

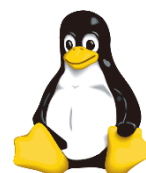
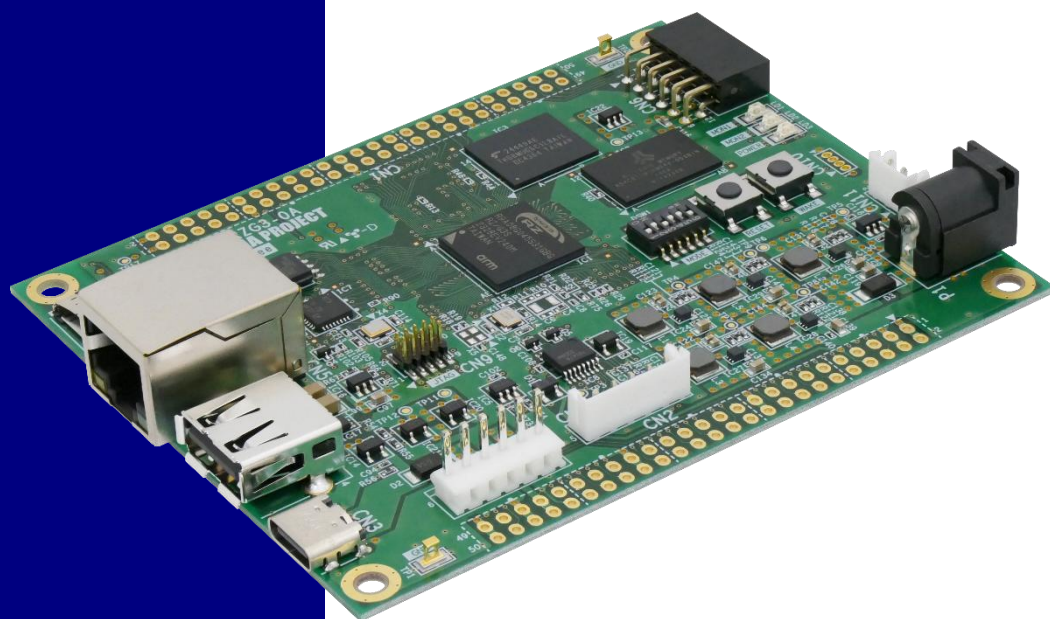
Alpha Board Series
LK-RZG3-A01

RZファミリ RZ/G3S CPUボード

Development Environment Install Guide

Rev 1.0

ダイジェスト版



ALPHA PROJECT
株式会社アルファプロジェクト

目次

1. 概要	1
1.1 はじめに	1
1.2 Linuxについて	1
1.3 VirtualBoxについて	1
1.4 Ubuntuについて	1
1.5 GNUとFSFについて	2
1.6 GPLとLGPLについて	2
2. 開発環境	3
2.1 ハードウェアについて	3
2.2 ソフトウェアについて	3
3. VirtualBox	4
3.1 VirtualBoxの構成	4
3.2 インストール手順	5
3.3 仮想マシンの作成手順	9
4. Ubuntu	12
4.1 インストール手順	12
4.2 端末アプリケーションの起動	23
4.3 ネットワーク設定	25
4.4 NFS	29
4.5 TFTP	31
4.6 FTP	33
5. クロス開発環境用パッケージ	35
5.1 クロス開発環境について	35
5.2 パッケージのインストール手順	36
6. 製品サポートのご案内	37
7. エンジニアリングサービスのご案内	38

2. 開発環境

VirtualBoxをインストールするPCが必要になります。
以下に、必要となるPCの環境を記載しますので、ご用意ください。

2.1 ハードウェアについて

本開発環境を利用するには、以下のハードウェア環境が必要になります。
なお、CPU性能、HDD空き容量、メモリ容量は、推奨環境となります。

種類	環境
CPU	DualCore 2GHz以上
HDD空き容量	150GByte以上 (Windowsが使用する容量は除きます。)
メモリ容量	8GByte以上(推奨16GByte以上)
LANポート	1ポート (外部のネットワークに接続できる環境が必要です。)
CDドライブ	開発キット付属のメディアが読み込み可能なドライブ (開発キットに付属するメディアによって必要なドライブが変わります。)

Table 2.1-1 ハードウェア環境

2.2 ソフトウェアについて

本開発環境を利用するには、以下のソフトウェア環境が必要になります。

種類	環境
ホストOS	Windows10/11 (64bit版)
ゲストOS	Ubuntu 22.04LTS 64bit版
VirtualBox	7.1.10 以降

Table 2.2-1 ソフトウェア環境



32bit版ではビルドできません。
VirtualBoxのゲストOSに64bit版のOSを使用する場合には、CPUに以下のサポートが必要になります。

- ・インテル製CPUの場合は、仮想化支援機能「VT-x」のサポートが必要です。
- ・AMD製CPUの場合は、ロングモードでのセグメントリミットのサポートが必要です。

上記対応にはBIOSの設定が必要な場合があります。なお、BIOSの設定に関しては、お使いのPCの説明書等でご確認ください。

3. VirtualBox

VirtualBoxは仮想マシンを実行するソフトウェアです。

VirtualBoxを利用することにより、Windows上でLinuxの開発を行うことができます。

本製品では、VirtualBoxのゲストOSにUbuntuを利用しています。

3.1 VirtualBoxの構成

VirtualBoxはWindows上でLinux環境を構築した仮想マシンを実行できるソフトウェアです。

CPUを全てエミュレートするのではなく、カーネルモード命令のみをエミュレートすることで、コマンド変換におけるオーバーヘッドを少なくし、実ハードウェアに近い性能を実現しています。

また、ネットワーク機能も実ハードウェアと同等程度の速度で動作します。

本ドキュメントでは、VirtualBoxが動作するWindowsOSをホストOS、VirtualBox上で動作するLinuxOS (Ubuntu) をゲストOSと表現します。

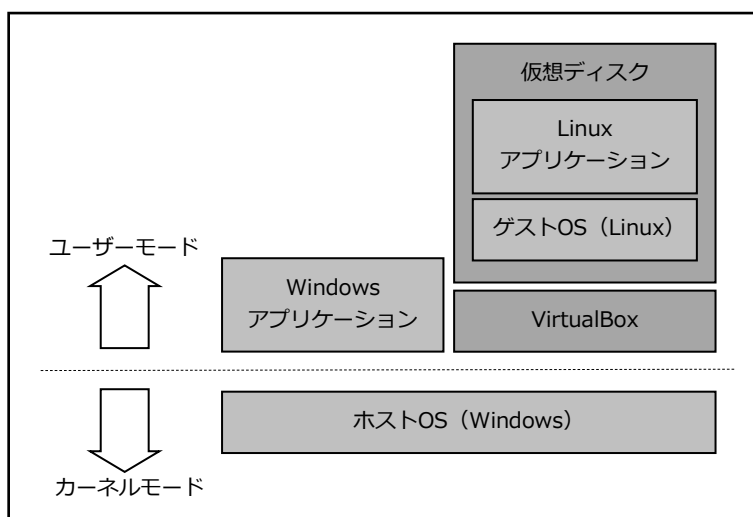


Fig 3.1-1 VirtualBox構成例

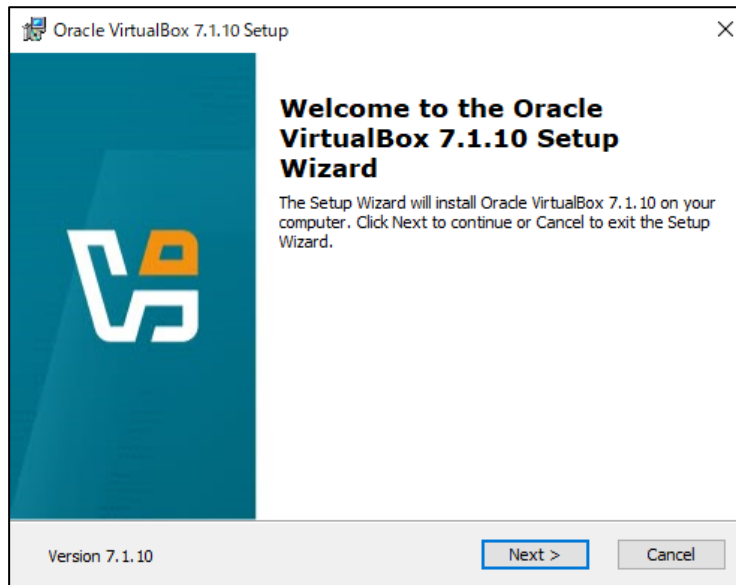
3.2 インストール手順

VirtualBoxのインストール手順を説明します。

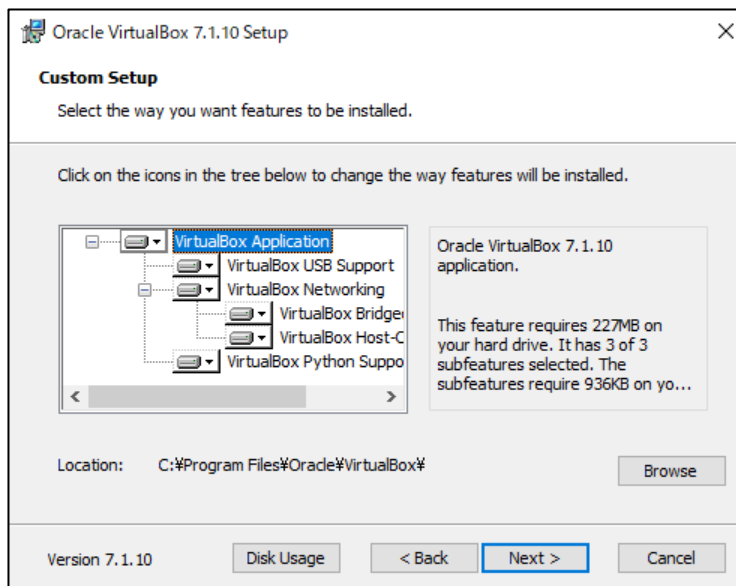
VirtualBoxのインストーラは本開発環境には付属していないため、VirtualBoxのダウンロードサイトから取得する必要があります。
なお、本ドキュメントで説明するバージョンに関しては、『[2.2 ソフトウェアについて](#)』でご確認ください。

① VirtualBoxのインストーラを実行します。

インストーラ開始画面が表示されますので、『**Next >**』ボタンをクリックして開始します。



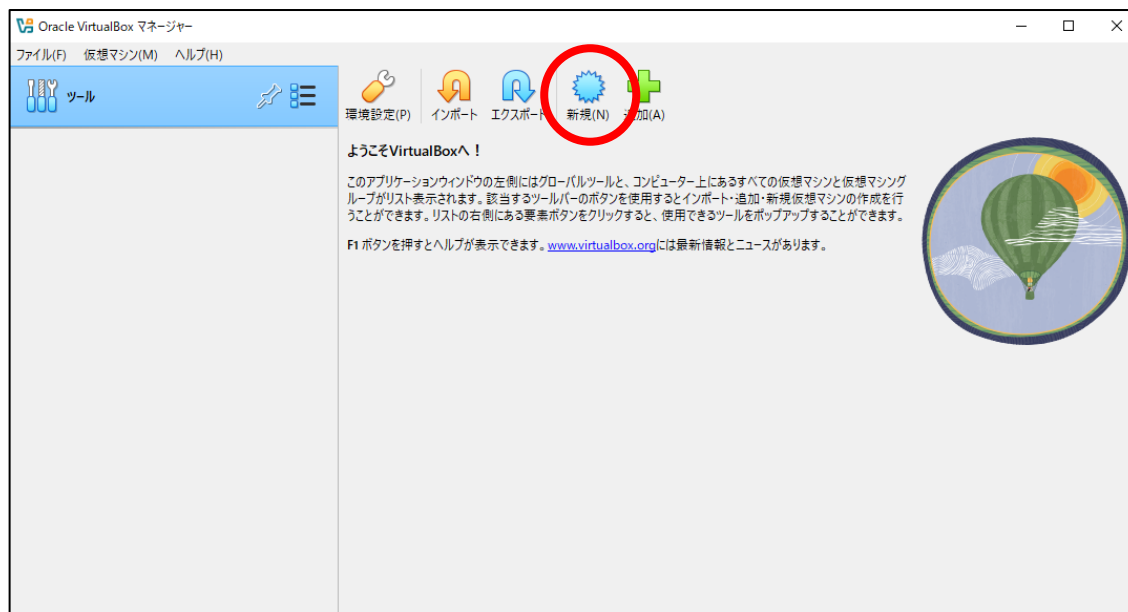
② インストール先フォルダ等を設定し、『**Next >**』ボタンをクリックします。



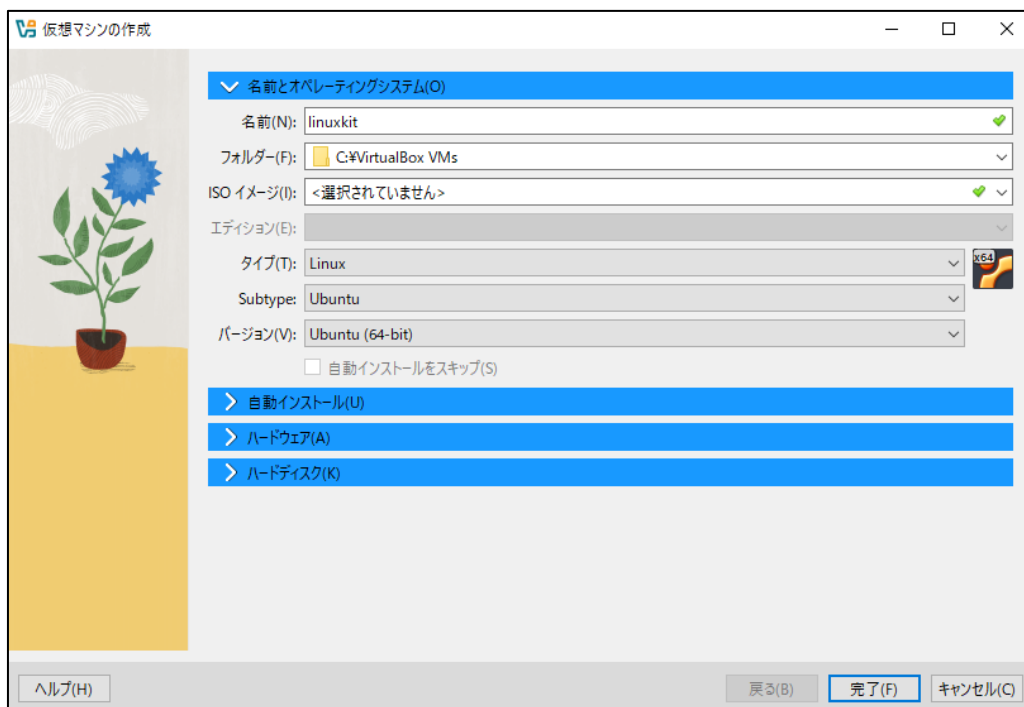
3.3 仮想マシンの作成手順

Ubuntuを動作させるため、仮想マシンを作成する必要があります。
作成するには、以下の手順で行います。

- ① VirtualBoxを起動して、『新規(N)』ボタンをクリックします。



- ② 名前、タイプ等を入力します。
名前、フォルダーは任意ですが、タイプは『Linux』、Subtypeは『Ubuntu』、バージョンは『Ubuntu (64-bit)』を設定してください。
ISO イメージは、次のUbuntuインストール手順にて設定しますので、ここでは選択しません。



4. Ubuntu

本章では、VirtualBoxで作成した仮想ディスクにUbuntuをインストールする手順を説明します。

4.1 インストール手順

VirtualBoxで作成した仮想ディスクにUbuntuをインストールする手順を説明します。

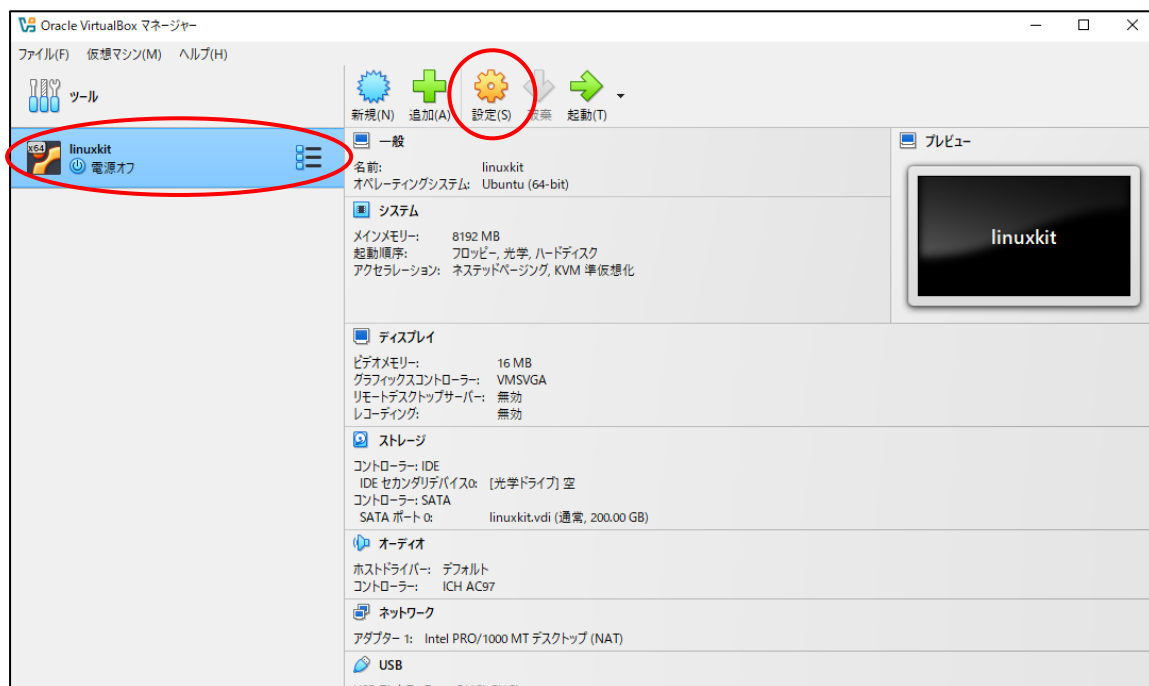
インストールにはUbuntuのISOイメージファイルが必要ですが、本開発環境には付属していないため、Ubuntuのダウンロードサイトから取得する必要があります。

<https://ubuntu.com/download/alternative-downloads>

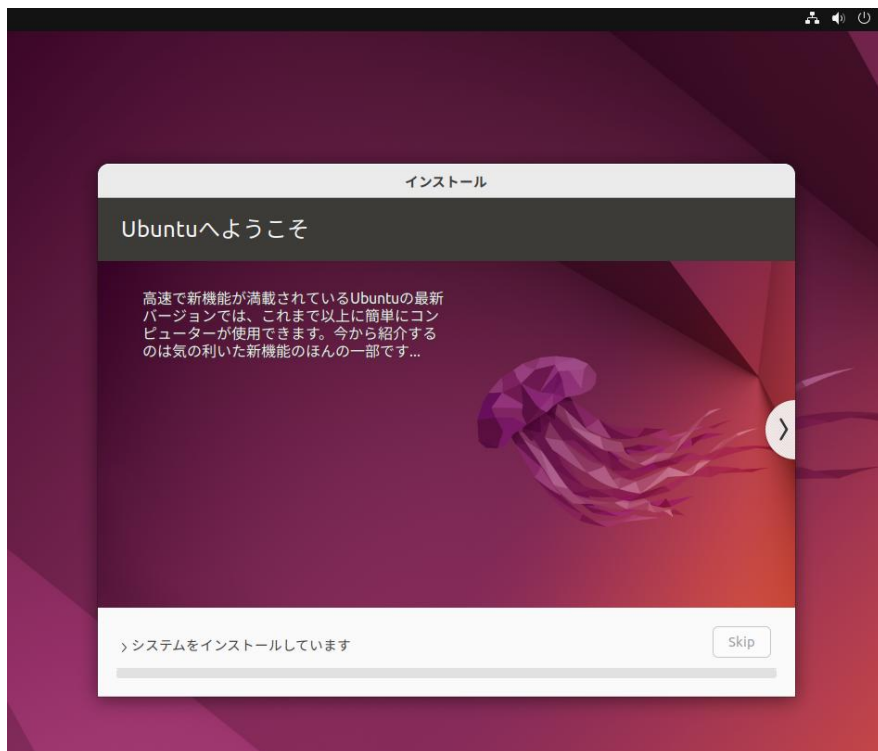
なお、本ドキュメントで説明するバージョンに関しては、『[2.2 ソフトウェアについて](#)』でご確認ください。

仮想マシン環境の設定

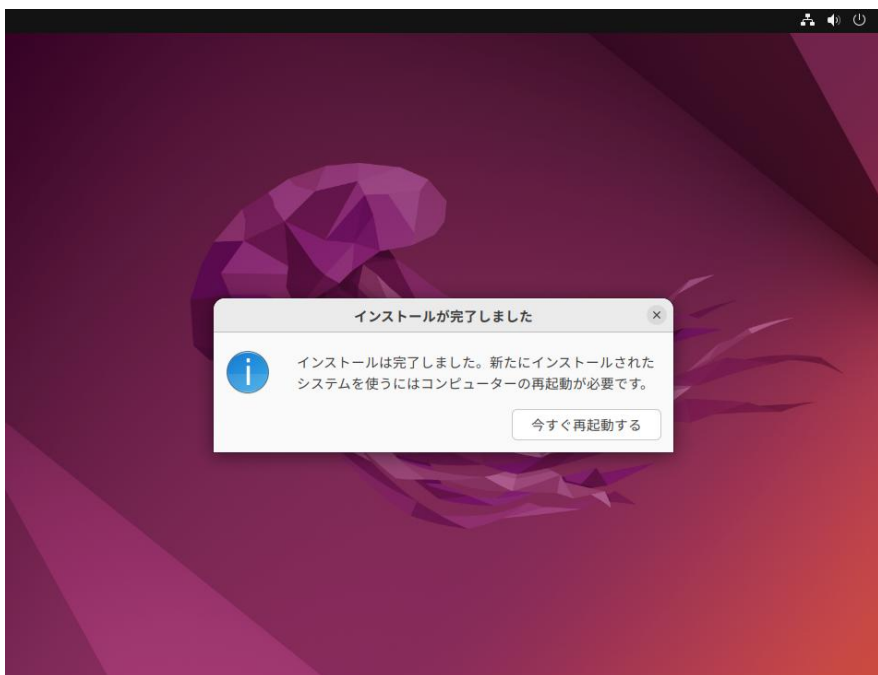
- ① VirtualBoxを起動し、左側から変更したい仮想マシンを選択し、『設定(S)』ボタンをクリックします。



- ⑨ インストール中です。完了するまでお待ちください。



- ⑩ 正常にインストールが完了しますと、メッセージが表示されます。
『今すぐ再起動する』ボタンをクリックして、再起動します。

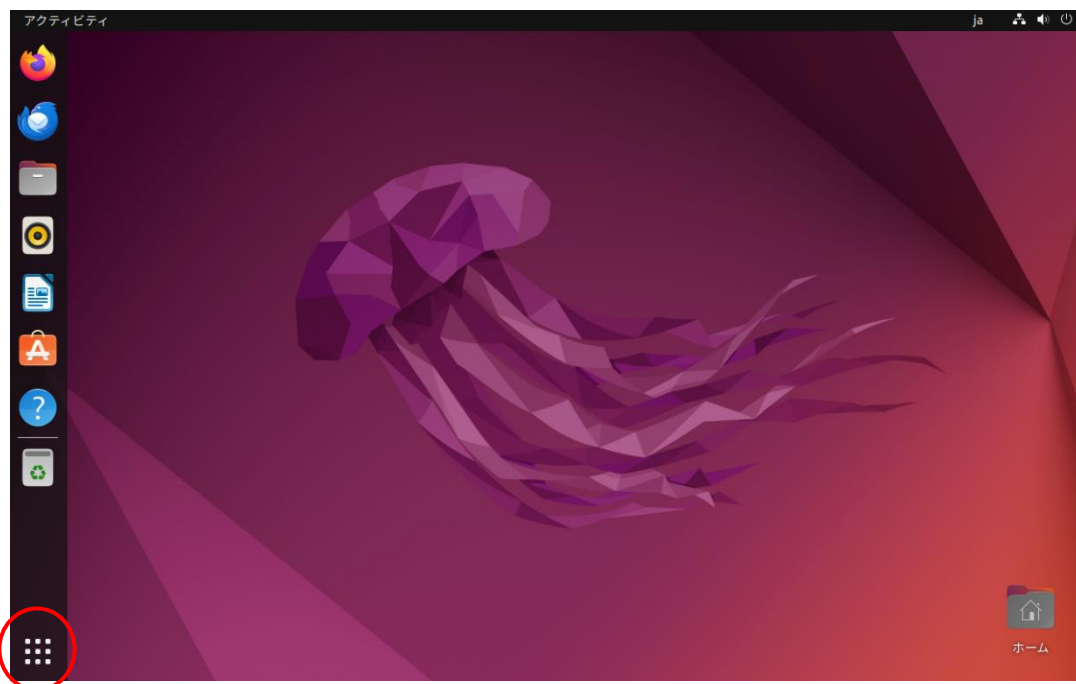


4.2 端末アプリケーションの起動

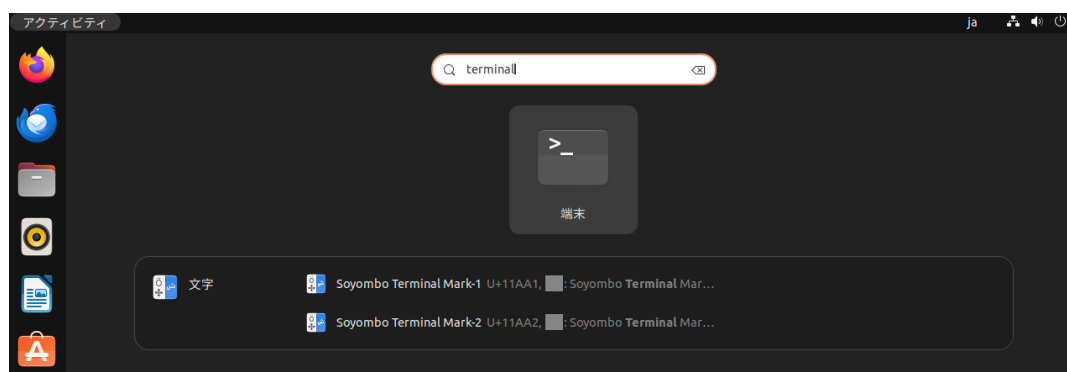
ソフトウェアの開発には、コマンド入力が必要となります。

本節では、コマンド入力可能な端末アプリケーションの起動方法を説明します。

- ① 左下のアプリケーションを表示するアイコンをクリックします。



- ② アクティビティの検索に『terminal』または『端末』と入力します。
入力中にも該当するアプリがリアルタイムに表示されます。



4.3 ネットワーク設定


DHCPを使用する場合はデフォルトのままで使用できます。固定IPを使用する場合は以下の手順に従って設定してください。

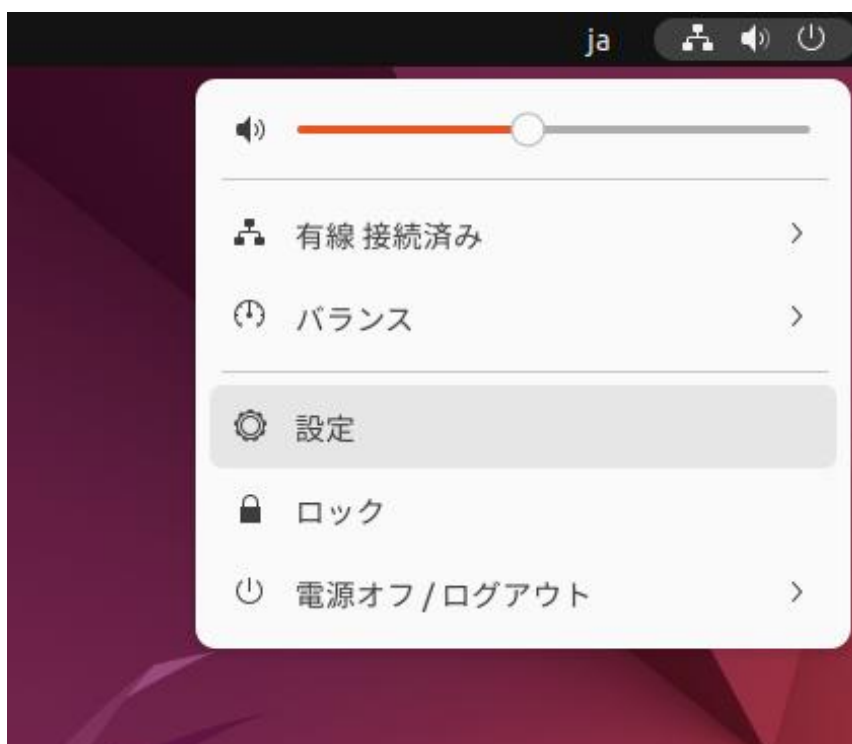
ネットワーク設定の変更

ここではネットワークとの接続はLAN（有線）接続されているものとして説明します。
以降の説明では、以下の設定値に変更することとします。

ネットワークの設定	
IPアドレス	192.168.128.210
サブネットマスク	255.255.255.0
ゲートウェイ	192.168.128.254
DNSサーバ	192.168.128.1

Table 4.3-1 ネットワーク設定

- ① Ubuntuデスクトップ右上端の  ボタンをクリックし、ポップアップメニューにて設定を選択します。



- ② 設定ダイアログが表示されましたら、左ペインにてネットワークを選択しネットワークの設定ページを開きます。

4.4 NFS

NFS(Network File System)とはネットワークを介した分散ファイルシステムです。

NFSを使用すれば、ゲストOS (Ubuntu) の共有ディレクトリのファイルを他のLinuxマシンから共有することができます。

本章では、ゲストOS上でNFSサーバを起動し、共有ディレクトリにマウントします。

- ① 端末が起動していない場合は、起動します。
起動方法に関しては、『[4.2 端末アプリケーションの起動](#)』でご確認ください。

- ② 『apt』 コマンドを使用して、インストールを行います。

```
$ sudo apt install -y nfs-kernel-server ←
パッケージリストを読み込んでいます... 完了
依存関係ツリーを作成しています... 完了
状態情報を読み取っています... 完了
以下の追加パッケージがインストールされます:
  keyutils libevent-core-2.1-7 libnfsidmap1 nfs-common rpcbind
提案パッケージ:
  open-iscsi watchdog
以下のパッケージが新たにインストールされます:
  keyutils libevent-core-2.1-7 libnfsidmap1 nfs-common nfs-kernel-server
  rpcbind
アップグレード: 0 個、新規インストール: 6 個、削除: 0 個、保留: 276 個。
615 kB のアーカイブを取得する必要があります。
この操作後に追加で 2,235 kB のディスク容量が消費されます。
取得:1 http://jp.archive.ubuntu.com/ubuntu jammy/main amd64 libevent-core-2.1-7 amd64 2.1.12-
stable-1build3 [93.9 kB]
:
<途中省略>
:
Creating config file /etc/default/nfs-kernel-server with new version
man-db (2.10.2-1) のトリガを処理しています ...
libc-bin (2.35-0ubuntu3.8) のトリガを処理しています ...
```

- ③ NFS共有ディレクトリ『/nfs』を作成し、パーミッションを設定します。

```
$ sudo mkdir /nfs ←
$ sudo chmod 777 /nfs ←
```

5. クロス開発環境用パッケージ

CPUボードで動作するLinuxカーネルやアプリケーションプログラムを作成するには、クロス開発環境を構築する必要があります。クロス開発環境はYocto Projectを使用します。本章ではYocto Projectによる開発環境に必要なパッケージのインストールをします。

5.1 クロス開発環境について

クロス開発環境とは、以下の図のように実際に動作するシステム（CPUボード）とは異なるシステム（PC/AT互換機）で開発ができる環境のことを言います。

クロス開発環境で使用する特殊なコンパイラやアセンブラをクロスコンパイラやクロスアセンブラと呼びます。

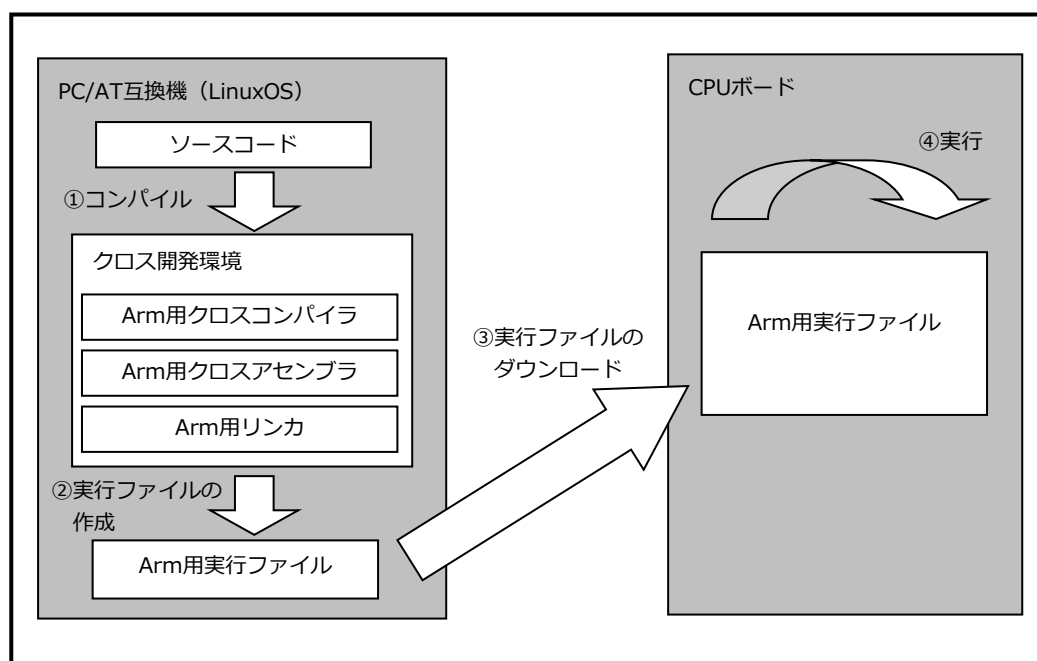


Fig 5.1-1 クロス開発環境

5.2 パッケージのインストール手順

必要なライブラリ等のパッケージを『apt』コマンドを使用してインストールします。
パッケージのインストール手順を以下に説明します。

- ① 端末が起動していない場合は、起動します。
起動方法に関しては、『[4.2 端末アプリケーションの起動](#)』でご確認ください。

- ② パッケージ情報を更新します。

```
$ sudo apt update ←
```

- ③ 開発に必要なパッケージをインストールします。

```
$ sudo apt install -y gawk wget git diffstat unzip texinfo gcc-multilib ¥ ←
> build-essential chrpath socat cpio python3 python3-pip python3-pexpect ¥ ←
> xz-utils debianutils iputils-ping libssl1.2-dev xterm curl p7zip-full ¥ ←
> libncurses5-dev libyaml-dev libssl-dev bmap-tools ←
パッケージリストを読み込んでいます... 完了
依存関係ツリーを作成しています... 完了
状態情報を読み取っています... 完了
debianutils はすでに最新バージョン (5.5-1ubuntu2) です。
debianutils は手動でインストールしたと設定されました。

:
<途中省略>
:

libglx-dev:amd64 (1.4.0-1) を設定しています ...
libgl-dev:amd64 (1.4.0-1) を設定しています ...
libglu1-mesa-dev:amd64 (9.0.2-1) を設定しています ...
libssl1.2-dev (1.2.15+dfsg2-6) を設定しています ...
```



依存関係によりインストールされるパッケージは省略しております。

本文書について

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



株式会社アルファプロジェクト
〒431-3114
静岡県浜松市中央区積志町834
<https://www.apnet.co.jp>
E-Mail : query@apnet.co.jp
