

本資料は、サイバートラストが提供する「組み込み Linux 開発力養成講座」のテキストを一部抜粋したものです。複製、改変、頒布などを行うことはできません。
本資料に記載されている情報は予告なしに変更されることがあります。

組み込みLinux開発力養成講座 テキスト

サイバートラスト(株) ミツ木 祐介



はじめに

SAMPLE

■ 講師

● 三ツ木祐介

- サイバートラスト株式会社勤務
 - IoT事業本部 システム開発部所属
 - Yocto Projectを使用した組み込み Linuxポータリングが得意
- リネオソリューションズ株式会社(出向)
 - [Yocto Project 公式実践講座 \(LFD461-JP\)](#)講師
- 個人活動
 - [みつきのブログ](#)
 - CQ出版「YoctoProjectではじめる組み込み Linux開発入門」

SAMPLE

- 本講座の目的
- ハンズオン実施環境について
- 組込みLinuxについて
- Yocto Projectについて
- 演習1: 最初のBitBake実行
- Yocto Projectのビルドシステム
- 演習2: ボードの起動
- Monoについて
- 演習3: meta-monoの組込み
- 作成したレシピの解説
- Q&A

SAMPLE

本講座の目的

SAMPLE

- 組込みLinuxの概要を理解する
 - 組込みLinuxとは
 - メリット / デメリット
- Yocto Projectの基本的な使い方を理解する
 - 基本的な考え方
 - レシピやレイヤの作成方法
- .NET Frameworkアプリケーションの移植方法を理解する
 - 移植にはMonoを使用
 - Monoとは

SAMPLE

ハンズオン実施環境について

SAMPLE

PCに以下の環境が構築されていることが前提となります。

- Ubuntu 22.04 (VM, Baremetal は問わない)がインストールされている PC
- Ubuntu が WiFi を経由してローカル NW にアクセスできること
- USB 2.0 Type Aが接続できること
- Ubuntu 上のシリアルコンソールが USB-Serial に接続できること

SAMPLE

■ NXP 8MPLUSLPD4-EVK

- SoC
 - i.MX 8M Plus Quadアプリケーション・プロセッサ
 - Arm[®] Cortex-A53(最大1.8 GHz)x 4
 - Arm[®] Cortex-M7(最大800 MHz)x 1
 - Cadence[®] Tensilica[®] HiFi4 DSP(最大800 MHz)
 - ニューラル・プロセッシング・ユニット
- メモリ
 - 6 GB LPDDR4
- ストレージ
 - 32 GB eMMC 5.1
 - 32 MB QSPI NOR
 - micro SDカード

SAMPLE

組込みLinuxについて

SAMPLE

以下に組み込みLinuxを使用するメリット挙げてみます。

- さまざまなアーキテクチャ、ボードに移植されている
- オープンソースソフトウェア(OSS)との親和性が高い
- 目的に合わせたカスタマイズが容易
- カーネルのソースコードが公開されており、ドライバなどの開発が比較的容易
 - ただし、一から開発することは稀
- ネットの情報やマニュアルなど参照できる情報が多い

SAMPLE

Yocto Projectについて

SAMPLE

Yocto Projectは、組込み機器、IoT機器向けにカスタマイズされたLinuxを作成するための仕組みを提供するオープンソース・コラボレーションプロジェクトです。

Yocto ProjectはOpenEmbeddedと共同で開発している BitBakeとメタデータを使用することで、ビルド環境からデバイスを起動するためのディスクイメージまでをコマンド1つで作成することができるようになっています。

後に説明するレイヤモデルを使用することで、BSPのレイヤを差し替えることで、別のボードにアプリケーションを移植することも比較的容易になっています。

SAMPLE

Yocto Projectのビルドシステム

SAMPLE

■ Poky

- bitbake
- メタデータ
- ドキュメント

■ その他ツール

- Toaster
- recipetool
- Autobuilder
- ..etc



PokyとはYocto Projectが提供するリファレンスの名称となります。
[リファレンスマニュアル](#)によると Pokyの読み方は以下のようになっています。

Poky, which is pronounced Pock-ee

一節によるとお菓子のポッキーが由来となっているそうです。

Pokyには次の2つの側面があります。

- リファレンスビルドシステムの名称
- リファレンスディストリビューションの名称

SAMPLE

Monoについて

SAMPLE

Monoとは .NET Frameworkのオープンソース実装です。
Microsoftが提供していた.NET Frameworkとの違いは以下のようになります。

- .NET Framework
 - 仕様は公開
 - オープンソースではない
 - Windowsのみで動作
- Mono
 - ECMA標準のC#およびCommon Language Runtime(CLR)の仕様に基づく
 - オープンソース
 - Windows以外でも動作

現在はマイクロソフトがスポンサーとなっており、すでにオープンソースとなっている.NET 5以降のフレームワークにコードが統合されています。

演習3: meta-monoの組み込み

SAMPLE

- meta-monoの組み込み
- C#アプリの作成
- カスタムレイヤの作成
- アプリのレシピ化
- ボード上で動作確認

SAMPLE

講師PCからダウンロードしたデータの中にmeta-monoも含まれています。
ビルドディレクトリで下記のコマンドを実行し、meta-monoをビルド対象に追加します。

[3.1]

```
$ bitbake-layers add-layer ../sources/meta-mono
```

追加されたことを確認します。[3.2]

```
$ bitbake-layers show-layers
```

SAMPLE

レシピ格納ディレクトリに移動します。[3.11]

```
$ pushd ../sources/meta-mono-app/recipes-app/mono
```

hello.csをfilesディレクトリに格納します。[3.12]

```
$ cp /tmp/hello.cs ./files
```

SAMPLE

講師PCから起動用スクリプトをダウンロードします。[3.13]

```
$ wget http://DESKTOP-7DDDDD7/example/hello.in -O  
./files/hello.in
```

起動スクリプトhello.inの内容を以下に示します。

```
#!/bin/sh
```

```
exec @MONO@ @APP@ "$@"
```

“@MONO@”と“@APP@”はレシピによって適切に置き換えられます。

SAMPLE

講師PCからレシピをダウンロードします。[3.14]

```
$ wget http://DESKTOP-7DDDDD7/example/hello-cs_0.1.bb
```

ビルドディレクトリに移動します。[3.15]

```
$ popd
```

SAMPLE

conf/local.confでhello-csパッケージを追加します。[3.16]

```
$ echo 'IMAGE_INSTALL:append = " hello-cs"' >>  
./conf/local.conf
```

イメージをビルドします。[3.17]

```
$ bitbake core-image-base -k
```

SAMPLE