 NetBSDブース展示の記録	https://togetter.com/li/1321833
OSC2018東京秋 NetBSDブース展示の記録	https://togetter.com/li/1281694
OSC2018東京春 NetBSDブース展示の記録	https://togetter.com/li/1202581
OSC2017東京秋 NetBSDブース展示の記録	https://togetter.com/li/1148810
OSC2017東京春 NetBSDブース展示の記録	https://togetter.com/li/1089005
OSC2016東京秋 NetBSDブース展示の記録	https://togetter.com/li/1044981
OSC2016東京春 NetBSDブース展示の記録	http://togetter.com/li/943037
OSC2015東京秋 NetBSDブース展示の記録	http://togetter.com/li/891197
OSC2015東京春 NetBSDブース展示の記録	http://togetter.com/li/788681
OSC2014東京秋 NetBSDブース展示の記録	http://togetter.com/li/733588
OSC2014東京春 NetBSDブース展示の記録	http://togetter.com/li/635747
OSC2013東京秋 NetBSDブース展示の記録	http://togetter.com/li/578893
OSC2013東京春 NetBSDブース展示の記録	http://togetter.com/li/460484
OSC2012東京秋 NetBSDブース展示の記録	http://togetter.com/li/369471
OSC2012東京春 NetBSDブース展示の記録	http://togetter.com/li/274145

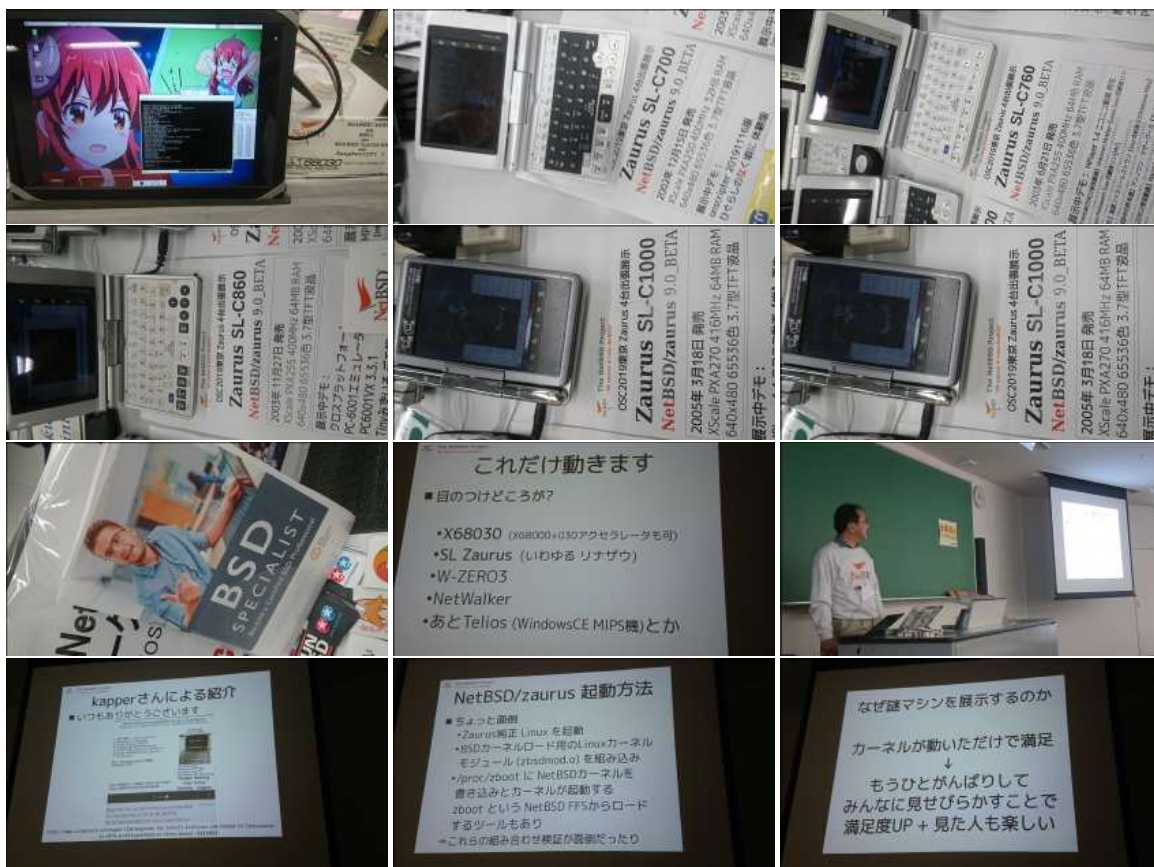
2.3. よく行く店

みぞのくち醸造所	高津	11:00-23:00 ウタウト ワラウト
いくだん柿生店	柿生	村人が集う焼肉店。いしはらさんの合言葉は「レバー大辛」。
VERTERE	奥多摩	クラフトビール 月金14:00-19:30 土日12:00-19:30
Snark Liquidworks	要町	クラフトビールとDEC 水木土日 12:00-21:00 金17:00-21:00
riot beer	祖師ヶ谷大蔵	クラフトビール 木~土15:00-21:00 土日13:00-18:00
フタコビール	二子玉川	クラフトビール 11:30-23:00
エビナビール	海老名	クラフトビール 月~金17:00-22:30 土日14:00-22:30
TDM1874	十日市場	ブルワリー+酒屋。IPA #バージョン番号
さかづきブルーイング	北千住	クラフトビール 水~金16:00-22:30 土日13:00-22:30
アボットチョイス	町田	21タップ
GreenBud!!	本厚木	村人が集うヨーロッパビール店
ピガール	三軒茶屋	オリジナルビールL.S.D(Love Saison Darling)
ノベルクラフト	千歳烏山	18:00-
Welders diner	石神井公園	11:00-23:00
グレムリン	幡ヶ谷	15:00- 弊社から60秒でクラフトビール。
サトー電気	町田/川崎/小机	エレクトロセンターにあった部品が一部ある
パパのいうことを聞きなさい!	中央大学	ひな役がちゃんおぶ五十嵐裕美さんだお。
ポパイ	両国	ビール屋さんが研修に来る店。ぶらっと後藤さんが常連。
古書ドリス	鶯谷	幻想図書を扱う古書店。徳島から移転。
古書ほうろう	根津	池之端門前
すた城	東小金井	NBUGみずのさんと事務局長が育った居酒屋。しこ天==じゃこ天。
富士ランチ	東小金井	大盛りとは何か考える。
加賀屋	本郷三丁目	もつMLというBSD関連の集まりがありましてな。
丹青通商	亀戸	AppleII 5000円 マイコンマニュアル&ばぶ
三ちゃん	荻窪	ピングドラムに出てきたラーメン店。スタミナラーメン。
ムーンパイス	門前仲町	個人経営の粋なプラモデル店
ムーンライト	生田	ビール醸造所。季節のフルーツビール。
ヌビチヌ	野毛	素晴らしいスタンディングビアバー。
大衆	生麦	焼肉ジギスカンホッピー焼肉ジギスカンホッピー。
たん清	秋葉原	BSDマガジンの焼肉夜話。
名曲喫茶ライオン	渋谷	百軒店の伝説。
ヴィオロン	阿佐谷	名曲喫茶。真空管アンプの最終目的地。
Galaxy	原宿	マニアックラヴ&シャッフルマスター&REBOOT
池田電子	町田	町田市金森187-20 木曜定休 うずたかく積まれている

2.4. OSC東京主な展示物

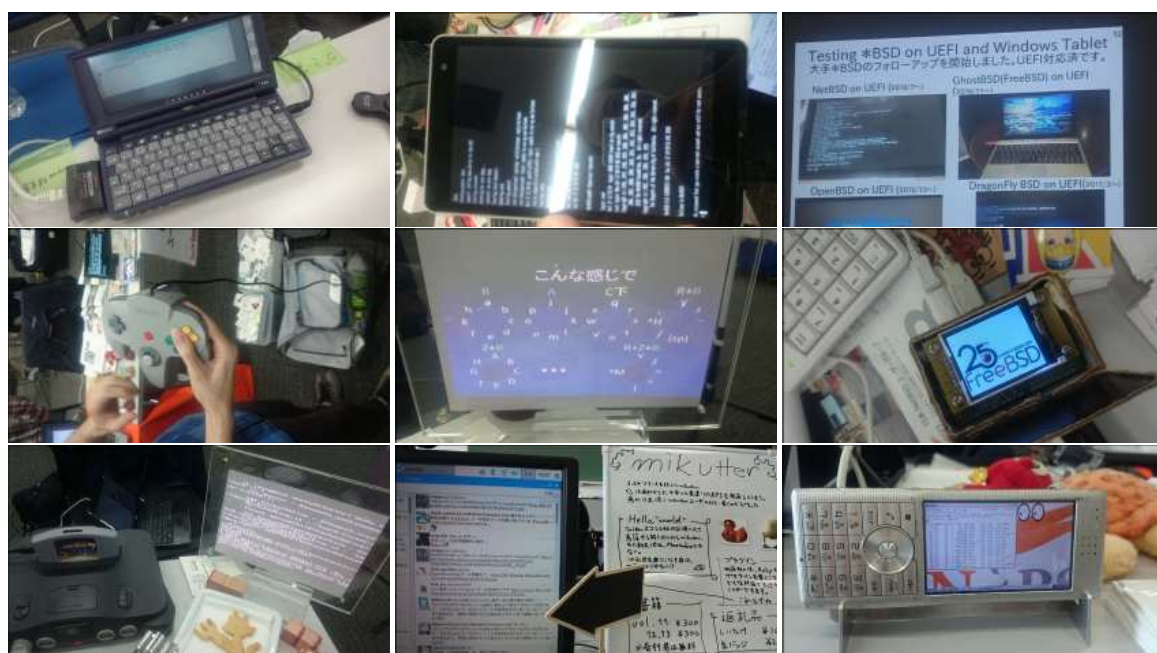
2019/11/23-24	OSC2019東京秋	NetBSD/zaurus8.1
2019/2/22-23	OSC2018東京春	Zrouter FreeBSD/mips NetBSD/nintendo64
2018/10/27-28	OSC2018東京秋	Linux on Nintendo64
2018/2/23-24	OSC2018東京春	RPI3+XM6i GPD Sparcbook BananaPis
2017/09/9-10	OSC2017東京秋	OrangePi One NanoPi RPI3+XM6i+NetBSD/x68k
2017/03/10-11	OSC2017東京春	RPI3+XM6i+NetBSD/x68k RPI2+omxplayer NetWalker WZERO3 GPDWIN
2016/11/5-6	OSC2016東京秋	DreamCast+IDE HDD
2016/2/26-7	OSC2016東京春	fdgw2 RK3188 WM8750 USL-5P RouterBoard AR9331 pcDuino
2015/10/24-5	OSC2015東京秋	RK3188 WM8750 EP9315 AM3358 Allwinner A20 405GP BCM2836 AR9331
2015/2/27-8	OSC2015東京春	Radxa RocK APC8750 RPI OpenBlockS sandpoint
2014/10/18-19	OSC2014東京秋	Xen dom0 BSD全部 RadxaRock zaurusC700音源
2014/2/28-3/1	OSC2014東京春	Netwinder RPI ZAURUS第六艦隊 OSCアワード受賞
2013/10/19-20	OSC2013東京秋	Netwinder OpenblocksA6 RPI
2013/2/22-23	OSC2013東京春	RPI KOBO netwalker mikutter WZERO3
2012/9/7-8	OSC2013東京秋	OpenblocksA6 Openblocks266/600 RPI KOBO netwalker USL5P かめぬい
2012/3/16-17	OSC2012東京春	ておくれイメージ WZero3 Zaurus Jornada680/720 yeeloong ZBOXNANO
2011/11/19-20	OSC2011東京秋	ておくれイメージ BBC A3000 WZero3全部 Zaurus全部 hpcmips/sh/arm
2011/3/4-5	OSC2011東京春	hpcmips/hpcarm/hpcsh
2010/9/10-11	OSC2010東京秋	hpcmips/sh/arm XM6i
2010/2/26-27	OSC2010東京春	hpcmips/sh/arm XM6i
2009/10/30-31	OSC2009東京秋	USBLCD USL5P LS-S250L tadpole hpcmips/sh/arm
2009/2/20-21	OSC2009東京春	VAIO-P tadpole hpcmips/sh/arm
2008/10/3-4	OSC2008東京秋	Linkstation tadpole PCBSD/Xen hpcmips/sh/arm
2008/2/29-3/1	OSC2008東京秋	hpcmips/sh/arm EeePC
2007/10/5-6	OSC2007東京秋	X68030Nereid itojun最後のプレゼン hpcmips/sh/arm
2007/3/16-17	OSC2007東京春	hpcmips/sh/arm スピーシーズ
2006/10/28	OSC2006東京秋	hpcmips/sh/arm
2006/3/17-18	OSC2006東京春	hpcmips/sh/arm
2005/3/25-26	OSC2005	hpcmips/sh/arm
2004/9/4	OSC2004	hpcmips/sh/arm

2.5. 2019年





2.6. 2018年





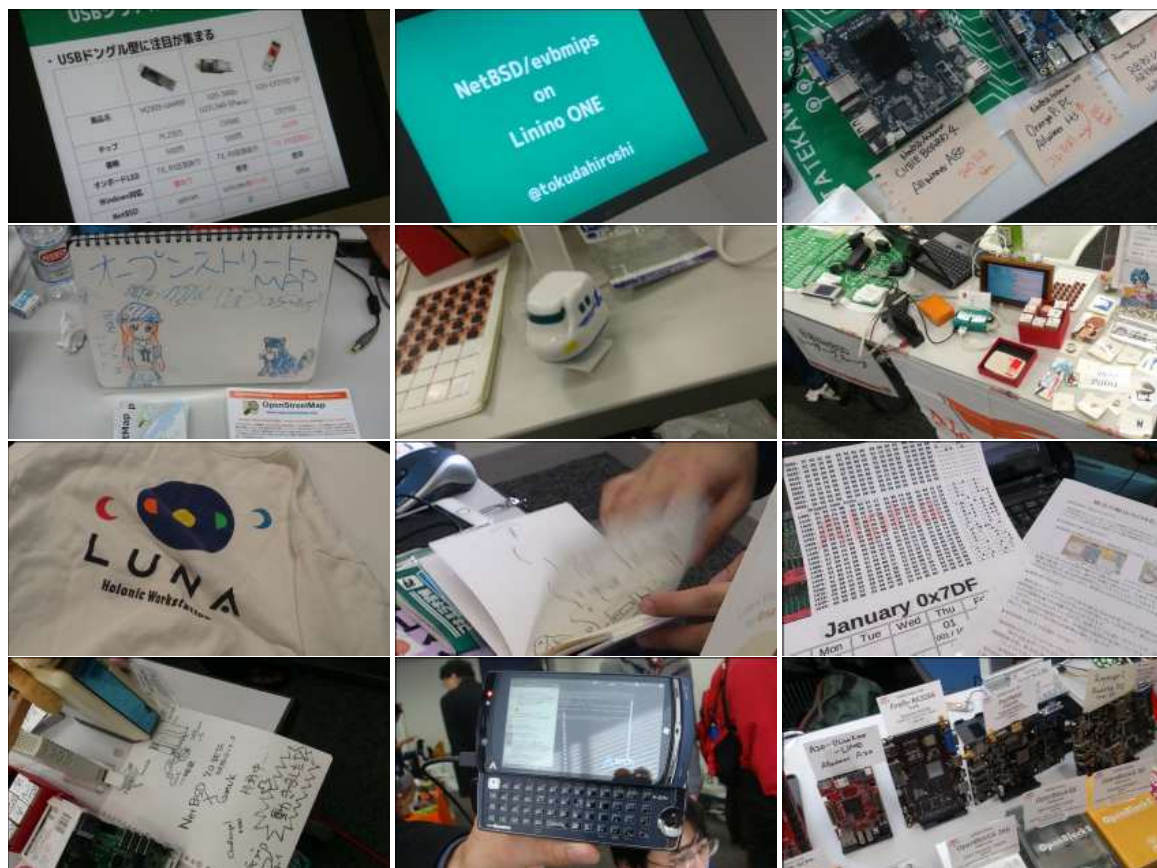
2.7. 2017年



2.8. 2016年



2.9. 2015年



2.10. 2014年



2.11. 2013年



2.12. 2012年



2.13. 2011年/2007年





3. RaspberryPIのNetBSDイメージ2023進捗どうですか

3.1. RaspberryPIのNetBSDイメージについて

今年もオープンソースカンファレンスごとにRaspberryPI用のNetBSDイメージを作って配布しています。この一年、どんなことがあったのか表にしてまとめてみました。

年月	NetBSD	mikutter	mlterm	OpenSSL	ネタ	OSC	URL
2019/8/3	8.99.51→9.99.1	3.9.2			9.0 BETA	OSC京都	http://mail-index.netbsd.org/port-arm/2019/07/31/msg005994.html
2022/1/29	9.99.92	5.0.2	3.9.2	1.1.1m	drm大改修	OSC大阪	http://mail-index.netbsd.org/port-arm/2022/01/10/msg007542.html
2022/3/12	9.99.94	5.0.4			tor	OSC東京春	http://mail-index.netbsd.org/port-arm/2022/03/05/msg007597.html
2022/4/09	9.99.96			1.1.1n			http://mail-index.netbsd.org/port-arm/2022/03/05/msg007628.html
2022/5/28	9.99.96		3.9.2nb1		OpenSSH9.0	OSC名古屋	http://mail-index.netbsd.org/port-arm/2022/03/05/msg007674.html
2022/6/25	9.99.98				icu-71.1	OSC北海道	http://mail-index.netbsd.org/port-arm/2022/06/22/msg007697.html
2022/7/30	9.99.99		3.9.2nb4		Python3.10	OSC京都	http://mail-index.netbsd.org/port-arm/2022/07/25/msg007730.html
2022/8/28	9.99.99	5.04nb1	3.9.2nb5		nono 0.4.3	ODC	http://mail-index.netbsd.org/port-arm/2022/08/30/msg007814.html
2022/9/24	9.99.100		3.9.2nb6			OSC広島	http://mail-index.netbsd.org/port-arm/2022/09/24/msg007863.html
2022/10/29	9.99.102				rust/earmv6hf	OSC東京	http://mail-index.netbsd.org/port-arm/2022/10/26/msg007897.html
2023/1/28	10.99.2	5.0.4nb2	3.9.2		10ランチ	OSC大阪	http://mail-index.netbsd.org/port-arm/2023/01/19/msg008093.html
2023/2/17	10.99.2	5.0.4nb3	3.9.2nb10	1.1.1t		NBUG2月	http://mail-index.netbsd.org/port-arm/2023/02/17/msg008116.html

2023/3/11	10.99.2					OSC東京春	http://mail-index.netbsd.org/port-arm/2023/03/04/msg008133.html
年月	NetBSD	mikutter	mlterm	OpenSSL	ネタ	OSC	URL

OSCはほぼ毎月のように日本各地で行われています。前に、OpenBSDのTheoさんに、自分のノートPCのアップデートをどのくらいの周期でやってるのかきいてみました。2週間くらいごとかかと答えてくれて、あーだいたいそんなものなのかと思っていました。

NetBSDのイメージを配るとしたとき、どのくらいの周期でアップデートしていけばいいのでしょうか？イメージを配る理由は、何かソフトウェアが新しくなって新しい機能が入ったとか、ハードウェアのサポート種類が増えたとか、ソフトウェアの脆弱性が出たとか、理由はいくつかあると思いますが、試しにずっと更新して配りつづけることにしてみました。

イメージのサイズは2GBにしてみました。ダウンロードにかかる時間とか考えると、これ以上でっかくすると使ってもらえません。2GBのカードのサイズはこんくらいにすればいいよとFreeBSDのワーナーさんに教えてもらってずっとそのサイズにしていますが、手狭になったので増やしました。

イメージに入れるソフトを何にするか考えたんですが、mikutterとmltermにしてみました。RubyのGUI環境+ネットワーク認証を使うソフトと、基本的なターミナルソフトで、sixelグラフィックも表示できるのでおもしろそうです。

作り方は

<https://github.com/ebijun/NetBSD/blob/master/Guide/RPI/RPIImage.rst>

みたいに作って、あらかじめ作っておいたパッケージを組み込んで動作テストをします。mikutterで「あひる焼き」とつぶやいて返事が帰ってくればネットワーク認証と画面表示とRubyまわりと漢字入力がかまくらいいです。

3.2. 新しいハードウェア対応

1. RPI4:OSC2019島根から:<http://mail-index.netbsd.org/port-arm/2019/10/03/msg006208.html>

<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIImage/Image/aarch64/README>

1. RPI3/RPI0WのBluetooth/無線LAN:OSC2019広島版からテストをはじめました
2. Raspberry PI zero 2 W earmv7hf版で動きます。

<https://github.com/ebijun/NetBSD/blob/master/RPI/RPIImage/Image/earmv7hf/README>

3.3. ソフトウェア配布方法

NetBSDのftpサイトはCDN対応のところからダウンロードできるようになりました。漫喫でも楽勝です。 - <http://cdn.netbsd.org/> - <http://nycdn.netbsd.org/>

3.4. OSCでやっているデモ

RaspberryPIっぽいにかとということで、omxplayerを使ってcrontabで動画を流すデモと、XM6iでNetBSD/x68kを動かすデモをやっていました。

3.5. security.pax.mprotect.enabled

```
man security
man paxctl
sysctl -a |grep pax
If application failed, such as omxplayer.
try to test
sysctl -w security.pax.mprotect.enabled=0
```

3.6. GPIOのドキュメント

GPIOの使い方をまとめてくれた方が。

- NetBSD GPIO DOC by Marina Brown <https://github.com/catskillmarina/netbsd-gpio-doc/blob/master/README.md>

3.7. 64bit対応

ryo@netbsd さんによる rpi64wip実装が進み、NetBSD/aarch64としてRPI3/4で利用できます。

- <https://github.com/ryo/netbsd-src>
- <http://mail-index.netbsd.org/port-arm/2018/02/20/msg004631.html>
- <http://mail-index.netbsd.org/port-arm/2018/12/03/msg005297.html>

3.8. RPI4

- pinebookとpkgsrcを共用しています。
- <http://mail-index.netbsd.org/port-arm/2020/11/18/msg007066.html>
- <https://github.com/ebijun/NetBSD/blob/master/RPI/RPIImage/Image/aarch64/README>

3.9. armv7のいろいろ

NetBSD ARM Bootable Imagesがあります。

- <https://nycdn.netbsd.org/pub/arm/>

3.10. ご注文はなんとかですか(弱点)

- RPI4のGPUとか

3.11. まとめ

OSCごとにイメージをつくっていると、だいたいBINDとOpenSSLの脆弱性に対応できていい感じです。なんでOSCの直前になると脆弱性が見つかるんでしょうか。たまたまBSD自体の10年もののバグとかも発掘されて楽しいです。リリース間隔があげばあくほど、ひとりで対応できる作業量を越えてしまう気がするので、いまんとここれでいいのかほんとうに。

4. RaspberryPIでNetBSDを使ってみる

4.1. 特徴

- NetBSDをRaspberryPIで利用するために、ディスクイメージを用意しました。
- Xが動いて、ご家庭のテレビでmikutterが動きます。
- うまく動いたら、動いた記念写真をツイートだ！
- fossil(<http://www.fossil-scm.org/>)も入れてあります。家庭内Webサーバとかチケットシステムとかwikiサーバになるんでないかい。

4.2. 準備するもの

- RaspberryPI本体
- HDMI入力のあるテレビ/ディスプレイ
- USBキーボード
- USBマウス
- 有線ネットワーク

ケースは Geekworm Raspberry Pi 3/4ケースをそれぞれ使っています。

4.3. 起動ディスクの作成

- ディスクイメージのダウンロード

```
earmv6hf
# ftp http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/2016-11-12-earmv6hf/2016-11-12-netbsd-raspi-earmv6hf.img.gz
```

- 2GB以上のSDカードを準備します。
- ダウンロードしたディスクイメージを、SDカード上で展開します。

```
disklabel sd0 ..... 必ずインストールするSDカードか確認してください。
gunzip < 2016-11-12-netbsd-raspi-earmv6hf.img.gz | dd of=/dev/rsd0d bs=1m
```

4.4. NetBSD Arm Bootable Images

NetBSD Arm Bootable Images が、<https://nycdn.netbsd.org/pub/arm/> 以下にあります。RPIと同じ手順で起動できます。

4.5. RaspberryPIの起動

1. HDMIケーブル/USBキーボード/USBマウス/有線ネットワークをRPIにさします。
2. 電源を入れてRPIを起動します。
3. 少し待つと、HDMIからNetBSDの起動メッセージが表示されます。
4. メモリカードの容量にあわせたサイズまでルートパーティションを自動調整します。(現在、RPI2では自動調整プログラムの起動が失敗します)
5. 容量調整後に再起動します。再起動した後は、起動プロセスが最後まで進み、ログインできる状態になります。
6. 起動しない場合、まず基板上のLEDを確認してください。

赤いランプのみ点灯している場合

- OSを正しく読み込んでいません。
- 少なくともMSDOS領域に各種ファームウェアファイルが見えていることを確認する。
- SDカードの接触不良の可能性があるので、SDカードを挿しなおしてみる。
- ファームウェアが古いため起動しない

緑のランプも点灯している場合

- OSは起動しているのに画面をHDMIに表示できていません。
- HDMIケーブルを差した状態で電源ケーブルを抜き差しして、HDMIディスプレイに何か表示するか確認する。
- HDMIケーブル自体の接触不良。ケーブルを何度か差し直してください。
- 電源アダプタ容量には、少なくとも800mA程度の容量を持つアダプタを使ってみてください。スマートフォン用のアダプタならまず大丈夫です。起動途中で画面が一瞬消えたり、負荷をかけるといきなり再起動したりする場合は、電源やUSBケーブルを気にしてみてください。

4.6. ログイン

rootでログインできます。rootアカウントではリモートからログインすることはできません。

```
login: root
```

```
startxでicewmが立ち上がります。
```

```
# startx
```

4.7. mikutter を使ってみよう

- xterm から dillo と mikutter を起動します。

```
# dillo &
# mikutter &
```

- しばらく待ちます。
- mikutter の認証画面がうまく出たら、https からはじまる URL をクリックすると dillo が起動します。
- twitter の ID とパスワードを入力すると、pin 番号が表示されます。pin 番号を mikutter の認証画面に入力します。
- しばらくすると、mikutter の画面が表示されます。表示されるはずですが、落ちてしまう場合は時計が合っているか確認してください。
- 漢字は [半角/全角] キーを入力すると漢字モードに切り替わります。anthy です。
- 青い鳩を消したいとき: mikutter のプラグインを試してみる

```
% touch ~/.mikutter/plugin/display_requirements.rb
```

すると、鳩が消えます。mikutter はプラグインを組み込むことで、機能を追加できる自由度の高い twitter クライアントです。プラグインに関しては、「mikutter の薄い本 プラグイン」で検索してみてください。

4.8. fossil を使ってみよう

fossil は、Wiki/チケット管理システム/HTTP サーバ機能を持つ、コンパクトなソースコード管理システムです。fossil バイナリひとつと、リポジトリファイルひとつにすべての情報が集約されています。ちょっとしたメモをまとめたり ToDo リストを簡単に管理できます。

```
% fossil help
Usage: fossil help COMMAND
Common COMMANDS: (use "fossil help -a|--all" for a complete list)
add      changes  finfo      merge     revert    tag
addremove clean    gdiff     mv        rm        timeline
all      clone     help      open     settings  ui
annotate commit   import    pull     sqlite3   undo
bisect   diff     info      push     stash     update
branch   export   init      rebuild  status    version
cat      extras   ls        remote-url sync
% fossil init sample-repo
project-id: bcf0e5038ff422da876b55ef07bc8fa5eded5f55
server-id: 5b21bd9f4de6877668f0b9d90b3cff9baecea0f4
admin-user: jun (initial password is "f73efb")
% ls -l
total 116
-rw-r--r-- 1 jun users 58368 Nov 14 18:34 sample-repo
% fossil server sample-repo -P 12345 &
ブラウザでポート12345にアクセスし、fossil initを実行した時のユーザとパスワードでログインします。
```

4.9. キーマップの設定を変更する

- ログインした状態でのキーマップは /etc/wscons.conf で設定します。

```
encoding jp.swapctrlcaps .... 日本語キーボード, Ctrl と CAPS を入れ替える。
```

- X でのキーマップは、xinitrc で設定します。

```
setxkbmap -model jp106 jp -option ctrl:swapcap
```

4.10. コンパイル済パッケージをインストールする

- コンパイルしたパッケージを以下の URL に用意しました。

```
% cat /etc/pkg_install.conf
```

```
PKG_PATH=http://cdn.netbsd.org/pub/NetBSD/misc/jun/raspberry-pi/earmv6hf/2016-11-12
```

- パッケージのインストール

pkg add コマンドで、あらかじめコンパイル済みのパッケージをインストールします。関連するパッケージも自動的にインストールします。

```
# pkg_add zsh
```

- パッケージの一覧

pkg info コマンドで、インストールされているパッケージの一覧を表示します。

```
# pkg_info
```

- パッケージの削除

```
# pkg_delete パッケージ名
```

4.11. /usr/pkgsrc を使ってみよう

たとえば wordpress をコンパイル/インストールする時には、以下の手順で行います。

```
# cd /usr/
# ls /usr/pkgsrc          ... 上書きしてしまわないか確認
# ftp http://cdn.netbsd.org/pub/pkgsrc/current/pkgsrc.tar.gz
# tar tzvf pkgsrc.tar.gz |head ... アーカイブの内容確認
# tar xzvf pkgsrc.tar.gz
# ls /usr/pkgsrc
# cd /usr/pkgsrc/www/php-ja-wordpress
# make package-install
```

```
# cd /usr/pkgsrc
# cvs update -PA
```

4.12. パッケージ管理

pkg_chk コマンドを使って、インストールしたパッケージを管理してみましょう。あらかじめpkgsrcの内容を更新しておきます。どこからパッケージファイルを取得するかは、/etc/pkg.install.confのPKG_PATHに書いておきます。

```
# pkg_info ... インストールしているパッケージ名と概要を出力します。
# pkg_chk -g ... 使っているパッケージの一覧を/usr/pkgsrc/pkgchk.confに作ってくれます。
# pkg_chk -un ... パッケージをアップデートします。(nオプション付きなので実行はしません)
# pkg_chk -u ... パッケージをアップデートします。
```

4.13. ユーザー作成

```
# useradd -m jun
# passwd jun
```

root権限で作業するユーザーの場合:

```
# useradd -m jun -G wheel
```

```
# passwd jun
```

4.14. サービス起動方法

/etc/rc.d以下にスクリプトがあります。dhcpクライアント(dhcpd)を起動してみます。

```
テスト起動:
/etc/rc.d/dhcpd onestart
テスト停止:
/etc/rc.d/dhcpd onestop
```

正しく動作することが確認できたら/etc/rc.confに以下のとおり指定します。

```
dhcpd=YES
```

/etc/rc.confでYESに指定したサービスは、マシン起動時に同時に起動します。

```
起動:
/etc/rc.d/dhcpd start
停止:
/etc/rc.d/dhcpd stop
再起動:
/etc/rc.d/dhcpd restart
```

4.15. vndconfigでイメージ編集

NetBSDの場合、vndconfigコマンドでイメージファイルの内容を参照できます。

```
# gunzip 2016-11-12-netbsd-raspi-earmv6hf.img.gz
# vndconfig vnd0 2016-11-12-netbsd-raspi-earmv6hf.img
# vndconfig -l
vnd0: /usr (/dev/wd0e) inode 53375639
# disklabel vnd0
:
8 partitions:
   size  offset  fstype [fsize bsize cpq/sgs]
a:  3428352  385024   4.2BSD    0    0    0 # (Cyl.  188 - 1861)
b:   262144  122880   swap                # (Cyl.   60 - 187)
c:  3690496  122880  unused    0    0    # (Cyl.   60 - 1861)
d:   3813376    0  unused    0    0    # (Cyl.    0 - 1861)
e:   114688    8192  MSDOS                # (Cyl.    4 - 59)
# mount_msdos /dev/vnd0e /mnt
# ls /mnt
LICENCE.broadcom  cmdline.txt      fixup_cd.dat     start.elf
bootcode.bin     fixup.dat        kernel.img        start_cd.elf
# cat /mnt/cmdline.txt
root=ld0a console=fb
#fb=1280x1024      # to select a mode, otherwise try EDID
#fb=disable       # to disable fb completely

# umount /mnt
```



```
# vndconfig -u vnd0
```

4.16. HDMIじゃなくシリアルコンソールで使うには

- MSDOS領域にある設定ファイルcmdline.txtの内容を変更してください。

<https://raw.githubusercontent.com/Evilpaul/RPi-config/master/config.txt>

```
fb=1280x1024      # to select a mode, otherwise try EDID
fb=disable        # to disable fb completely
```

4.17. 起動ディスクを変えるには

- MSDOS領域にある設定ファイルcmdline.txtの内容を変更してください。

root=sd0a console=fb ←ld0をsd0にするとUSB接続したディスクから起動します

4.18. 最小構成のディスクイメージ

NetBSD-currentのディスクイメージに関しては、以下の場所にあります。日付の部分は適宜読み替えてください。

```
# ftp://nyftp.netbsd.org/pub/NetBSD-daily/HEAD/201502042230Z/evbarm-earmv6hf/binary/gzimg/rpi_inst.bin.gz
# gunzip < rpi_inst.bin.gz |dd of=/dev/rsd3d bs=1m .... sd3にコピー。
```

```
RaspberryPIにsdカードを差して、起動すると、# プロンプトが表示されます。
# sysinst .... NetBSDのインストールプログラムが起動します。
```

4.19. X11のインストール

rpi.bin.gzからインストールした場合、Xは含まれていません。追加したい場合は、

<ftp://nyftp.netbsd.org/pub/NetBSD-daily/HEAD/201310161210Z/evbarm-earmv6hf/binary/sets/> 以下にあるtarファイルを展開します。tarで展開するときにpオプションをつけて、必要な権限が保たれるようにしてください。

```
tar xzpvf xbase.tar.gz -C / .... pをつける
```

4.20. クロスビルドの方法

- ソースファイル展開
- ./build.sh -U -m evbarm -a earmv6hf release
- earm{v[4567],}{hf,}{eb} earmv4hf
- <http://mail-index.netbsd.org/tech-kern/2013/11/12/msg015933.html>

acorn26	armv2
acorn32	armv3 armv4 (strongarm)
cats shark netwinder	armv4 (strongarm)
iyonix	armv5
hpcarm	armv4 (strongarm) armv5.
zaurus	armv5
evbarm	armv5/6/7

4.21. 外付けUSB端子

NetBSDで利用できるUSBデバイスは利用できる(はずです)。電源の制約があるので、十分に電源を供給できる外付けUSBハブ経由で接続したほうが良いです。動作しているRPIにUSBデバイスを挿すと、電源の関係でRPIが再起動してしまう場合があります。その場合、電源を増強する基板を利用する方法もあります。

4.22. 外付けSSD

コンパイルには、サンディスク X110 Series SSD 64GB(読込 505MB/s、書込 445MB/s) SD6SB1M-064G-1022I を外付けディスクケース経由で使っています。NFSが使える環境なら、NFSを使い、pkgsrcの展開をNFSサーバ側で実行する方法もあります。RPIにSSDを接続した場合、OSの種類と関係なく、RPI基板の個体差により、SSDが壊れる場合があるので十分注意してください。

4.23. 液晶ディスプレイ

液晶キット(<http://www.aitendo.com/page/28>)で表示できています。

aitendoの液晶キットはモデルチェンジした新型になっています。On-Lap 1302でHDMI出力を確認できました。HDMI-VGA変換ケーブルを利用する場合、MSDOS領域にある設定ファイルcmdline.txtで解像度を指定してください。

```
https://twitter.com/oshimya/status/399577939575963648
```

とりあえずうちの1024x768の液晶の場合、 hdmi_group=2 hdmi_mode=16 の2行をconfig.txtに書いただけ。なんと単純。disable_borderはあってもなくても

4.24. inode

inodeが足りない場合は、ファイルシステムを作り直してください。

```
# newfs -n 500000 -b 4096 /dev/rvnd0a
```

4.25. bytbench

おおしまさん(@oshimya)がbytbenchの結果を測定してくれました。

<https://twitter.com/oshimya/status/400306733035184129/photo/1/400303304573341696/photo/1>

<https://twitter.com/oshimya/status>

4.26. 壁紙

おおしまさん(@oshimya)ありがとうございます。

<http://www.yagoto-urayama.jp/~oshimaya/netbsd/Proudly/2013/>

--

4.27. パーティションサイズをSDカードに合わせる

2GB以上のSDカードを利用している場合、パーティションサイズをSDカードに合わせることができます。この手順はカードの内容が消えてしまう可能性もあるため、重要なデータはバックアップをとるようにしてください。

手順は、http://wiki.netbsd.org/ports/evbarm/raspberry_pi/ のGrowing the root file-systemにあります。

4.27.1. シングルユーザでの起動

1. /etc/rc.confのrc_configured=YESをNOにして起動します。
2. 戻すときはmount / ;vi /etc/rc.conf でNOをYESに変更してrebootします。

4.28. 参考URL

- http://wiki.netbsd.org/ports/evbarm/raspberry_pi/
- NetBSD Guide <http://www.netbsd.org/docs/guide/en/>
- NetBSD/RPiで遊ぶ(SDカードへの書き込み回数を気にしつつ) <http://hachulog.blogspot.jp/2013/03/netbsdrpisd.html>
- <http://www.raspberrypi.org/phpBB3/viewforum.php?f=86> NetBSDフォーラム
- <http://www.raspberrypi.org/phpBB3/viewforum.php?f=82> 日本語フォーラム

5. ルナ式練習帳、またはLunaの薄い本2023 [1]

「私が誰かは わかっているはずだ。」その声は天使の声だった。

--- さまよえる天使 [2] [3] バーナード マラムッド [4]



5.1. History and Background of LUNA

The LUNA hardware had two different operating systems; a 4.3BSD derivative and a SVR3 variant. The first one, named UNIOS-B, was a port of Integrated Solution Inc. UNIX product. ISI manufactured m68k based VME UNIX boxes. Their OS had an interesting feature of TRFS (Translucent Remote File System) as well as the popular SMI's NFS. The paper of TRFS was published at USENIX Technical Conference (late '80, details unknown in this moment). [174]

5.2. nono - LUNA-I emulator

「nono は NetBSD とかで動作する LUNA-I とかのエミュレータです。でもまだ動きません。」 [167]

nono 0.0.3 (2020/05/16) 置いときますね。
<http://pastel-flower.jp/~isaki/nono/>
 なんちゃってROM用意したので、実機ROMなくても一応起動はすると思う。けど起動しかできないのと、こっからどうしたもんかは追々…。

「nonoさんが実機ROMなしでも起動するようなのでとりあえず最小インストールイメージを置きました」 [168]

NetBSD/luna68k 9.0 minimam liveimage 20200518版
<http://teokurebsd.org/netbsd/liveimage/20200518-luna68k/>

- pkgsrc経由でのnonoインストール

```
pkgsrc/emulators/nono
make package-install
https://gnats.netbsd.org/55761
https://twitter.com/isaki68k/status/1315996525919518724
http://www.pastel-flower.jp/~isaki/NetBSD/patch/pkgsrc-nono-20201013.diff
http://www.pastel-flower.jp/~isaki/NetBSD/patch/nono-20201013.diff
add /etc/mk.conf
ACCEPTABLE_LICENSES+= nono-license
cd /usr/pkgsrc/emulators/nono;make ;make package-install
```

- nonoからのNetBSD/luna68k liveimage起動 [169]

```
1) nono-0.1.1 をダウンロード
https://twitter.com/isaki68k/status/1261646479816404992
2) 展開して doc/index.html を読んでビルド
3) liveimage をダウンロードして gunzip
4) nono.cfg を作って置く
https://gist.github.com/tsutsui/340546bdc064cee786ed2473fb510463
5) wx/nono で実行
6) Emulated ROM上で以下のコマンドを実行
k
[enter]
[enter]
d
boot
g
x
```

```
vmtype=luna
#ethernet-hostdriver=tap
#prom-use-rom=0 #外部ROMを指定しなければ内蔵なんちゃってROMで上がるので指定しなくても動く
#spc0-id6-writeprotect=1 #ディスクライトプロテクト。デモとかで^Cで落とす用。
spc0-id6-image=hd,liveimage-luna68k-raw-20200518.img
```

- Luna88Kの起動

「設定ファイルでvmtype=luna88kにして、O/luna88kのリリースセットの中のboot を-Aオプションで指定とかまでは出来ます。」 [170]

「it was made from scratch.」 [171]

#OpenBSD/luna88k 6.8-current runs on nono-0.1.4 on #OpenBSD/amd64. Now I can login to virtual luna88k machine!
 Great! [178]

For anyone interested in nono and luna88k, I put OpenBSD/luna88k live image. (990MB gzip'ed, 2.0G uncompressed) Set this image as spc0-id6-image in nono.cfg, and start nono with OpenBSD/luna88k bootloader, i.e. "nono (other options) -A boot" [179]

```
#VER=6.8
VER=snapshots
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/miniroot68.fs
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/bsd
ftp https://cdn.openbsd.org/pub/OpenBSD/${VER}/luna88k/bsd.rd
nono -A bsd.rd
```

```
vmtype=luna88k
#luna-dipsw1=11111111 #ディップスイッチの初期値設定
#ethernet-hostdriver=tap
#prom-use-rom=0
ram-size=64
spc0-id6-image=hd,spc0-id6-image
```

うえーい、進んだー(° ▽ °)ー!! [173]
 MFP通過した。 [175]
 ROMやっど動いた(´Д`) [180]

- library aslr [181]

/etc/rc.conf.local に library_aslr=NO と書いておくと reordering libraries をスキップします。起動後であれば、# rcctl disable library_aslr でも良いです。man.openbsd.org/rc.conf

5.3. LUNA前夜 - 誕生と再生のためのテーマ

オムロン株式会社が1990年代初めに発売していたワークステーションLUNA。

「オムロンはインターネットの基礎となるUNIXにパイオニアとして取り組んできました。最初のマシンは1984年に出荷を開始した「スーパーメイト」と言うモトローラ社の68000をCPUとして使ったワークステーションでした。、当時サンマイクロシステムズが設立されたばかりで、10MHzのチップを使っていましたが、我々は国産の最新の12.5MHzのチップを使い、当時の最高速マシンとして登場しました。

当然UNIXを使うと自然にインターネットを使う事になるのですが、UNIXそのものが技術者しか使えないものだったため、技術者同士の通信手段として細々と使われていたに過ぎません。しかし、それでもUNIXの将来性に気づいていた我々は草の根的にオムロンの中でネットワークを拡大していきました。シグマワークステーション、LUNAワークステーションとマシンは進化していきましたが、その一方で「オムロン・インターネット」が着々と規模を拡大していきました。最終的には全国30個所以上、2000人が使うネットワークにまで成長しました。当時はだれもこれが「イントラネット」だと言う意識は無く、ひたすら便利なネットワークとして整備を続けました。もちろん今ではこれが典型的なイントラネットである事は間違いありません。しかし、この段階では依然として技術者専用の情報インフラでありました。草の根的な従って統率の無いものでした。」[50]

5.3.1. スーパーメイト

LUNAは、SX-9100以降の愛称なんですけど、その前のSX-8700の時代は、スーパーメイトという愛称でした。[149]

そしてソースリーディングを開始して約半年後、いよいよオムロン標準の16ビットボード(68000MPU)を改造して68451MMUを追加したボードコンピュータに、UNIXを移植する作業がスタートしました。開発環境は、EXORmacs上のIdrisを用い、移植中のOSのロードモジュールは、RS-232C接続で、実機にダウンロードしました。

そして、AH(アドバンスハード)プロジェクトと合流して開発したのが、68000搭載のUNIXワークステーションSX-8700でした。当時はUNIX System III注19であり、開発環境としてVAX-780上のSystem IIIを使いました。VAXと68000のバイトオーダーが逆だったので苦労しました。このマシンがスーパーメイトという商品名で、1984年春のマイコンショーで、オムロンのコンピュータとして華やかにデビューしたのです。そして間もなく発表されたUNIX System Vを移植し直し、その年の秋に出荷が開始されました。[150]

5.3.2. トラッカー台分

引越のため、収集していた古いワークステーションを廃棄している(しようとしている)ところです。そんななか、希少マシン?はNetBSDの移植に使用されていることを知り、事務局様を通じて、junk-jaへのポストをお願いしました。[11]

5.3.3. 1985年12月20日

SEA設立総会の夜。「1985年12月20日という日付は、日本のソフトウェアの歴史に永久に記録されよう。」(c)岸田さん 「詳しくはシグマせんとのこと」[54]

5.3.4. 1986年10月

「SIGMA サイドで作っていたオムロンのLUNA マシンは、そのころ、まだ影も形もなかった。わたしがソニーにアドバイスしたマーケティング戦略は、とりあえず最初のロットでできた何十台かのマシンを日本全国の大学の研究者に無料で配って使ってもらおうというもの。これもみごとに図にあたってと思います。」[53]

5.3.5. 1982年

「一九八二(昭和五七)年頃の話。ワークステーション開発部長をしていた市原達朗は、その利用法を考えているうちに、ワークステーションを使った産学連携を思いついた。数学のノーベル賞とも言われるフィールズ賞の受賞者・広中平祐をトップに、全国から一〇〇人の教授を選んで、一人一〇大、合計一〇〇〇台のワークステーションを寄付し、それをネットワークで結んで研究成果を無償解法してもらう。そしてそれを企業が事業に活用するというのがその趣旨だった。協力してもらうのは立石電機のほかに、東芝、日立製作所、IBMといった企業四社を想定していた。」[156]

5.4. LUNAシリーズ概要

5.4.1. SX-9100

1987年発表 for Σプロジェクト [36] [51] [86] 「札幌Σサブセンターに設置されて、地場企業によるΣ CAI ソフトウェアの開発に利用されていた。」[52]

5.4.2. LUNA [13]

- 1989年発表 MC68030 20MHz
- 起動動画 [45] [46] [38]

シリーズ構成 [65]					
ディスクレス	ベーシック	スタンダード	スタンダード	ファイルサーバ	ハイエンド
DT10	DT20/25	DT30/35	DT32/37	DT40/45	DT50/55
4MB	4MB	4MB	8MB	4MB	8MB
•	70MB	100MB	100MB	172MB	172MB
LAN	•	LAN	LAN	LAN	LAN
ブラック	ブラック	ホワイト	ホワイト	ホワイト	ホワイト
55万円	88/103万	115/135万円	140/160万円	165/185万円	190/210 万円

- PC98インターフェースはホワイトタイプに装備,DT20/25はオプション
- DT25,35,37は、フロッピーの代わりにテープストリーマ付き

5.4.3. オプション

- ビットマップボード:モノクロ:2048x1024,1プレーン
- ビットマップボード:カラー:2048x1024,4プレーン,4096色中16色/16階調同時表示
- 増設メモリボード:最大4MBx2枚
- LANボード:イーサネット,チーパネット(DT20/25)
- PC-98インターフェースボード

6. GPXボード: X.25,GP-1B

5.4.4. UNI-OS [37]

UniOS-B

Unix4.3BSDを移植したものをLunaで稼動。

UniOS-U

UnixAT&TSystemV R2.1をベースに4.2BSDの機能等を付加し、移植したものをLuna、Luna-IIで稼動。

UniOS-Σ

ΣOS-VOR1準拠したものをLuna-Σで稼動。(要出典:Luna-Σという呼び方) [83]

UniOS-Mach

Machをベースに移植したものをLuna-II、Luna-88Kで稼動。

本校のワークステーションはオムロン株式会社の「LUNA」(DT65及びFS180)というもので、CPUに「MC68030」(メインメモリー16MB)を、基本ソフトウェアに統合化OS(後述します)である「Unios-U」を採用した高性能なものです。(注釈:この部分を読めば、最近のコンピュータの進化が実感できますね) [66]

5.4.5. LUNAI

- 1991/6 MC68040 25MHz
- 68040を搭載したワークステーションLUNA-IIのハードウェア
- 互換性を重視し、CISC CPUを採用したワークステーションについて [21]
- カーネル起動問題

シリーズ構成 [73]			
DT2460	DT2465	DT2660	DT2665
8/16MB	8/16MB	8/16MB	8/16MB
250MB	250MB	250MB	250MB

- PC98インターフェースを2スロット装備
- DT2465,2665は、フロッピーの代わりにテープストリーマ付き
- イーサネット/チーパネット(標準)+イーサネット(オプション)

- ビットマップボード:モノクロ:2048x1024,1プレーン
- ビットマップボード:カラー:2048x1024,4プレーン,4096色中16色同時表示
- ビットマップボード:カラー:2048x1024,8プレーン,1670万色中256色同時表示

88Kでも88K2でも、hwplanebits(=ROMモニタのワークエリアの値)は、PW7131(8bpp) → 8 PW7102(4bpp) → 1 となる。 [101] [102] [103]

5.4.6. LUNA88k [10]

- モトローラRISC CPU MC88100(m88k)を採用
- マルチCPU対応(最大4つ) 1CPU時25Mips ,4CPU時100Mips
- Mach2.5,X11.4/X11R5(Luna88K2),Wnn4.1,Motif1.1.4
- PC-98用バス対応
- OpenBSDはm88kの実機とtoolchainがメンテナンスされている唯一のBSD
- 1992/9「マルチRISCワークステーションLUNA-88K2 - 33MHz MC88100 CPUを最大4個搭載したマルチRISCワークステーションについて」 [16]
- 88Kと88K2ではNVRAM/Timekeeperが違う。 [84]
- 起動動画 [30] [48] [49]
- ユニマガ紹介記事 [74] と、製品仕様 [75] と、まとめ [71]
- miod@openbsd.org さんのOpenBSD/luna88k ページ [90]
- LUNA-88K2 姉妹生存報告。10月にリリースされた #OpenBSD 6.8 [176]
- MC88100 バグ対応の一部 [177]

シリーズ構成 [73]	
DT8840	DT8860
8/16MB	32/64MB
250MB	250MB
270万円	350万円

5.4.7. omron3

omron3.sp.cs.cmu.edu (オムロン製 LUNA88k) は 1990年から1997年の間 CMU の日本語コンピュータ環境を提供するべくボランティア達によって運用されてきた計算機です。1997年5月をもって komachi.sp.cs.cmu.edu (Intel Pentium 120Mhz FreeBSD) に役目を引き継ぎ引退しました。 [68]

5.4.8. OEM版

「LUNAのシグマOSのやつで日本無線からでいていたOEMのワークステーションというやつを使っていたことがあります。なんか日本語フォントがX-Window立ち上げなくても使えたようなおぼろげな記憶。銀座にあったオムロンのセミナー会場で講習をうけたのだけど、そこではOEMでなくて普通のLUNAだったからなんかちがってとまどったような...」 [64]

5.4.9. LUNA2010 [78]

Introduces Omron Electronic BV's Luna 2010, a multiprocessor Unix workstation that supports configurations of up to four

88110 CPUs. Compatibility with Data General's DG/UX 5.4 Release 2.10 operating system; Other features; Prices. [79]

そして、新しいワークステーション用のチップセットのコードネームにAsteroidという名前をつけました。火星と木星の間にある無数にある小惑星群のことで。今から考えると、何でこんな名前を付けたのだろうかを反省してしましますが、とにかくそういう名前をつけてしまいました。[77]

- 88110
- 1993/9ごろ
- DC/UX5.4.X

5.5. OSCを中心とするイベント駆動開発

1. 動きそうなLUNAを探す
2. ハードウェアを整備する
3. ソフトウェアを書くために必要な情報を交換する
4. 行き詰ったらツイッターで相談する
5. 定期的にイベントで展示する
6. 昔使っていた人に直接話を聞く
7. 集めた情報を整理して、公開する

5.5.1. OSC2011Kansai@kyoto - LUNA復活

NetBSD/m68k will never die!

当日いきなりSun/NEWS/Luna展示 [31]

2011/7/16のコミットメッセージ [35]

Revive NetBSD/Luna68k.

Even after almost a lost decade since NetBSD/Luna68k was switched to using ELF format by default back in 2001, actually only one fix (bus.h) is required for a GENERIC kernel itself to get multiuser login: prompt on a real hardware. Hurrahh!!!

Demonstrated with a working Xorg mono server on the NetBSD booth at Open Source Conference 2011 Kansai @ Kyoto: <http://www.ospn.jp/osc2011-kyoto/>

"Very impressed," commented by Tomoko YOSHIDA, Program Committee Chair of the Conference, and some other OMRON guys.

Special Thanks to Tadashi Okamura, for providing a working SX-9100/DT "LUNA" for this mission.

5.5.2. なぜNetBSD/luna68kなのか

LUNAを使っていたわけでもないのになぜNetBSD/luna68kにこだわるのか。それはNetBSDのyamt-kmemブランチマージ作業の際の話にまで遡る。[91] [92]

違った yamt-km のほうだった orz [93] [94] yamt-km では hp300由来の m68k pmap でカーネルKVA用のセグメントテーブルをKVAの最上位に移動する必要があった。大部分のm68kではKVA空間上位は空いていたが luna68kだけは 0x4000000以降のデバイスアクセスにTTレジスタを使っていた。[95] で、hp300由来のpmapのセグメントテーブルとページテーブルの初期化は壮絶に何をやっているのかさっぱりわからない記述になっていて、かつ030と040は別の初期化が必要で、yamt-kmマージ当初はyamtさんがそれなりに書き換えただけで誰もテストしていなかったわけですよ [96] その後 NetBSD 4.0 が出る前に yamt-kmマージで動かなくなっていた atari を修正して、そのあとをm68k全部のpmap初期化をすべて読み解いてそれぞれのpmap bootstrap.cを初期化意図が読み取れるようにゴリゴリ書きなおしたわけなんですよ [97] 実機テストできない機種種のソースを4つも5つも書きなおして、1年後に見直すと致命的なtypoがあったりして、誰も持っていないマシンのコードなんか何の意味があるのか消してしまえなどと言われて、でもOSC2011京都で入手したluna68k実機ではそのままのソースで起動した、というお話 [98]

5.5.3. OSC2011Kansai@kyotoの波紋

- 「オムロンからLUNA88Kが発掘された」from よしだともこ先生 [29]
- 「LUNA88KはOpenBSD開発者の方へ」→あおやさんと連絡がとれる
- 「ツイッターで『LUNA-IIはありますか』とつぶやくと」
- 「ふと、横を見ると『LUNA II』と書いたマシンが。。。」

5.5.4. LunaII対応

1. 同じオペラントで68030と68040で違う命令の異
2. %ttl (PA/VA透過変換レジスタ)設定値修正
3. M68040共通部分の修正適用
4. 外付けSCSIアタッチ追加
5. LCD表示を「SX-9100/DT2」に変更

5.5.5. KOF2011 - LUNAII展示

- 「NetBSDが謎マシンを動かす理由＝そこに山があるからw」[24]
- 「明日11日(金)からのKOFのNetBSDブースで OSC京都のOMRON LUNA展示の後に発見された LUNA-IIで動くNetBSD/luna68k を展示します。」
- 「LUNA資料は手書きだ」
- 「NetBSD/luna68k画面表示の裏でひっそりと活躍する自作LUNA-II内蔵型B/WビデオtoVGA変換。」
- 「「そんなことよりそれはなんだ」と言われそうなLUNAならぬ初代SX-9100 Mr.文具セット。裏によしだ先生サイン(?)入りの貴重品。」

5.5.6. isibootd(8)

LUNA専用ネットブートサーバープログラムisibootd(8)をNetBSDツリーにコミット。

5.5.7. FPU判別ルーチン

1. ローエンド、ベーシックタイプは68881
2. サーバタイプは68882

5.5.8. OSC2012Kansai@Kyoto

1. NetBSD/luna68k近況 [58]
2. wsconsコンソールフレームバッファ修正 (OpenBSD/luna88kから)
3. オムロンフロントで表示
4. 電源トラブル:電解コンデンサ全交換
5. PROM起動仕様 HDDから起動する条件調査
6. bootarg問題
7. SSD on LUNA
8. Xorgサーバー

5.5.9. 円頓寺LUNAエンカウント

NBUG2013/2月例会。いきなりLuna68K/Luna88k/BigNEWSがNBUG例会にタクシーで持ち込まれる。「掲示板でLUNA88kをNBUG例会に持ち込もうか聞いている人がいる」と噂になっていたその人だった。[56]

5.5.10. OSC2013Tokushima

1. NetBSDこの20年 [55]
2. NetBSD/luna68kブートローダー起動展示 [8]

```
Module Name:      src
Committed By:    tsutsui
Date:            Sat Jan  5 17:44:25 UTC 2013
```

```
Added Files:
src/sys/arch/luna68k/include: loadfile_machdep.h
src/sys/arch/luna68k/stand/boot: Makefile autoconf.c bmc.c bmd.c boot.c
boot.ldscript conf.c cons.c device.h devopen.c disklabel.c font.c
getline.c init_main.c ioconf.c kbd.c kbdreg.h locore.S machdep.c
omron_disklabel.h parse.c preset.h prf.c rcvbuf.h romcons.c
romvec.h samachdep.h sc.c screen.c scsi.c scsireg.h scsivar.h sd.c
sio.c sioreg.h status.h stinger.h trap.c ufs_disksubr.c vectors.h
version
```

```
Log Message:
First cut at NetBSD/luna68k native bootloader.
```

```
Based on 4.4BSD-Lite2/luna68k "Stinger" loader revision "Phase-31"
http://svnweb.freebsd.org/csr/sys/luna68k/stand/
and MI libsa glue stuff are taken from hp300 etc.
```

```
Tested on LUNA-I and old DK315C SCSI disk drive.
```

```
LUNA's monitor PROM can load only an a.out binary in 4.3BSD FFS partition
(i.e. created by "newfs -0 0") on disks with OMRON's UniOS disklabel,
but now we can load an ELF kernel in root partition via this bootloader.
(See luna68k/disksubr.c for details of UniOS label)
```

```
TODD:
- LUNA-II support (check 68040 to adjust cpuspeed for DELAY())
- secondary SCSI support for LUNA-II
- netboot via le(4) (should be trivial)
- support boot options on bootloader prompt
- bootinfo (passing info about booted device and kernel symbols)
- support "press return to boot now, any other key for boot menu" method
  like x86 bootloader (needs cnsan() like functions)
- tapeboot (anyone wants it?)
```

5.5.11. OSC2013Nagoya - Luna88K&Luna68K

- あおやまさんと江富さんによるLuna88K/Luna68K完全動態展示 [57]
- Luna88K2 & Luna68K プロトタイプ7号機
- OpenBSD/luna88k開発者あおやまさんによるプレゼンテーション [10]

5.5.12. OSC2013 Kansai@Kyoto

非力なマシンで最新のOSを動かすためには、大変な努力と工夫が必要です。その展示を実現させた方は、その努力と工夫を楽しんでおられたというわけです。[100]

5.5.13. NBUG 2013/9

- OpenBSD/luna88k 近況報告

5.5.14. KOF2013

- 関西オープンソース2013NetBSDブースの記録 [117]

5.5.15. OSC2014 Kansai@kyoto

- OSC2014京都 NetBSDブース展示への道 [134]
- OSC2014京都 NetBSDブースの記録 [135]

5.5.16. LUNAグッズ

- LUNAグッズ持参でブースに遊びに来てくれる元関係者の方が！
- シールとフロッピー [59]
- たれまく
- ベンセット [60]
- ベンケースとバンダナ [76] [121] [130]
- dpNote - 図形グッズ:シール・定規
- ホッチキス [133]
- トレーナー [144]
- LUNAII ボールペン

5.5.17. LUNA関連書籍

LUNAユーザーグループとは何か - mikutterの薄い本 [81] を会場に忘れたら、なぜか一緒に送られてくる
UNIXワークステーションがわかる本 [61]

@tsutsui LUNAの薄い本2013作れってことすね [7]

5.6. LUNAについて私が知っている二、三の事柄

5.6.1. Project Mach

Project Mach was an operating systems research project of the Carnegie Mellon University School of Computer Science from 1985 to 1994.

"It's never too late. When it's over, you get to tell the story" -- Garrison Keillor [67]

- luna88kカーネルソース [72]

5.6.2. LUNAインストール方法

- インストールマニュアル [27]
- NetBSD/luna68kの起動ディスク作り方メモ [28]

5.6.3. NetBSD/luna68Kのブート方法 [13]

2種類のブート方法: [27]

1. PROMがUNIOS-B /a.outをロードする
2. PROMが独自プロトコルでサーバからカーネルロード

- NetBSD1.5以降 m68kはELFフォーマット移行:どうやって起動するか
- 実機がないままソースツリーはメンテされ続ける

5.6.4. PROMモニタ

1. newfs -O o で作った4.3BSD形式のFFS上のa.outカーネルは読める。
2. LUNAIIでのネットブートは無理？
3. HDD起動時の制約は？(SCSI ID,カーネルサイズ,ファイルシステム)
4. LUNAIIは外部SCSI HDDから起動できるか

5.6.5. LUNA68Kのブートローダー

1. NetBSD/luna68k の起動ディスクの作り方メモ [28]
2. なんかも出た。これでデバッグできる [8]
3. native bootloader update for NetBSD/luna68k [116]

5.6.6. LUNA88Kのブートローダー

1. OpenBSD/luna88k standalone bootloader by @MiodVallat works fine on my LUNA-88K2! [107]

5.6.7. BSD広告条項

4.4BSD-Lite2由来のコードに含まれる3項目(All advertising materials ..)、広告条項削除OKについて、文書で許可を出してくれるOMRONの方がいらっしやると2-clause BSDで配布できる。

5.6.8. 電源問題

1. OMRONワークステーション LUNA-II 電源ユニット修理記 [25]
2. 「KOF本番週の日曜日に電源が不調になり急遽部品手配して展示直前に修理していた」
3. OSC2012京都前に再度補修 [26]
4. LUNA-II, LUNA-88K 電源ユニット(PTD573-51) 四級塩電解コンデンサー一覧 [88]
5. 88K2は88Kより分解しやすいような気がする。[127]

5.6.9. LUNAI

1. 1MB SIMM/4MB SIMM切替→SIMM脇に謎ジャンパが
2. 4bppフレームバッファのX11R5ソースは? [85]

5.6.10. PC98-Cバス

1. 86音源ボード on LUNA [138]
2. C-bus拡張ボード on LUNA-88K2 [146]

5.6.11. UniOS-Machと西暦2000年問題

でも、同じマシンでUniOS-Mach立ち上げると時刻が変になる。昔調べたとおりdate(8)コマンドでは2000年以降の日付は設定できないので、OS内部で元々Y2K対応が甘いだけかもしれない。[82] [84]

5.6.12. 質問日時:2009/10/28 17:51:08

会社でワークステーション(オムロン製LUNAI)を使用していますが、プリンターが不良となりました。エプソンVP-4000です。中古をさがしていますが、これと互換性のあるプリンターはないでしょうか? [14]

もう捨てようかと思う...が、未練があり捨てられず [10]

Sometimes you get so lonely / Sometimes you get nowhere / I've lived all over the world / I've left every place / Please be mine / Share my life / Stay with me / Be my wife [5]

5.7. LUNA年表 - 月の刃

「次の日からオレとルナ先生の生活がはじまったんだ」 [62]

青:あおやまさん,江:江富さん,オ:オムロン,筒:筒井さん,菅:菅原さん,モ:モトローラ

1986/10		いけない!ルナ先生連載開始
1987	Σ	SX-9100 オムロンから発表 [47]
1988	オ	グッドデザイン賞受賞 [15]
1988/7		いけない!ルナ先生連載終了
1989	オ	SX-9100/DT LUNA MC68030 20MHz
1989/3	オ	LUNAのハードウェア Omron Tech No.29 p.8-15 [19]
1990/7	オ	Luna88k Omron Tech p.81-92 [20]
1991/6	オ	LUNA-II Omron Tech No.31 p. 91-9 [21]
1991/10/11		春奈るなさんの誕生日
1991/11	モ	MC88110の存在が明らかになる [17]
1992/12	オ	LUNA-88K2 Omron Tech No.32 p.336-344 [16]
1992/12	オ	MC88110ワークステーション Omron Tech No.32 p.345-350 [18]
1993/9	オ	LUNA2010
1994		いけない!ルナ先生 復刻版
1994		4.4BSD Lite luna68k
1994	オ	LUNA-IIの生産終了
1994/3	オ	LUNA2010用システム診断プログラムの開発について [80]
1998/6		NetBSD/luna68k やってるひと、いますよ。[netbsd 02006] [23]
1999/12		NetBSD/news68kマージ
2000/1/6		NetBSD/luna68kマージ
2000		いけない!ルナ先生 復刻版
2000/2/18		榎田さんluna68k起動成功 [netbsd 05132] [22]
2000/08	青	LUNA-88K2入手
2001/12	青	LUNAにOpenBSD移植決意
2002/01/27	青	OpenBSD/sparc上でm88kクロス環境構築
2002/03/29	青	シリアルコンソールでカーネルCopyright表示
2002/06/05	青	network bootでIPアドレス取得 [69]
2003/08	青	コンパイラのバグがなおったようなので再開
2003/09/20	青	tarのオプションを間違えてソース消去、CVS導入
2003/10/05	青	なんとかもとの状態に戻る
2003/12/10	青	NFS rootでシングルユーザ&マルチユーザ
2004/02/17	青	SCSI動作
2004/03/21	青	Miod Vallatさん(OpenBSD/mvme88kのport maintainer)に見つかる [70]
2004/04/21	青	OpenBSD本家treeにcommit
2004/11/01	青	OpenBSD 3.6: 初の正式リリース
2007/08/31	青	LUNA-88K2の電源ユニット故障により起動できなくなる
2007/9/5	青	Luna88K検索願ひ、[nbug:10540]

2009/10/28		Yahoo知恵袋にLUNAIIに関する]質問が [14]
2011/07	筒	OSC2011Kansai@KyotoでLUNA/NEWS/Sun3展示 [31]
2011/7	筒	NetBSD/luna68k on OMRON LUNA - Bootstrap [37]
2011/07	オ	LUNA88Kオムロンにて発掘される!
2011/08	青	ご好意により、オムロンで発掘されたLUNA-88Kが届く
2011/08/15	青	上記LUNA-88Kの電源ユニットを移植して復活
2011	筒	KOFでLUNA-II展示 [9]
2012/05/01	青	OpenBSD 5.1: 久しぶりの正式リリース
2012/02/28	青	10年目にして一応マルチプロセッサ対応
2012/08/03	筒	OSC2012関西@京都でLuna&LunaII展示 [32]
2013/01	青	OpenBSD m88k portのELF化&共有ライブラリ化
2013/01/27		Luna88k(白と黒) Luna88K2Luna2010を青山さんに送る
2013/02/16	江	Luna88K&Luna68k&BigNEWSをNBUG例会に持ち込む
2013/03/09	江	Luna88kをあおやまさんに送る
2013/03/09	筒	OSC2013徳島でLuna68K展示 [33]
2013/03/19	筒	筒井さんから江富さんにLuna68k起動ディスクが送られる
2013/04/13	青	Monochrome X serverが動作
2013/06/22	青	Luna88K OSC 2013 Nagoyaで初展示。江富さんのLuna68kも初展示。 [119]
2013/08/2	筒	OSC2013京都でLuna/LunaII tw/mikutter展示。 [120]
2013/08/24	筒	OSC2013島根でLunaII+mlterm-fb+mikutterd展示 [118]
2013/09/21	青	OpenBSD/luna88k近況報告 無印/4bpp/リリース [109]
2013/11/8-9	筒	KOF2013 NetBSDブースでLuna+mikutter展示 [117]
2013/12/21	青	OpenBSD/luna88k近況報告 PC98バス 音源ボード [110]
2014/01/11	筒	native bootloader update for NetBSD/luna68k [129]
2014/03/05		いけない! ルナ先生 DVD全6巻発売開始 [114] [115]
2014/04/19	青	NBUG2014/4例会 OpenBSD/luna88k 2014/4近況報告 [125]
2014/07/05	青	OSC 2014 NagoyaでLuna88K+PC98 86音源ボード展示 [122] [123] [124]
2014/07/13	筒	LunaII+8bpp ボードでカラー表示 [126]
2014/07/20	筒	LUNA's keyboard driver changes from OpenBSD/luna88k [131]
2014/07/21	筒	luna68k 4/8bpp framebuffer as a monochrome server [132]
2014/08/01	筒	lunaII+mikutterd今年はカラーだ展示 [134]
2014/08/13	筒	LUNA framebuffer mod for LCDs without Sync on Green support [142]
2014/11/07-08	筒	関西オープンソース2014 NetBSDブース展示の記録 [143]
2014/11/29	青	yaft × LUNA [140]
2014/12/20	青	86音源ボード on LUNA [138]
2014/12/29	青	86音源ボードコードコミット [139]
2015/02/21	青	LUNA-88K2はPCカードの夢を見るか? [141]
2015/03/21	青	LUNA-88K2はPCカードの夢を見るか? <補遺> [145]
2015/05/01	青	OpenBSD 5.7リリース [148]
2015/05/22	青	C-bus拡張ボード on LUNA-88K2 [146]
2015/07/06		「いけない! ルナ先生」コラボ読切で復活 [147]
2015	青	OpenBSD/luna88k移植物語 [152]
2015/10	青	FM音源の調べ on LUNA http://www.slideshare.net/ao_kenji/nbug201510
2016/3	江	Luna68K AsiaBSDCon2016ブース展示
2016/7	筒	Luna68K PSG Z80 sound driver for PC6001 to NetBSD/luna68k OSC京都 [164]
2016/11	藤	Implimentation of 4.4BSD luna68k by Akito Fujita KOF
2016/11	筒	Luna68K PSG Z80 sound driver for PC6001 to NetBSD/luna68k OSC広島
2017/3	青江	Luna88K&Luna68K AsiaBSDCon2017ブース展示
2018/5	筒	RaSCSI + OMRON 初代LUNA 起動
2018/5	青	LUNA-88K2 OSC2018Nagoya ブース展示
2018/8	菅	LUNA68K OSC2018Kyoto ブース展示 LUNAのPSG音源でPCM再生
2018/8	筒	LUNA68K OSC2018Kyoto ブース展示 sayaka+mlterm-fb
2018/7	青	LUNA-88K2 OSC2018Nagoya ブース展示 [165]
2020/4		オムロン元社長・立石義雄氏逝去 [172]
2020/5	井	nono-0.0.3リリース [167]
2020/7	井	nono-0.1.0リリース
2021/5	青	Here come LUNA-88K emulators! [182]
2021/6	井	nono-0.2.0リリース [181]
2022/4	井	nono-0.3.0リリース
2022/5	筒	psgpam(4)マージ [183]
2022/6	井	nono-0.4.0リリース NetBSD/x68kブート
2022/7	井	nono-0.4.1リリース [185]
2022/7	筒	NetBSD/luna88k シングルユーザーブート [184]
2022/9	井	lunafb(4) improvement [186]
2022/10	井	nono-0.4.4リリース [187]
2023/1	井	nono-0.5.0リリース [188]
2023/2	井	nono-0.5.1リリース [189]
2023/2	青	LUNA88K Boot 0.8 [190]

5.8. 最近のLUNA

5.8.1. メモリ64M

というわけで LUNA-IIの64MB設定でも NetBSD/luna68kカーネル起動した。これでしばらく耐久テストするか… [12] [34]

5.8.2. mlterm-fb & tw

ツイッタークライアント! [39] [40] [41] [42] [43] [44] [111]

5.8.3. mlterm-fb + mikutterd

LunaIIならmlterm-fbとmikutterdを組み合わせてタイムラインを展示できます。

5.8.4. 画像の2値化

モノクロ画面で効率的にデモ画面を作る方法:(サーベイする)

5.8.5. LUNA-88K:NVRAM and Timekeeper registers

On 'original' LUNA-88K, NVRAM contents and Timekeeper registers are mapped on the most significant byte of each 32bit word. (i.e. 4-bytes stride) Also, add small 'Wrong year set by UniOS-Mach after Y2K' hack. [89]

5.8.6. LUNA88K謎ボード

- PWB7183 [99]
- 専用チップが載っている [102]

5.8.7. KOF2014におけるLUNA展示

LUNAフォントとSONYフォント [137]

5.8.8. yaft X LUNA

yet another frame buffer terminal [136]

5.8.9. 老ハード介護問題

- 電源修理
- SCSI HDD確保
- ブラウン管を知らない子どもたち
- 3ボタンマウスを知らない子どもたち [128]
- 液晶接続問題 [63]
- ハード保守
- 詳しい人がいなくなる

「それなら、なぜ、先祖代々の墓を守って山間に生活したがる農民を、ダム工事のためにおいだすんだね?それぞれの人間にそれぞれの幸福がある。それなら、なぜ、彼らを一般化の中に投げ込むんだ。君はなぜ、そういう役割をひきうけるんだ?」 [6]

5.9. 脚注

Luna関連コメントは筒井さんに紹介してもらったものです。

[1] カフカ式練習帳 http://www.bunshun.co.jp/cgi-bin/book_db/book_detail.cgi?isbn=9784163813301

[2] The Angel Levine: <http://www.blackmovie-jp.com/movie/angellevine.php?act=a#.Uei719f75z0>

[3] Look Back in Anger: http://en.wikipedia.org/wiki/Look_Back_in_Anger_%28song%29

[4] バーナード・マラマッドに関する研究 <http://www.ishikawa-nct.ac.jp/lab/G/koguma/www/ehp/suzukihp.pdf>

[5] Be My Wife: http://en.wikipedia.org/wiki/Be_My_Wife

[6] 高橋和巳『散華』論 -生活人としての大家-, 東口昌央, 1988, <http://ir.lib.osaka-kyoiku.ac.jp/dspace/handle/123456789/15270>

[7] <https://twitter.com/ebijun/status/231983148118970368>

8(1,2) NetBSD/luna68kブートローダー実装作業日記, 2013/1/4 <http://togetter.com/li/433650>

[9] NetBSDブース @ 関西オープンソース 2011, 2011/11/13 <http://togetter.com/li/213724>

10(1,2,3) OpenBSD/luna88k on LUNA-88K2 <http://www.nk-home.net/~aoyama/osc2013nagoya/OpenBSD-luna88k.pdf>

[11] トラッカー一台? <http://www.jp.netbsd.org/ja/JP/ml/junk-ja/201301/msg00005.html>

[12] というわけで <https://twitter.com/tsutsui/status/357219819289985024/photo/1>

13(1,2) 展示マシン紹介(3) <http://www.ceres.dti.ne.jp/tsutsui/osc2011kyoto/>

14(1,2) Yahoo!知恵袋 http://detail.chiebukuro.yahoo.co.jp/qa/question_detail/q1132299146 まだ動いていたのか! 人々に勇気を与えた質問。

- [15] GOOD DESIGN AWARD <http://www.g-mark.org/award/describe/15097> ... 価格にマルが一つ足りない
- 16(1,2) 「マルチRISCワークステーションLUNA-88K2 - 33MHz MC88110 CPUを最大4個搭載したマルチRISCワークステーションについて」
http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902066730130379 ... これも1992年12月でさっきのと同じだから同時進行だったのかな
- [17] MC88110とは <http://en.wikipedia.org/wiki/MC88110> "... the MC88110 was ultimately unsuccessful and was used in few systems."
諸行無常
- [18] 「MC88110を搭載したワークステーションのハードウェア - 64bits,Superscalerを採用したMC88110 CPUを搭載したワークステーションのハードウェアについて」
http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902068908677809 ... 幻のLUNA88K3計画なのだろうか
- [19] 「LUNA(デスクトップWS)のハードウェア 従来機に比べて小形化,低価格化を実現したハードウェアについて」
http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902088071486407
- [20] 「マルチプロセッサワークステーションのハードウェア - RISCマルチプロセッサのワークステーションへのインプリメンテーション技術について」
http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902066853173587 実はLUNA88Kの設計のほうがLUNA-IIより先なんですよ
- 21(1,2) 「68040を搭載したワークステーションLUNA-IIのハードウェア」
http://jglobal.jst.go.jp/detail.php?JGLOBAL_ID=200902048488142806
..., Omron Tech 31巻 2号 91-97頁, 1991年06月
- [22] 古文書に見る現実逃避パワー <http://www.re.soum.co.jp/~fukaumi/ml/netbsd/200002/msg00122.html> ...
- [23] 古文書に見る謎の痕跡 <http://www.re.soum.co.jp/~fukaumi/ml/netbsd/199806/msg00068.html>
- [24] なぜ山に登るのか <http://www.ceres.dti.ne.jp/tsutsui/kof2011/Why-enigmatic-machines.html>
- [25] OMRONワークステーション LUNA-II 電源ユニット修理記 <http://togetter.com/li/215988>
- [26] OMRONワークステーション LUNA-II 電源ユニット修理記 その2 <http://togetter.com/li/354562>
- 27(1,2) インストール方法 <http://www.jp.netbsd.org/ports/luna68k/install.html>
- 28(1,2) NetBSD/luna68k の起動ディスクの作り方メモ <https://gist.github.com/tsutsui/5196134> とかですが netboot の説明を isibootd(8) に合わせて更新するのをさぼっている(プーメラン)
- [29] LUNA88K,オムロンにて発掘される! <http://www.jp.netbsd.org/ja/JP/ml/port-mac68k-ja/201107/msg00011.html>
- [30] Luna88k 起 動 動 画 https://twitter.com/ao_kenji/status/353469599871799296 https://twitter.com/ao_kenji/status/353476705521905664
- 31(1,2) "NetBSD/m68k will never die!" <http://www.ceres.dti.ne.jp/tsutsui/osc2011kyoto/NetBSD-m68k-will-never-die.html>
- [32] OSC2012京都NetBSDブース展示の記録 <http://togetter.com/li/350035>
- [33] OSC2013徳島 NetBSDブース展示の記録 <http://togetter.com/li/468577>
- [34] OMRONワークステーションLUNA 工作日記 <http://togetter.com/li/535307>
- [35] コミットメッセージ <http://mail-index.netbsd.org/source-changes/2011/07/16/msg024675.html>
- [36] Σプロジェクト <http://ja.wikipedia.org/wiki/Σプロジェクト>
- 37(1,2) [http://ja.wikipedia.org/wiki/Luna_\(ワークステーション\)](http://ja.wikipedia.org/wiki/Luna_(ワークステーション))
- [38] NetBSD/luna68k on OMRON LUNA - Bootstrap http://www.youtube.com/watch?v=c1_e-A9Osr0
- [39] Twitter timeline on NetBSD/luna68k and mlterm-fb (final) http://www.youtube.com/watch?v=djBwE0G_LMI 2013/5/24
- [40] mlterm-fb demonstration on NetBSD/luna68k (revised) <http://www.youtube.com/watch?v=BP8AIceWgxA> 2013/5/18
- [41] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 4) http://www.youtube.com/watch?v=yKKT_Z1P9Xo 2013/05/04
- [42] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 3) <http://www.youtube.com/watch?v=C1ICaO5scHY> 2013/05/01
- [43] Twitter timeline on NetBSD/luna68k and mlterm-fb (take 2) <http://www.youtube.com/watch?v=8sC5XpK-Hxs> 2013/04/29
- [44] Twitter timeline on NetBSD/luna68k and mlterm-fb <http://www.youtube.com/watch?v=nzD0A279mcg> 2013/04/27
- [45] NetBSD/luna68k on OMRON LUNA - Start X.Org <http://www.youtube.com/watch?v=NRh60c420Mc> 2011/07/30
- [46] mlterm-fb demonstration on NetBSD/luna68k wsfb console <http://www.youtube.com/watch?v=jHU876RexCo> 2013/04/25
- [47] Σワークステーション(SX-9100)の概要 <http://jglobal.jst.go.jp/public/20090422/200902068890346915> 1987/9/30 Omron Tech p.207-213
- [48] OpenBSD/luna88k on OMRON LUNA-88K2 - bootstrap screen <http://www.youtube.com/watch?v=btwiiZw3B2s> 2013/07/06
- [49] OpenBSD/luna88k on OMRON LUNA-88K2 - starting X.org http://www.youtube.com/watch?v=_EUpSpUD0Qw 2013/07/06
- [50] オムロンのイントラネットの歴史 <http://www.masuda.org/intra/rekisi.html>
- [51] [IT]日本のITの歴史—SONY『NEWS』の戦略(3)(1989-03-20 <http://www.miraikeikaku-shimbun.com/article/13282000.html>
- [52] さっぽろコンピュータ博物館 <http://www.sec.or.jp/elecen/museum/>
- [53] 歴史的コンピュータとソフトウェアプロジェクトに関する昔話(社外公開版) <http://katsu.watanabe.name/doc/comphist/>
- [54] SEA Mail Vol.1 No.1 http://www.sea.jp/office/seamail/1986/1986_1_honan.pdf
- [55] NetBSDこの20年 <http://www.slideshare.net/tsutsui/osc2013tokushima-net-bsd20th>
- [56] 名古屋*BSDユーザグループ(NBUG)2013/2月例会の記録 <http://togetter.com/li/456972>

- [57] OSC2013名古屋 NBUG&NetBSDブース展示の記録 <http://togetter.com/li/522396>
- [58] NetBSD/luna68k 近況 <http://www.ceres.dti.ne.jp/tsutsui/osc2012kyoto/NetBSD-luna68k-updates.html>
- [59] LUNAシールとフロッピー <http://movapic.com/pic/2013062214270151c535a5bd627>
- [60] LUNAペンセットと本 <https://twitter.com/tsutsui/status/135565130372104192>
- [61] UNIXワークステーションがわかる本 (LUNAの本シリーズ) <http://www.amazon.co.jp/dp/4526029963>
- [62] いけない!ルナ先生 全5巻 http://www.comicpark.net/readcomic/index.asp?content_id=COMC_AKC01155_SET
- [63] PS2Linux Kit(Sync on Green)対応モニター一覧 <http://www.ps2linux.dev.jp/monitor.html>
- [64] かやまさん https://www.facebook.com/jun.ebihara.18/posts/692735874076690?comment_id=30643585&offset=0&total_comments=1
- [65] LUNAのカタログ Holonic Workstation LUNA[マニュアル・データシート類] <http://www.h2.dion.ne.jp/~dogs/collect/ds/luna.html>
- [66] ワークステーション操作入門 http://www.kumamotokokufu-h.ed.jp/kokufu/comp/ws_tx1.html
- [67] "It's never too late. When it's over, you get to tell the story" -- Garrison Keillor <http://www.cs.cmu.edu/afs/cs/project/mach/public/www/mach.html>
- [68] ピ ッ ツ バ ー グ 便 利 帳 サ ー バ ー の 歴 史 <http://komachi.sp.cs.cmu.edu/benricho/Komachi#.E3.82.B5.E3.83.BC.E3.83.90.E3.83.BC.E3.81.AE.E6.AD.B4.E5.8F.B2> 「1993年、オムロンのワークステーション業務撤退に伴い藤田さんと作業マシン達に突然の引き上げ命令が下りました。」
- [69] OpenBSD/luna88k「network bootでIPアドレス取得」(2002/06/05)の頃のページ <http://t.co/VRxXgWWpTO>
- [70] 同じく「Miod Vallatさんに見つかる」(2004/03/21)の頃のページ <http://t.co/3QmzWm7reR>
- [71] OMRON Luna88Kについてのまとめ <http://t.co/rt5kUB74VG> 作者も忘れてる説
- [72] <http://www.cs.cmu.edu/afs/cs/project/mach/public/src/mkernel/src/kernel/luna88k/>
- 73(1,2) LUNA-II スペック表というサーベイ漏れ <https://t.co/KV9f6XS8bU>
- [74] ユニマガのluna88k発売の記事。 <https://twitter.com/alkawa/status/360427576717611008>
- [75] LUNA-88K2 の製品仕様 プロセッサ以外は同じという見方もある <https://twitter.com/tsutsui/status/361463750982778880/photo/1>
- [76] もうひとつあった。2011年OSC京都 わざわざ2日目に持ってきていただいた超重要LUNAグッズ ペンケースとバンドナ <http://movapic.com/ebijun/pic/3812352> たしか、来場者の方の奥様の所有で、「持って行くのはいいけれど絶対に持って返ってくるように」と申し渡された、というお話だったような
- [77] 名は体を表す <http://ameblo.jp/hirokun39/entry-11345138649.html>
- [78] LUNA2010 Good Design Award <http://www.g-mark.org/award/describe/20641>
- [79] Omron spins four 88110s at Data General Aviiion line <http://connection.ebscohost.com/c/articles/9402180800/omron-spins-four-88110s-data-general-aviion-line>
- [80] システム診断プログラムの開発 LUNA2010用システム診断プログラムの開発について <http://jglobal.jst.go.jp/public/20090422/200902172571690192>
- [81] mikutterの薄い本製作委員会 <http://home1.tigers-net.com/brsywe/mikutter.html>
- [82] https://twitter.com/ao_kenji/status/360775880198459394/photo/1
- [83] Wikipediaの「LUNA-Σ」という呼称は果てしなく要出典という感想。 <https://twitter.com/tsutsui/status/360430992638492672>
- 84(1,2) "RTC" の stamp のオフセットをそれぞれ x4してやればいいような気がします <https://twitter.com/tsutsui/status/360418015600312320>
- [85] まずはDIP SW操作してみて変わるかどうか <https://twitter.com/tsutsui/status/360416804876722177>
- [86] マンガソフトウェア革命—Σプロジェクトの全貌 <http://www.amazon.co.jp/dp/4339022543>
- 87 仁和寺 <http://randen.keifuku.co.jp/map/17.html>
- [88] <https://gist.github.com/tsutsui/6203477> OMRON LUNA-II および LUNA-88K の電源ユニットに使用されている要交換な四級塩電解コンデンサのリスト。
- [89] <http://marc.info/?l=openbsd-cvs&m=137617369920936>
- [90] miod@openbsd.org さんのOpenBSD/luna88k resource page <http://gentiane.org/~miod/software/openbsd/luna88k/>
- [91] <https://twitter.com/tsutsui/status/365121355001237505>
- [92] <http://nrx.netbsd.org/xref/src/doc/BRANCHES?r=1.330#623>
- [93] <https://twitter.com/tsutsui/status/365121528309891072>
- [94] <http://nrx.netbsd.org/xref/src/doc/BRANCHES?r=1.330#1611>
- [95] <https://twitter.com/tsutsui/status/365121928526184448>
- [96] <https://twitter.com/tsutsui/status/365122443951616001>
- [97] <https://twitter.com/tsutsui/status/365122859305140225>
- [98] <https://twitter.com/tsutsui/status/365123833402896384>

- [99] https://twitter.com/ao_kenji/status/366154076565680128/photo/1
- [100] 20年前のコンピュータで最新のOSを動かす意味とは? <http://notredameningen.kyo2.jp/e422862.html>
- [101] 88Kと88K2のグラフィックボードを交換して調査。 https://twitter.com/ao_kenji/status/366066990093303809
- 102(1,2) 専用のゲートアレイでしょうか。 https://twitter.com/ao_kenji/status/366056571609939968
- [103] PWB7102 https://twitter.com/ao_kenji/status/366009479285854208
- 104 UniOS-Machを参考に1bpp/4bpp/8bppを自前で識別するようにした。 https://twitter.com/ao_kenji/status/368294458996948992
- 105 Luna88k マニュアル https://twitter.com/ao_kenji/status/395857381818519552
- 106 Luna88k FaceBook ページ <https://www.facebook.com/Luna88k>
- [107] OpenBSD/luna88k standalone bootloader by @MiodVallat works fine on my LUNA-88K2! https://twitter.com/ao_kenji/status/395551245563219969
- 108 LUNA-88KのPC-98拡張バス(Cバス)についてのメモ <https://gist.github.com/ao-kenji/7843096>
- [109] OpenBSD/luna88k近況報告 NBUG 2013/9 http://www.slideshare.net/ao_kenji/openbsdluna88k-news-at-nbug-meeting-2013
- [110] OpenBSD/luna88k近況報告 NBUG 2013/12 http://www.slideshare.net/ao_kenji/nbug201312
- [111] test tweet from OpenBSD/luna88k https://twitter.com/ao_kenji/status/482151248502591488
- 112 OMRONワークステーションLUNA 工作日記 <http://togetter.com/li/535307>
- 113 OMRONワークステーション LUNA-II 電源ユニット修理記 その3 <http://togetter.com/li/548989>
- [114] 「いけない! ルナ先生」実写化! 6人のアイドルが先生に <http://natalie.mu/comic/news/105048>
- [115] <http://www.cinemart.co.jp/ikenai-runa/>
- [116] native bootloader update for NetBSD/luna68k <http://mail-index.netbsd.org/port-luna68k/2014/01/11/msg000038.html>
- 117(1,2) 関西オープンソース2013NetBSDブース展示の記録 <http://togetter.com/li/587422>
- [118] オープンソースカンファレンス2013島根 NetBSDブース展示の記録 <http://togetter.com/li/553529>
- [119] OSC2013名古屋 NBUG&NetBSDブース展示の記録 <http://togetter.com/li/522396>
- [120] オープンソースカンファレンス2013関西@京都 NetBSDブース展示記録 <http://togetter.com/li/542885>
- [121] 泣いて喜びそうなもの発掘 https://twitter.com/goinkyoko_hacker/status/482528142930620416
- [122] OpenBSD/luna88kのご紹介 http://www.slideshare.net/ao_kenji/osc2014-nagoya
- [123] OpenBSD/luna88kのご紹介 パンフレット <http://www.nk-home.net/~aoyama/osc2014nagoya/obsd-luna88k-leaflet.pdf>
- [124] オープンソースカンファレンス2014 Nagoya&NBUG7月例会 の記録 <http://togetter.com/li/688742>
- [125] OpenBSD/luna88k 2014/4近況報告 http://www.slideshare.net/ao_kenji/nbug201404
- [126] Setup Bt458 color palette to support ANSI color text on 8bpp framebuffer. <http://mail-index.netbsd.org/source-changes/2014/07/13/msg056309.html>
- [127] 88K2電源交換 https://twitter.com/ao_kenji/status/485393846314872832
- [128] どうしてこのマウスにはボタンが3つもあるのか https://twitter.com/ao_kenji/status/485275421768814592
- [129] native bootloader update for NetBSD/luna68k <http://mail-index.netbsd.org/port-luna68k/2014/01/11/msg000038.html>
- [130] 泣いて喜びそうなバンダナ https://twitter.com/goinkyoko_hacker/status/482528142930620416/photo/1
- [131] LUNA's keyboard driver changes from OpenBSD/luna88k <http://mail-index.netbsd.org/source-changes/2014/07/20/msg056548.html>
- [132] luna68k 4/8bpp framebuffers as a monochrome server <http://mail-index.netbsd.org/source-changes/2014/07/21/msg056590.html>
- [133] LUNAホットキス https://twitter.com/goinkyoko_hacker/status/497392417478156288/photo/1
- 134(1,2) OSC2014京都NetBSDブース展示への道 <http://togetter.com/li/703494>
- [135] OSC2014 Kansai@Kyoto NetBSDブース展示の記録 <http://togetter.com/li/700617>
- [136] yaft×Laan http://www.slideshare.net/ao_kenji/nbug201411-yaft-42177561
- [137] KOFにおけるLUNA展示 <https://speakerdeck.com/tsutsui/kof-and-luna-at-netbsd-booth>
- 138(1,2) PC-9801-86 sound board on LUNA http://www.slideshare.net/ao_kenji/nbug201412
- [139] OpenBSD/luna88k用の86音源ボードドライバを整理してcommit. https://twitter.com/ao_kenji/status/549203137001553921
- [140] yaft×LUNA http://www.slideshare.net/ao_kenji/nbug201411-yaft-42177561
- [141] LUNA-88K2はPCカードの夢を見るか http://www.slideshare.net/ao_kenji/nbug201502
- [142] LUNA framebuffer mod for LCDs without Sync on Green support <http://mail-index.netbsd.org/port-luna68k/2014/08/13/msg000043.html>

- [143] 関西オープンソース2014 NetBSDブース展示の記録 <http://togetter.com/li/742243>
- [144] Lunaトレーナー <http://movapic.com/ebijun/pic/5232493>
- [145] LUNA-88K2はPCカードの夢を見るか <補遺> http://www.slideshare.net/ao_kenji/nbug201503
- 146(1,2) C-bus拡張ボード on LUNA-88K2 http://www.slideshare.net/ao_kenji/osc2015-nagoya/
- [147] 伝説のHコメディ「Oh! 透明人間」×「いけない! ルナ先生」コロバ読切で復活 <http://natalie.mu/comic/news/152961>
- [148] INSTALLATION NOTES for OpenBSD 5.7 <http://ftp.openbsd.org/pub/OpenBSD/5.7/luna88k/INSTALL.luna88k>
- [149] LUNAは、SX-9100以降の愛称なのですが、その前のSX-8700 の時代は、スーパーメイトという愛称でした。 <https://www.facebook.com/events/1062729970410808/permalink/1153405211343283/>
- [150] このマシンがスーパーメイトという商品名で <http://www.tomo.gr.jp/root/new/root82.html>
- 151 LUNA88Kに付属しているxzoomというデモの画像です。 <http://moon.hanya-n.org/comp/luna/luna88k.html>
- [152] OpenBSD/luna88k移植物語 http://www.slideshare.net/ao_kenji/a-story-of-porting-openbsd-luna88k
- 153 FM音源の調べ on LUNA http://www.slideshare.net/ao_kenji/nbug201510
- 154 PSG音源の調べ <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- 155 LUNAのPSGというか647180実装 <https://twitter.com/tsutsui/status/759793635898515456>
- [156] 「できません」と云うなーオムロン創業者 立石一真 <https://www.amazon.co.jp/dp/4478006334/>
- 157 <http://www.ustream.tv/recorded/90107872>
- 158 <https://speakerdeck.com/tsutsui/osc2016-hiroshima-psg-tunes-on-netbsd-luna68k-again>
- 159 <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- 160 <https://speakerdeck.com/tsutsui/osc2014-kansai-at-kyoto-netbsd-luna68k-report>
- 161 http://www.ceres.dti.ne.jp/tsutsui/kof2013/NetBSD_bootloader.html
- 162 http://www.ceres.dti.ne.jp/tsutsui/osc2013kyoto/NetBSD-luna68k.mlterm-fb_Twitter.html
- 163 <https://twitter.com/tsutsui/status/991191717050118144>
- [164] <https://speakerdeck.com/tsutsui/osc2016-kyoto-psg-tunes-on-netbsd-luna68k>
- [165] https://www.slideshare.net/ao_kenji/osc2019-nagoya
- 166 <http://www.pastel-flower.jp/~isaki/nono/>
- 167(1,2) <https://twitter.com/isaki68k/status/1261646479816404992>
- [168] <https://twitter.com/tsutsui/status/1262429647364427783>
- [169] <https://twitter.com/tsutsui/status/1262430960718508033>
- [170] <https://twitter.com/isaki68k/status/1262375954883772418>
- [171] <https://twitter.com/isaki68k/status/1262949576362930180>
- [172] <https://ja.wikipedia.org/wiki/%E7%AB%8B%E7%9F%B3%E7%BE%A9%E9%9B%84>
- 173 <https://twitter.com/isaki68k/status/1317441952107827201>
- [174] <http://wiki.netbsd.org/ports/luna68k/luna68k.info/#behindthescene>
- 175 <https://twitter.com/isaki68k/status/1322807313741148160>
- [176] https://twitter.com/ao_kenji/status/1324952816884985857
- [177] https://twitter.com/ao_kenji/status/1324990436390268928
- [178] https://twitter.com/ao_kenji/status/1330473862686003202
- [179] https://twitter.com/ao_kenji/status/1330504720516063235
- 180 <https://twitter.com/isaki68k/status/1330124516333412361>
- 180 https://twitter.com/ao_kenji/status/1330019763775365120
- 181(1,2) <https://twitter.com/isaki68k/status/1406522668321366022>
- [182] Here come LUNA-88K emulators! https://www.slideshare.net/ao_kenji/osc2021-nagoya-248742459
- [183] <http://mail-index.netbsd.org/source-changes/2022/06/10/msg139155.html>
- [184] <https://twitter.com/tsutsui/status/1551955355101691904>
- [185] <https://twitter.com/isaki68k/status/1551020192037097473>
- [186] <http://mail-index.netbsd.org/source-changes/2022/09/25/msg141239.html>
- [187] <https://twitter.com/isaki68k/status/1585612735932674049>

[188] <https://twitter.com/isaki68k/status/1618215457722617856>

[189] <https://twitter.com/isaki68k/status/1624651445684563968>

[190] https://twitter.com/ao_kenji/status/1625846099394330625

5.9.1. このページ

- <https://github.com/ebijun/NetBSD/blob/master/Guide/Luna.rst> にあります。
- `/usr/pkgsrc/textproc/py-sphinx` をインストールして、`make html`とか。

6. nono のバージョンアップ

`pkgsrc/emulators/nono` 以下を更新して、nonoのバージョンアップをしてみます。

6.1. バージョン指定

Makefile 内のバージョンをあげます。

```
DISTNAME=      nono-0.5.1
```

6.2. チェックサム作成

```
% make makesum
-rw-r--r--  1 jun  wheel  1182 Feb 14 06:42 Makefile
-rw-r--r--  1 jun  wheel   558 Feb 14 06:43 distinfo
```

`make package` して動作を確認します。

6.3. pkglint

`pkglint` をして正しいか確認します。

```
pkg_add pkglint
pkglint .
% pkglint .
Looks fine.
```

6.4. commit

バージョンアップ時のアップデート内容をしらべておきます。nonoの場合は、`changes.html`に日英併記された更新内容があるので、英語部分だけを変更点とします。更新ログの一行目は「パッケージ名: Update to バージョン」の形式にします。

<http://www.pastel-flower.jp/~isaki/nono/doc/changes.html>

```
cvs commit
```

```
nono: update to 0.5.1.
```

```
Changes in 0.5.1:
```

```
vm(Fix): "Fix a bus error in areas with no extended memory in X68030."
vm(Update): "Implement DAA instruction on HD647180."
debugger(Fix): "Fix breakpoints set while HD647180 is in reset state not working."
GUI(New): "Implement a device map monitor."
GUI(Update): "Link control size with font size, on DIPSW window."
GUI(Fix): "Support 24 dot fonts in a bitmap monitor window."
GUI(Fix): "Fix an abnormal termination when a bitmap monitor window is resized."
```

6.5. doc をcommit します

`doc` をcommit します。コメントは、「`doc: Updated カテゴリー/パッケージ名 to バージョン`」に設定します。

```
doc: Updated emulators/nono to 0.5.1
```

```
cvs diff -u CHANGES-2023
Updated emulators/nono to 0.5.1 [jun 2023-02-14]
```

7. sphinxのドキュメントをlatex経由でpdfに変換する

7.1. sphinxのインストール

```
# pkg_add py38-sphinx
# ln -s /usr/pkg/bin/sphinx-build-3.8 /usr/pkg/bin/sphinx-build
# which sphinx-build
/usr/pkg/bin/sphinx-build
```


conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

8.3. 2条項BSDライセンス(訳)

<http://www.jp.NetBSD.org/ja/about/redistribution.html>

ソースおよびバイナリー形式の再配布および使用を、変更の有無にかかわらず、以下の条件を満たす場合に認める:

1. ソースコードの再配布においては、上記の著作権表示、この条件の列挙、下記の注意書きを残すこと。
2. バイナリー形式の再配布においては、上記の著作権表示、この条件の列挙、下記の注意書きを、配布物に附属した文書および/または他のものに再現させること。

このライセンスの前には著作権表示そのものが付きます。この後には注意書きが付き、このソフトウェアに関して問題が生じて、作者は責任を負わないと述べます。

9. NetBSD

NetBSDは1個のソースツリーをコンパイルすることで実行イメージを作ることができます。

9.1. ソースコードから作る

tar 形式のファイルをダウンロード&展開し、build.shというスクリプトでコンパイルすると、NetBSDの実行イメージができます。

このtarファイルの中には、これまでNetBSDがサポートしてきた50種類以上のハードウェアと、無数の周辺機器の仕様が含まれています。しかもコンパイルすると、実際にハードウェア上でNetBSDが動作します。

NetBSDのコンパイルはNetBSDでも、NetBSDではないOSでも、どのハードウェアでも、ほぼ同じ手順でコンパイルできます。(できるはずですが)

```
# ftp ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/src.tar.gz
# tar xzvf src.tar.gz
# ./build.sh -U -m i386 release .... -U:root以外で作成,この場合i386向け
```

9.2. Xを含んだシステムを作る

```
# ftp ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/xsrc.tar.gz
# tar xzvf xsrc.tar.gz
# cd src
# ./build.sh -u -U -m i386 -x -X ../xsrc release ... -u:更新,-x Xも作る
```

9.3. CD-ROMイメージを作る

```
# ./build.sh -m i386 iso-image ... CD-ROMイメージ作成
```

10. pkgsrc - ソースコードからソフトウェアを作る

世界中にあるいろいろなプログラムをコンパイル・インストールする手順は、プログラムごとにまちまちです。世界中のプログラムを、すべて同じ手順でコンパイルして、インストールするためには、どのような仕組みがあればよいでしょうか？

ソースコードからプログラムをコンパイル・インストールする時、NetBSDでは主に、pkgsrcを利用します。pkgsrcでは、13000種類以上のプログラムについて、コンパイル手順を分野ごとにまとめて、収集しています。

pkgsrcの役割を挙げてみます。

1. 適切なサイトからソースコードをダウンロード展開する。
2. 適切なオプションをつけて、コンパイルする。
3. インストールする。
4. コンパイルした結果からパッケージを作る。
5. 他のマシンにパッケージをインストールする。

それではpkgsrcを実際に使ってみましょう。pkgsrc.tar.gzというファイルを展開して利用します。ここでは、すぐれたtwitterクライアントであるmikutterをインストールします。makeコマンドを実行すると、関連するソフトウェアをインストールします。

```
# cd /usr
# ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-current/tar_files/pkgsrc.tar.gz
# tar xzvf pkgsrc.tar.gz
(cd /usr/pkgsrc/bootstrap;./bootstrap) .. NetBSD以外のOSで実行する
# cd /usr/pkgsrc/net/mikutter
# make package-install
```

pkgsrc.tar.gz ファイルの中には、12000種類以上のソフトウェアをコンパイルし、インストールする方法が含まれています。しかもコンパイルすると、実際

10.1. gitをインストールしてみる

```
# cd /usr/pkgsrc/devel/git-base
# make install
```

```
# which git
/usr/pkg/bin/git
```

10.2. baserCMSをインストールしてみる

典型的なCMSは、この手順でインストールできます。

```
# cd /usr/pkgsrc/www/ap-php ... php54+apache
# make package-install ... 関連するソフトウェアが全部コンパイル・インストール
# vi /usr/pkg/etc/httpd/httpd.conf
LoadModule php5_module lib/httpd/mod_php5.so
AddHandler application/x-httpd-php .php

# cd /usr/pkgsrc/converters/php-mbstring
# make package-install

# vi /usr/pkg/etc/php.ini
extension=mbstring.so

baserCMSはMySQLをインストールしなくても利用できますが、利用する場合
# cd /usr/pkgsrc/databases/php-mysql ... php+mysqlインストール
# vi /usr/pkg/etc/php.ini
extension=mysql.so

# vi /usr/pkg/etc/httpd/httpd.conf
DirectoryIndex index.php index.html

# vi /etc/rc.conf
apache=YES
# cp /usr/pkg/share/examples/rc.d/apache/etc/rc.d/apache
# /etc/rc.d/apache start

basercms.netからzipファイルをダウンロード
# cd /usr/pkg/share/httpd/htdocs
# unzip basercms-2.1.2.zip
# chown -R www.www basercms
# http://localhost/basercms
管理者のアカウントとパスワードがメールで飛んでくる！！

pkgsrcを使う場合：
# cd /usr/pkgsrc/www/php-basercms
# make package-install
```

10.2.1. SSL設定

証明書のファイルを指定して、httpd.confのコメントを外して、apacheを再起動します。

```
/usr/pkg/etc/httpd/httpd-ssl.conf
SSLCertificateFile
SSLCertificateKeyFile
SSLCertificateChainFile

/usr/pkg/etc/httpd/httpd.conf
Include etc/httpd/httpd-ssl.conf ... コメントはずす
```

10.2.2. 日本語Wordpress

```
# cd /usr/pkgsrc/www/php-ja-wordpress
# make package-install
```

10.3. LibreOfficeを動かしてみる

LibreOfficeをインストールしてみましょう。

```
# cd /usr/pkgsrc/misc/libreoffice
# make package-install
: 9時間くらいかかります。
# which libreoffice
/usr/pkg/bin/libreoffice
```

10.4. IRCクライアントを動かしてみる

IRCクライアントとして hexchatをインストールします。

```
# cd /usr/pkgsrc/chat/hexchat
# make package-install
#hexchat
# サーバ選択画面で http://Liberia.Chat 指定します。
# チャンネルに#netbsd を指定します。
# 設定→設定→インタフェース→外観→一般→フォントでフォント選んで、右端の表示が切れるのを直します。
```

10.5. 依存しているパッケージを調べる

```
cd /usr/pkgsrc/pkgtools/revbump
make package-install
finddepends lang/rust .... rustに依存しているパッケージを調べる
```

10.6. インストールするソフトウェアのライセンスを意識する

あるソフトウェアのソースコードをどのように取り扱えばいいのかは、ソフトウェアに含まれるライセンスに書かれています。GNUやBSDやMITやApacheなど有名なライセンスもあれば、有名なライセンスを少しだけ入れ替えて、目的にあったライセンスに作り替えたものなど、まちまちです。pkgsrcでは、pkgsrcに含まれるソフトウェアのライセンスを収集しています。実際に見てみましょう。

```
% cd /usr/pkgsrc/licenses ... ライセンス条項が集まっている
% ls |wc -l
228
% ls |head
2-clause-bsd
3proxy-0.5-license
CVS
acm-license
adobe-acrobat-license
adobe-flashsupport-license
amap-license
amaya-license
amazon-software-license
amiwm-license
:
```

特定のライセンスを持つソフトウェアのインストールを許可するかどうかは、/etc/mk.conf ファイルで定義します。星の数ほどあるソフトウェアのライセンスを受け入れるかどうかを、自分で決めることができます。

```
% grep ACCEPTABLE /etc/mk.conf |head
ACCEPTABLE_LICENSES+= ruby-license
ACCEPTABLE_LICENSES+= xv-license
ACCEPTABLE_LICENSES+= mplayer-codec-license
ACCEPTABLE_LICENSES+= flash-license
ACCEPTABLE_LICENSES+= adobe-acrobat-license
ACCEPTABLE_LICENSES+= adobe-flashsupport-license
ACCEPTABLE_LICENSES+= skype-license
ACCEPTABLE_LICENSES+= lha-license
ACCEPTABLE_LICENSES+= opera-eula
ACCEPTABLE_LICENSES+= lame-license
```

10.7. pkgsrc/packages

コンパイルしたパッケージは、pkgsrc/packages以下に生成されます。

```
% cd /usr/pkgsrc/packages/All/
% ls *.tgz |head
GConf-2.32.4nb7.tgz
GConf-ui-2.32.4nb11.tgz
ORBit2-2.14.19nb4.tgz
SDL-1.2.15nb7.tgz
SDL_mixer-1.2.12nb5.tgz
acoread9-jpnfont-9.1.tgz
:
# pkg_add gedit-2.30.4nb17.tgz ... インストール
# pkg_info ... 一覧表示
# pkg_del gedit ... 削除
```

10.8. pkgsrcに何か追加したい

```
# cd /usr/pkgsrc/pkgtools/url2pkg
# make package-install
# cd /usr/pkgsrc/ジャンル/名前
# url2pkg ダウンロードURL
Makefileとかができる
```

10.9. /usr/pkgsrc以下のメンテナンス

```
# cd /usr/pkgsrc/pkgtools/lintpkgsrc
# make package-install
# cd /usr/pkgsrc; cvs update -PA ... /usr/pkgsrcを最新にする
# lintpkgsrc -pr .... 古くなったバイナリパッケージを消す
# lintpkgsrc -or .... 古くなったソースファイルを消す
# lintpkgsrc -mr .... ソースファイルのチェックサムが/usr/pkgsrcと合っているか
```

10.10. pkgsrcの更新

pkg_chkを使う方法

```
# cd /usr/pkgsrc/pkgtools/pkg_chk
# make package-install
# cd /usr/pkgsrc
# cvs update -PAd
# pkg_chk -u      .... 古いパッケージをコンパイルして更新する
```

pkg_rolling-replaceを使う方法:依存関係に従って更新する

```
# cd /usr/pkgsrc/pkgtools/pkg_rolling-replace
# make package-install
# cd /usr/pkgsrc
# cvs update -PAd
# pkg_rolling-replace -u
```

10.11. ソースコードの更新

```
http://cvsweb.NetBSD.org/
# cd src
# cvs update -PAd      ... 最新に更新
# cvs update -Pd -r netbsd-7 ... NetBSD7.0
# cd pkgsrc
# cvs update -PAd      ... 最新に更新
# cvs update -Pd -r pkgsrc-2015Q3 ... 2015Q3に更新
```

10.12. バグレポート・追加差分

<http://www.NetBSD.org> → Support → Report a bug / Query bug database.

10.13. The Attic Museum

https://wiki.netbsd.org/attic_museum

メンテナンスするのがつらくなってきた機能を削除します。yurexとか。

11. NetBSDとブース展示

日本NetBSDユーザーグループは、日本各地のオープンソースイベントに参加し、ブース出展とセミナー枠を利用して、NetBSD関連の情報をまとめています。オープンソースカンファレンスへの積極的な参加が認められ、2014年2月に「第1回OSCアワード」を受賞しています。

11.1. ブース出展

オープンソース関連のイベントでは、たいいてい幅1.8m程度の長机と椅子二つ程度のブースを出展します。各地域でのイベント開催に合わせて、最新の活動成果を展示しようとしています。

11.2. セミナー枠

セミナー枠では、NetBSDに関する情報を紙にまとめて配布して、出版物でカバーできないような情報をイベント毎にまとめています。開催地にある電子部品店・コンピュータショップ・古書店・クラフトビールバー等、生活に必要な情報もまとめています。

11.3. シール関連まとめ

NetBSDブースでは、NetBSDのシールや、NetBSDがサポートしている・サポートしようとしている・みんなが好きで利用しているソフトウェアに関連したシールを持ち寄って配っています。OSの展示は単調になりがちで、OS開発やNetBSDについて通りすがりの数秒で理解してもらうのは不可能でしたが、シールなら数秒で何かわかってもらえます。かさばらないので、誰にも受け取ってもらいやすく、優れたデザインのシールに人気が出ると、ブース全体に活気が生まれて、思いもよらない進展を呼ぶことがあります。

みくったーシールずかん	http://together.com/li/566230
らこらこシール作成の記録	http://together.com/li/554138

12. オープンソースカンファレンスNetBSDブースこの一年

日本NetBSDユーザーグループは2022年もオープンソースカンファレンスを中心とするイベントに参加しています。2020/1のOSC2020大阪以降はZOOMでの参加になりました。NetBSD/aarch64からZOOM会議に参加する試みをしました。従来、ブースへの展示機材持ち込み&差し入れに代わり、事前にtwitter上でデモ動画等を投稿されたものをtogetherでまとめておいて紹介するようになってみました。引き続きセミナー時間での参加・発表を歓迎します。

12.1. NetBSD環境からのZOOM会議参加

NetBSDからZOOM等の会議に参加するためには、以下の手順をとります。

1. rustが動くようにする。
2. audioが動くようにする。
3. 内蔵カメラが動くようにする。
4. Firefox80以降をpkgsrcからインストールする
5. FirefoxのプラグインでLinuxまたはFreeBSDからインストールしているように見せかける。

12.2. これまでに参加した一覧

これまでに参加した一覧は以下のとおりです。

<https://github.com/ebijun/NetBSD/blob/master/Guide/OSC/OSC100.csv>

このファイルはオープンソースカンファレンス過去来場者数一覧 <http://www.ospn.jp/visitors/> を元に作成しています。

12.3. どのくらい参加しているか

- 2023/1/28までにOSCは213回開催されています。
- JNUGは190回参加しています。90.0%→89.2% (前年比0.8%減)

12.4. OSCこの一年

12.4.1. 2022年まとめ

- 日本全国各地で12回オンライン開催+ODC開催
- 参加者: 103,646人 年間参加者3140人(2020年)→3115人(2021年)→2383人(2022年)
- 参加団体:6198グループ 年間参加グループ 152グループ(2020) → 133グループ(2021)→127グループ(2022)

12.5. togetterアクセスで見たNetBSDブース

togetterのアクセスログは以下の場所にあります。

<https://github.com/ebijun/NetBSD/blob/master/Event/togetter/togetterview/view.csv>

この一年のアクセス数集計は以下の通りです。

OSC2023 Online/Spring	https://togetter.com/li/2094211	143
OSC2023 Online/Osaka	https://togetter.com/li/2060234	1376
OSC2022 Online/Fukuoka	https://togetter.com/li/1977568	819
KOF2022	https://togetter.com/li/1970580	1460
OSC2022 Online/Fall	https://togetter.com/li/1964480	1466
OSC2022 Online Hiroshima	https://togetter.com/li/1950893	1083
ODC2022 Online	https://togetter.com/li/1938050	1403
OSC2022 Online/Kyoto	https://togetter.com/li/1921498	2045
OSC2022 Online/Hokkaido	https://togetter.com/li/1905889	1608
OSC2022 Online/Nagoya	https://togetter.com/li/1891762	1375
OSC2022 Online/Spring	https://togetter.com/li/1856809	1155

12.6. netbsd-advocacyメーリングリストへの報告

netbsd-advocacyメーリングリストへの参加報告をしました。

NetBSD machines at Open Source Conference 2020 Osaka <http://mail-index.netbsd.org/netbsd-advocacy/2020/01/28/msg000823.html>

12.7. NetBSD観光ガイド作成

イベント毎に観光ガイドを作っています。セミナー参加者に配布しました。

一覧:

<https://github.com/ebijun/osc-demo/blob/master/README.md>

作成方法:

<https://github.com/ebijun/NetBSD/blob/master/Guide/Paper/sphinx.rst>

185	OSC2023 東京春	http://www.re.soum.co.jp/~jun/OSC2023tokyospring.pdf
184	OSC2023大阪	http://www.re.soum.co.jp/~jun/OSC2023osaka.pdf
183	OSC2022福岡	http://www.re.soum.co.jp/~jun/OSC2022fukuoka.pdf
182	KOF2022	http://www.re.soum.co.jp/~jun/KOF2022.pdf
181	OSC2022 東京秋	http://www.re.soum.co.jp/~jun/OSC2022tokyofall.pdf
180	OSC2022広島	http://www.re.soum.co.jp/~jun/OSC2022hiroshima.pdf
179	ODC2022	http://www.re.soum.co.jp/~jun/ODC2022.pdf
178	OSC2022京都	http://www.re.soum.co.jp/~jun/OSC2022kyoto.pdf
177	OSC2022 北海道	http://www.re.soum.co.jp/~jun/OSC2022hokkaido.pdf
176	OSC2022 名古屋	http://www.re.soum.co.jp/~jun/OSC2022nagoya.pdf
175	OSC2022 東京春	http://www.re.soum.co.jp/~jun/OSC2022tokyospring.pdf
174	OSC2022大阪	http://www.re.soum.co.jp/~jun/OSC2022osaka.pdf

12.8. 旅費

横浜からの旅費(=交通費+宿泊費),機材配送費,資料印刷費実費をまとめています。

月	イベント	旅費	機材配送
2020/1	OSC大阪	26080	1469

12.9. 2023年

2023年もしばらくの間はオンライン開催が続きそうですが、現地グループが集まったハイブリッド開催もできるようになってきました。OSCは2023/1/28のOSC Online大阪(<https://event.ospn.jp/osc2023-online-osaka/>)からはじまります。セミナー/ミーティング時間での発表を歓迎します。

13. NetBSD 2022 Annual General Meeting on 2022/05/21

<http://mail-index.netbsd.org/netbsd-announce/2022/05/11/msg000338.html>

America/Los_Angeles	Sat May 21 07:00:00 PDT 2022
America/Denver	Sat May 21 08:00:00 MDT 2022
America/Winnipeg	Sat May 21 09:00:00 CDT 2022
America/New_York	Sat May 21 10:00:00 EDT 2022
Europe/London	Sat May 21 15:00:00 BST 2022
Europe/Berlin	Sat May 21 16:00:00 CEST 2022
Europe/Sofia	Sat May 21 17:00:00 EEST 2022
Europe/Moscow	Sat May 21 17:00:00 MSK 2022
Asia/Calcutta	Sat May 21 19:30:00 IST 2022
Asia/Bangkok	Sat May 21 21:00:00 +07 2022
Asia/Manila	Sat May 21 22:00:00 PST 2022
Asia/Tokyo	Sat May 21 23:00:00 JST 2022
Australia/Adelaide	Sat May 21 23:30:00 ACST 2022
Australia/Melbourne	Sun May 22 00:00:00 AEST 2022
Pacific/Auckland	Sun May 22 02:00:00 NZST 2022

```
<pickles> good morning! (if applicable, else hello! :) )
<racoon> morning :)
<spz> Dear Gentlebeings,
<spz> ,
<spz> Welcome to the 2022 Annual General Meeting
<spz> of The NetBSD Foundation.
<spz> ,
<spz> I am the moderator, i.e. voice-bot, for todays AGM.
<spz> ,
<spz> This is the agenda for this AGM:
<spz> ,
<spz> Intro
<spz> ,
<spz> Reports from:
<spz> * board
<spz> * communications
<spz> * finance-exec
<spz> * membership-exec
<spz> * admins
<spz> * core
<spz> * security-team
<spz> * releng
<spz> * pkgsrc-pmc
<spz> * pkgsrc-security
<spz> * pkgbuild
<spz> * gnats
<spz> ,
<spz> Are there any last-minute additions to the reports? msg me.
<spz> ,
<spz> Q&A session for all of the above, and general questions.
<spz> ,
<spz> When Q&A begins, msg me "I have a question for <team>"
<spz> or "I have a question for <nick>" and I will give you voice
<spz> when it's your turn.
<spz> ,
```

13.1. Introduction

```
<spz> Here comes leot with the intro,
<spz> followed by the reports by board and communication:
<leot> Hello everyone and welcome! (this report was prepared by Cryo)
<leot> -
<leot> Hello, and welcome to the 20th Annual General Meeting of The
<leot> NetBSD Foundation.
<leot> -
<leot> First off, I'd like to thank spz (AGAIN) for volunteering to
<leot> handle moderating and admins for doing the behind the scenes
<leot> magic to make this event (and all our communication) possible.
<leot> -
<leot> Second, let me take a moment of your time to say that
<leot> we have all had a tough couple of years enduring a pandemic
<leot> that has cost some of us quite a bit. No one expected that
<leot> it would last this long, or that the geopolitical and social
<leot> climates would have resulted in where we are now. I hope that
<leot> you are well and safe during these unstable times we live in now.
<leot> NetBSD and Pkgsrc continue to make improvements during this,
<leot> albeit a little slower. We are on the cusp of branching NetBSD-10
```

```
<leot> after 3 years of development with a slew of new features,
<leot> improvements, and squished bugs. Your commits are very much
<leot> appreciated, as is your continued support of the foundation.
<leot> -
```

13.2. The report from the Board of Directors:

```
<leot> Now on to the report from the Board of Directors:
<leot> -
<leot> The NetBSD Foundation Board of Directors presents a consolidated
<leot> list of the relevant and major actions that occurred since last
<leot> AGM. Quite a few discussions, actions, and follow-ups crossed
<leot> multiple meetings. Very few meetings resulted in not reaching
<leot> quorum. Check our weekly meeting minutes in:
<leot> localsrc/tnf/board/minutes for the latest on our progress.
<leot> During this period, new directors were elected by the members
<leot> and officers were renewed or installed.
<leot> -
<leot> We continued our support of BSDcan, AsiaBSDcon, and
<leot> EuroBSDcon to improve our representation at conferences and
<leot> developer summits. Due to the never-ending pandemic, the
<leot> continued cancellations and move to virtual summits, any approved
<leot> contributions will be continue to be shifted until we can once
<leot> again meet somewhere in the future.
<leot> -
<leot> We participated in the Google Summer of Code for 2021 but there
<leot> was no virtual Google Summer of Code Mentor Summit. We are
<leot> currently participating in GSoC this year with 5 students!
<leot> -
<leot> Like last year, we have provided core with a pre-approved,
<leot> reasonable budget, to spend as they see most fit without an
<leot> additional confirmation step from us.
<leot> -
<leot> We continued to improve our interaction and relationships with
<leot> vendors, as well as participating in industry PSIRT/CSIRT
<leot> with commercial vendors and other open-source projects.
<leot> -
<leot> The funded contracts continued for:
<leot> o - improvements in release engineering by martin
<leot> o - improvements in WiFi by martin
<leot> -
<leot> We completed a fundraising campaign, started a new one, and
<leot> many many other minor things.
<leot> -
<leot> We would like to thank the vote coordinator, vote validator, and
<leot> nomcom for the handling of last year's slate. The new voting
<leot> system was approved by the members, and we are starting the
<leot> use of it for board seat openings very very shortly.
<leot> -
<leot> It has been an honor and pleasure to continue working with
<leot> leot, mef, m1elstv, nia, riastradh, and wiz to accomplish all
<leot> that we have in this year.
<leot> -
<leot> Respectfully submitted on behalf of the Board of Directors
<leot> .eof
<leot> Here the report from communication prepared by <billc> and <nia>, thanks!:
<leot> -
<leot> A non-scientific representation of Social Media Presence:
<leot> -
<leot> Twitter: (very active)
<leot> @netbsd has 10,000 followers up from 9,490
<leot> @pkgsrc has 685 followers up from 639
<leot> -
<leot> We have created an account on the distributed social network
<leot> ActivityPub ('the fediverse'), where we have a small but
<leot> dedicated fan base.
<leot> @netbsd@mastodon.sdf.org has 230 followers (very active)
<leot> -
<leot> Facebook: 2,000 up from 1,900 members (sort of active)
<leot> -
<leot> Since we moved from FreeNode, to Libera.chat our numbers have
<leot> stabilized. To help improve connectivity, we have created a
<leot> Matrix bridge to our IRC channel.
<leot> Libera IRC users: (very very active)
<leot> #NetBSD: 228 down from 248
<leot> #NetBSD-code: 59 up from 56
<leot> #pkgsrc: 102 up from 92
<leot> -
<leot> .eof
<spz> thank you leot
```

13.3. For finance-exec by Riastradh

```
<spz> next presentation is for finance-exec by Riastradh
<Riastradh> Hi! I'm Riastradh on finance-exec. We maintain The NetBSD
<Riastradh> Foundation's financial records and assets at board's direction.
```



```

<Riastradh> We make sure the books are balanced and and send thank-you
<Riastradh> letters to donors so they can get tax deductions (in the US).
<Riastradh> We are:
<Riastradh> - riastradh
<Riastradh> - christos
<Riastradh> .
<Riastradh> We keep the books with ledger-cli <https://www.ledger-cli.org/>,
<Riastradh> which we've been doing for a few years now, and use it to
<Riastradh> prepare the public financial report and internal reports with
<Riastradh> more detail. This way we have an audit trail, under source
<Riastradh> control, for all changes to the log of all transactions by TNF,
<Riastradh> which we reconcile with statements from financial institutions
<Riastradh> and payment processors.
<Riastradh> .
<Riastradh> The NetBSD Foundation's public 2021 financial report is at:
<Riastradh> https://www.NetBSD.org/foundation/reports/financial/2021.html
<Riastradh> .
<Riastradh> Highlights:
<Riastradh> - We have net assets of a bit over 200k USD.
<Riastradh> - We took in >50k USD last year -- hooray!
<Riastradh> - Not only that, but our expenses were unusually low, <20k USD.
<Riastradh> => It's good that we did better than break even!
<Riastradh> => But it also means we're not doing enough with the money.
<Riastradh> => (How to spend money is up to board and core.)
<Riastradh> - (That said, we are still bad at fundraising.)
<Riastradh> .
<Riastradh> This year, in view of the high level of inflation since the
<Riastradh> pandemic, the board authorized finance-exec to invest funds in
<Riastradh> broad-market low-cost passively managed index funds in stocks and
<Riastradh> bonds instead of just cash accounts -- which we maybe should have
<Riastradh> done years ago since our net assets are substantially higher than
<Riastradh> our annual revenue and expenses. But finance-exec prudently
<Riastradh> waited for the recent market crash so we haven't lost anything!
<Riastradh> (Except the value of the USD due to inflation.)
<Riastradh> .
<Riastradh> Happy to answer any questions about what finance-exec does!
<Riastradh> Thanks, -finance-exec
<spz> Thanks Riastradh

```

13.4. Membership-exec by wiz

```

<spz> next we're getting the report from membership-exec by wiz
<_wiz_> thanks, spz
<_wiz_> The current members of membership-exec are:
<_wiz_> - Christos Zoulas <christos>
<_wiz_> - Martin Husemann <martin>
<_wiz_> - Lex Wennmacher <wennmach>
<_wiz_> - Thomas Klausner <wiz>, and
<_wiz_> - Ken Hornstein <kenh> who is on sabbatical.
<_wiz_> -
<_wiz_> Membership-exec is responsible for all aspects of
<_wiz_> "membership", but in practice the main task is to handle
<_wiz_> membership applications. The number of active developers
<_wiz_> (as of 2022-05-21) is 144 (down from 152 last year). As
<_wiz_> in the last years, Thomas Klausner <wiz> conducted an
<_wiz_> account cleanup.
<_wiz_> -
<_wiz_> Since the last AGM we gained 8 new developers and,
<_wiz_> mostly due to the account cleanup, lost 16.
<_wiz_> -
<_wiz_> The difference between developers and active developers
<_wiz_> is explained in the bylaws - an active developer has
<_wiz_> actually committed something in the last year, or contributed
<_wiz_> in an active way, like admins.
<_wiz_> -
<_wiz_> We'd like to emphasize that we appreciate all your replies
<_wiz_> to our membership RFC e-mails, although we do not usually
<_wiz_> acknowledge them. Please keep on providing feedback to
<_wiz_> the RFC mails.
<_wiz_> that's it from membership-exec.
<spz> Donning my admins' hat:
<spz> good localtime() all
<spz> ,
<spz> admins is the following people:
<spz> christos, dogcow, gendalia, mspo, phil, riastradh, riz, seb, soda, spz, tls
<spz> ,
<spz> Statistics:
<spz> - admins runs the following TNF systems:
<spz> @ TastyLime
<spz> + 8 hardware systems and 6 Xen guests
<spz> = 1 earmv7hf, the rest amd64
<spz> @ Columbia University
<spz> + 11 hardware systems and 4 Xen guests,
<spz> = 2 i386 guests, the rest amd64
<spz> @ Washington University
<spz> + 7 hardware systems and 2 Xen guests
<spz> = 1 earmv7hf and the rest amd64
<spz> @ Regensburg (commercial housing)

```

```

<spz> + 2 hardware systems, one of them with 2 Xen guests,
<spz> = all amd64
<spz> ,
<spz> - CDN services donated by Fastly
<spz> ,
<spz> NetBSD versions in use:
<spz> 3 pre-8.1
<spz> 3 8.1_STABLE
<spz> 4 9.0_STABLE
<spz> 7 9.1_STABLE
<spz> 12 9.2_STABLE
<spz> 2 currentish
<spz> ,
<spz> Changes in use of system:
<spz> the two armbulk builders have been retired: the aarch64 system @WWU builds
<spz> their packages faster and with less fuss.
<spz> ,
<spz> Notable plans are to get all 8.x systems updated.
<spz> ,
<spz> Thanks to riz, tls and phil for their resources, time
<spz> and blood sacrifices, too. :)
<spz> ,
<spz> Back to moderator.
<spz> ,

```

13.5. core

```

<spz> The next three reports will be given by __martin, for core, security-team and re
<spz> leng:
<__martin> hello everyone - here is the core report:
<__martin>
<__martin> Core's job is to provide technical leadership for the
<__martin> project, manage funded projects, create roadmaps for
<__martin> the future of the project, and resolve disputes.
<__martin> Although we mean well, we have been doing less than
<__martin> what we've hoped to achieve, and we welcome help.
<__martin>
<__martin> Our group is:
<__martin>   Alistair Crooks <agc@NetBSD.org>
<__martin>   Christos Zoulas <christos@NetBSD.org>
<__martin>   Chuck Silvers <chs@NetBSD.org>
<__martin>   Robert Elz <kre@NetBSD.org>
<__martin>   Martin Husemann <martin@NetBSD.org>
<__martin>   Matthew Green <mrg@NetBSD.org>
<__martin>   Taylor R Campbell <riastradh@NetBSD.org>
<__martin>
<__martin> The group can be reached at <core@NetBSD.org>
<__martin>
<__martin> One of the tasks of the core team is to oversee,
<__martin> initiate and approve funded projects. We had a few of
<__martin> those in the last year and a few ongoing and planned
<__martin> for the future.
<__martin>
<__martin> - Martin working on merging wifi from FreeBSD, with
<__martin> nat@ and a student of Phil (James Browning) helping
<__martin> a lot.
<__martin> - releng work (martin)
<__martin> - last year we only received one GSoC slot, but had another
<__martin> worthwhile proposal, funded a second student implementing
<__martin> a posix_spawn(2) extension to support chdir(2). The
<__martin> official GSoC project failed, but the TNF funded student
<__martin> succeeded and the code has been merged. The same student
<__martin> is doing another project in this years GSoC.
<__martin> - this year we got five projects for GSoC (more details
<__martin> to be annouced shortly).
<__martin>   One of them is trying to get the Raspberry PI GPU
<__martin> supported (via a kernel module),
<__martin>   one is about automation of some tedious and time-
<__martin> consuming semi-manual jobs that finance-exec is doing
<__martin> now when we receive donations.
<__martin>   Another one is about adding support for two more wifi
<__martin> chipsets on top of the wifi renovation work.
<__martin>   The other two are enhancements to linux emulation
<__martin> and improvements for inetd.
<__martin>
<__martin> We are always open for suggestion from developers who would
<__martin> like to work on specific topics that require larger
<__martin> work or have been long neglected and need cleanup.
<__martin>
<__martin> Another task of the core team is to resolve disputes
<__martin> between developers and provide ruling for contested
<__martin> issues. This year was quiet in this regard, besides
<__martin> one open item where we have to decide how to deal with
<__martin> userland requests for strong entropy. A public
<__martin> statement about our ruling on this is being worked on.
<__martin>
<__martin> This needs to happen before the NetBSD 10 release.
<__martin> Next thing after that we plan to have the wifi work

```

```

<_martin> merged into -current (about one month after the
<_martin> netbsd-10 branch started), which initially might break
<_martin> a few chipsets. Hopefully this will be fixable
<_martin> quickly. Help from everyone with access to (supported)
<_martin> wifi hardware will be appreciated!

```

13.6. Security team

```

<_martin>
<_martin> The most painful task for core is to automatically be
<_martin> part of the security team and acting as fallback to make
<_martin> sure that security issues are handled in a timely fashion.
<_martin> Please consider joining our security team to help.
<_martin>
<_martin> Another way you can help the core team is to take over
<_martin> maintenance of some of our big 3rd party software.
<_martin> Currently most of this is done by both christos@ and mrg@,
<_martin> who could use more time for core and other business.
<_martin> If you feel you could help, please contact the current
<_martin> maintainer (see src/doc/3RDPARTY).
<_martin> thats it for core, happy to take questions later
<_martin> now for security team, this report was prepared mostly by shm:
<_martin> This is a brief report for security-team.
<_martin> -
<_martin> Since the last AGM, we have released two
<_martin> Security Advisories:
<_martin>
<_martin> * NetBSD-SA2022-001 PPPoE discovery phase memory corruption
<_martin> * NetBSD-SA2021-002 Incorrect permissions in kernfs
<_martin>
<_martin> Another SA for the recent mail.local(8) fixes is about
<_martin> to be published.
<_martin> -
<_martin> We'd like to encourage the NetBSD project members
<_martin> participate in the security-team work. The process of
<_martin> writing advisories and handling the patches is
<_martin> time-consuming but rewarding: one gets to learn about
<_martin> different ways used to subvert security which in turn
<_martin> improves their ability to write secure code, implement
<_martin> security fixes, and at the same time learn how to
<_martin> convey complex technical concepts in clear and concise
<_martin> manners. Because we are short-handed, we are slow
<_martin> analyzing issues, responding to known ones, or we are
<_martin> delayed on publishing already corrected security
<_martin> vulnerabilities. To be able to be more responsive, we
<_martin> need more hands on board! Feel free to contact us at
<_martin> security-team@.
<_martin> -
<_martin> Additionally we invite you to share with the team all
<_martin> your concerns about security of the NetBSD code,
<_martin> especially if you feel that the bug you just fixed
<_martin> could be connected with security. If you have any
<_martin> questions or ideas how to improve, we are here to
<_martin> help.
<_martin> -
<_martin> NetBSD continues to be represented in a product
<_martin> security incident response working group with other
<_martin> operating system vendors (CERT/VINCE/FULL DISCLOSURE)
<_martin> as well as software package distributors
<_martin> (ISC/OpenSSL/NTP), and we are in direct contact with
<_martin> the security teams of the other BSD projects.
<_martin> This framework allows us to better work with vendors
<_martin> requiring an embargoed and/or coordinated release with
<_martin> other operating systems. We can begin working on
<_martin> issues that affect NetBSD much faster, instead of
<_martin> only notified after an embargo is lifted.
<_martin> -
<_martin> Submitted respectfully on behalf of the
<_martin> security-officer(s), the security-team, and the sirt team.
<_martin> that's it for security-team, moving on the releng-hat

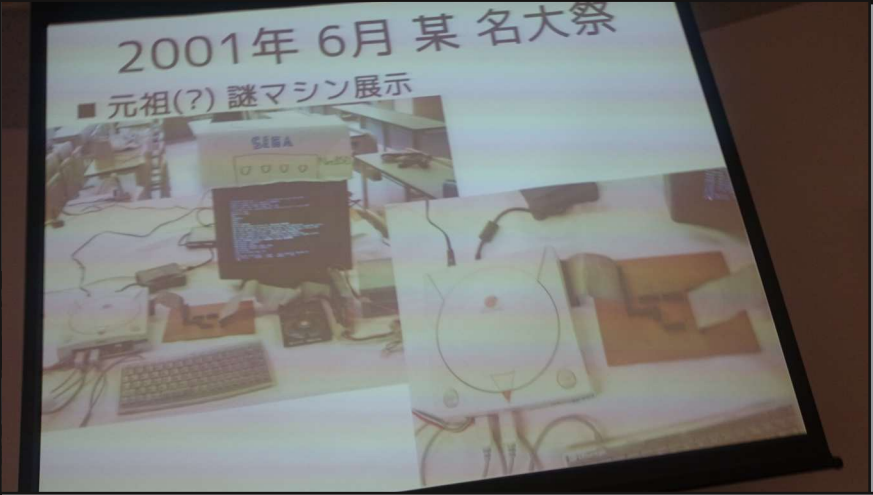
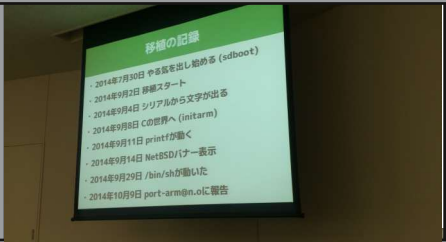
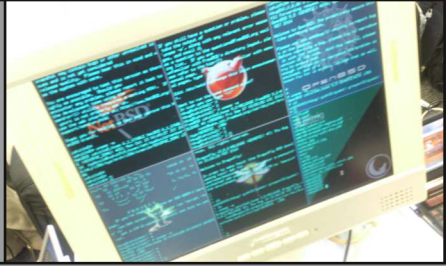
```

13.7. releng

```

<_martin> We are:
<_martin> abs agc bouyer he jdc martin msaitoh phil reed riz
<_martin> sborrill snj
<_martin>
<_martin> Since the last meeting, we have:
<_martin>
<_martin> - Released nothing (doh!) - last release (9.2) predates the last
<_martin> meeting by a few days
<_martin> - Processed hundreds of pullup requests.
<_martin> - Started initial preparations to make the build cluster
<_martin> independent from cvs (and also encouraged some developers
<_martin> to experiment with off-the-shelf alternatives)
<_martin>

```



Issue: 185
2023/03/11

contact: jun@soum.co.jp twitter: @ebijun
backnumber: github.com/ebijun/osc-demo/