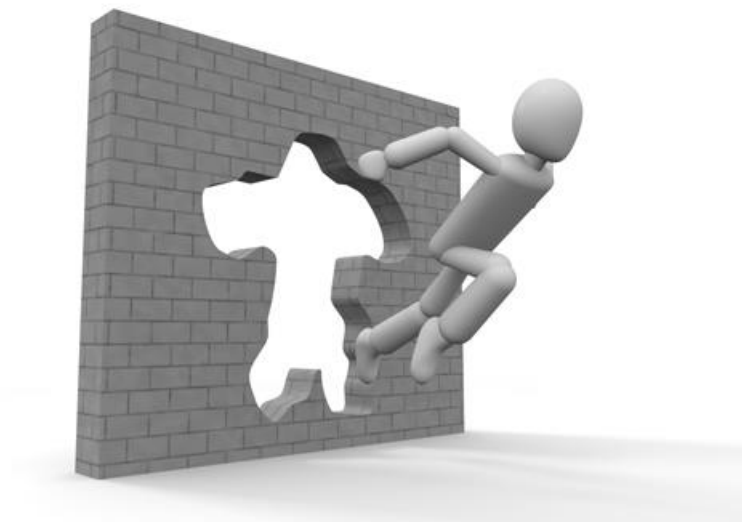


# EV3 開発環境の構築

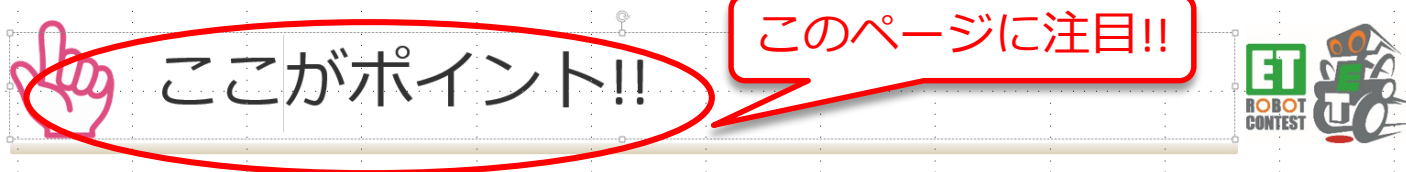
ここで、EV3開発環境をちゃちゃっと構築して、  
これから始まる“技術教育”に挑もうではないか！



# 本日のルール



- グループ内 (参加者同士) で助け合うこと
  - グループ内で解決できなければ、実行委員にサポートをお願いします
- グループ全員で”(本日の) ゴール”しましょう
- このスライドの見方
  - EV3 開発環境構築ガイド (TOPPERS/EV3RT) を主とします  
[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/WhatsEV3RT](http://dev.toppers.jp/trac_user/ev3pf/wiki/WhatsEV3RT)
  - スライドでは各工程のポイントとなることを紹介します



- EV3RT のパッケージ (ev3rt-beta6-2-release.zip)
  - 最終更新日が “3/17” 以降の “β6-2” をダウンロード

# 本日のゴール (おさらい)



- グループ全員で以下のことを達成すること！
  - ✓ TOPPERS/EV3RT 開発環境を構築する
  - ✓ サンプルプログラムをビルドし、正常にコンパイルできることを確認する
  - ✓ コンパイル済みアプリケーションを走行体上で実行し、正常に動作することを確認する

- 参加者同士“なかよし”になること！



# EV3 開発環境の構築 – アジェンダ



1. 持ち物確認
2. 開発プラットフォーム EV3RT のダウンロード & インストール
3. 開発環境 (クロスコンパイラー, ツール) のインストール
4. サンプルプログラムのダウンロード
5. サンプルアプリケーションのビルドと実行
6. サンプルコースの試走

# 1.持ち物確認

まずは、忘れ物がないか確認しましょう



# ここがポイント!!



## ■ 開発PC

- ✓ Windows / Mac

## ■ EV3走行体（+本体とパーツをつなげるケーブル）

- ✓ EV3way / HackEV

走行体は組み立て図どおりに組み立てたもの

## ■ microSDカード（+ SDカードアダプター）

- ✓ SDHC 4GB 以上

## ■ インターネットに接続できる機器

- ✓ もしくは事前にダウンロード

## ■ サンプルコース

- ✓ 持参できる方のみ

## 2.開発プラットフォーム EV3RT のダウンロード & インストール

[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/DevEnv](http://dev.toppers.jp/trac_user/ev3pf/wiki/DevEnv)  
#開発環境構築のマニュアル



[Login](#) | [Preferences](#) | [Help/Guide](#) | [About Trac](#)

|  |

wiki: [DevEnv](#) [Start Page](#) | [Index](#) | [History](#)

### 開発環境構築のマニュアル

#### 開発プラットフォームEV3RTのダウンロード&インストール

ダウンロードページからEV3RTのパッケージをダウンロードし、以下の手順に従ってインストールしてください。

- ① zipファイルを解凍し、プラットフォームのパッケージ (hrp2.tar.xz) を作業フォルダに移動します。
  - 作業フォルダの場所は基本的に自由ですが、パスにスペースや全角文字が含まれている場所は避けてください。
- ② ターミナルを開いて作業フォルダに移動します (もし、Windows環境でCygwinをインストールしていない方は、先に「開発環境 (クロスコンパイラ, ツール) のインストール」を実施してください)。
- ③ 下記のコマンドでプラットフォームの簡易パッケージを解凍します
  - 注意: 解凍ソフトを使用して解凍すると、アーカイブに含まれるシンボリックリンクが壊れてしまうことがありますので、以下のように tar コマンドを使用してください。

```
$ tar xvf hrp2.tar.xz
```

作業フォルダに、hrp2 フォルダができれば、プラットフォームのインストールが完了です。

**Table of Contents**  
[TOPPERS/EV3RT とは?](#)  
[ダウンロード](#)  
[開発環境構築のマニュアル](#)  
[サンプルアプリケーションのビルドと実行](#)  
[ユーザマニュアル \(アプリケーション開発者向け\)](#)  
[EV3RT C APIリファレンス](#)  
[EV3RT C++ APIリファレンス](#)  
[FAQ](#)  
[お問い合わせ](#)



# ここがポイント!!



- EV3RT のパッケージ (ev3rt-beta6-2-release.zip)
  - 最終更新日が “3/17” 以降の “β6-2” をダウンロード

## ダウンロード

### パッケージ

リリース	公開範囲	最終更新日	ファイル
β6-2	一般	2016年03月17日	<a href="#">➡ ev3rt-beta6-2-release.zip</a>
β6-1	一般	2016年03月14日	<a href="#">➡ ev3rt-beta6-1-release.zip</a>
β6	一般	2016年03月02日	<a href="#">➡ ev3rt-beta6-release.zip</a>
β5-2	一般	2015年12月09日	<a href="#">➡ ev3rt-beta5-2-release.zip</a>

[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/Download](http://dev.toppers.jp/trac_user/ev3pf/wiki/Download)





## ■ 手順①

- Cygwin インストールフォルダー配下に “作業フォルダー” を配置します
- そのため、EV3RT のパッケージをダウンロードしたら、先に「開発環境 (クロスコンパイラ, ツール) のインストール」を実施します
- “作業フォルダー” は、以下とします  
32bit 版 OS ・ ・ ・ C:¥cygwin¥ev3rt  
64bit 版 OS ・ ・ ・ C:¥cygwin64¥ev3rt



# ここがポイント!! (Windows 編) - 2/2



## ■ 手順②

- Cygwin を起動して、以下のコマンドを実行して“作業フォルダー (ev3rt)”に移動します

```
$ cd /ev3rt
```

\$は表示されているので、cd /ev3rt を入力します。

※管理者権限が付与されていないログインユーザでCygwinを起動する際には、Cygwin Terminalアイコンを右クリックして“管理者として実行”を選択して起動してください。※以降のCygwinを起動する際も同様です。

## ■ 手順③

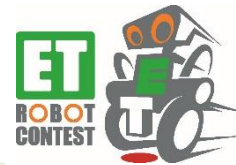
- 以下のコマンドを実行して、“hrp2.tar.xz”を解凍します

```
$ tar xvf hrp2.tar.xz
```

※“ev3rt-beta6-2-release.zip”を解凍した中身は、“作業フォルダー (ev3rt)”に置きましょう。  
“hrp2.tar.xz”は、これに含まれています。



# ここがポイント!! (Mac 編) - 1/2



## ■ 手順①

- “ev3rt-beta6-2-release.zip” をダブルクリックで解凍  
→ “ev3rt-bata6-2-release” というフォルダーができます
- 上記フォルダーを「ユーザ名のフォルダー (※)」の中に  
ドラッグ & ドロップで移動します  
※Finder の「よく使う項目」一覧の中の「家のアイコン」  
がついているところ
- フォルダーの名前が長いので、リネームします  
“ev3rt-bata6-2-release” → “ev3rt”



# ここがポイント!! (Mac 編) - 2/2



## ■ 手順②

- ターミナルは、[アプリケーション] > [ユーティリティ]の中にあります
- ターミナルで以下のコマンドを実行し、“作業フォルダー (ev3rt)” に移動します

```
$ cd ~/ev3rt
```

## ■ 手順③

- 以下のコマンドを実行して、“hrp2.tar.xz” を解凍します

```
$ tar xvf hrp2.tar.xz
```

# 3.開発環境 (クロスコンパイラー, ツール) のインストール

Windows 編

[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/DevEnvWin](http://dev.toppers.jp/trac_user/ev3pf/wiki/DevEnvWin)



**TOPPERS**  
Toyohashi Open Platform  
for Embedded Real-time Systems

Wiki: DevEnvWin

## 開発環境のインストール (Windows + Cygwin編)

Windows + Cygwinで開発する場合、以下の手順で開発環境を構築することができます。

① Cygwinのインストーラをダウンロードします

- ⇒ Cygwinのサイトからインストーラをダウンロードすることができます
- 64-bit版Cygwinのインストーラ(⇒ setup-x86\_64.exe)をオススメします
- 32-bit版Cygwin(⇒ setup-x86.exe)でも動作できる報告があります
- Cygwinのインストーラ(setup-x86\_64.exe又はsetup-x86.exe)を起動します

Cygwin Setup  
Cygwin Net Release Setup Program  
This setup program is used for the initial installation of the Cygwin environment as well as all subsequent updates. Make sure to remember where you saved it.  
The pages that follow will guide you through the installation. Please note that Cygwin consists of a large number of maintenance operations to the satisfaction of customers. We note.



# ここがポイント!! (Windows 編)



## ■ 手順⑥

- ダウンロードサイト → “<http://ftp.jaist.ac.jp>”

## ■ 手順⑩

GCC ARMのサイトのリンクは使わないでください。

- 動作確認済みバージョンを使用しましょう  
→ gcc-arm-none-eabi-4\_8-2014q3-20140805-win32.exe

## ■ 手順⑮

- “Mkimage.tar.gz” を “作業フォルダー (ev3rt)” に置きましょう  
32bit 版 OS . . . C:\cygwin\ev3rt  
64bit 版 OS . . . C:\cygwin64\ev3rt
- Cygwin を起動して、以下のコマンドを実行して  
“作業フォルダー (ev3rt)” に移動します

```
$ cd /ev3rt
```



# ここがポイント!! (Windows 編)



## ■ ダウンロードパッケージの確認

- Cygwinを起動して、以下のコマンドを実行します。

```
$ cygcheck -c gcc-core make diffutils perl
```

- パッケージの情報が出力されるので、バージョンが  
ていること、StatusがOKであることを確認します。

### Cygwin Package Information

Package	Version	Status
diffutils	3.3-3	OK
gcc-core	5.3.0-5	OK
make	4.1-1	OK
perl	5.22.1-2	OK

コマンドの実行結果の例です。  
手順⑦に書かれているバージョンより  
新しいことを確認します。

# 3.開発環境 (クロスコンパイラー, ツール) のインストール

Mac 編

[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/DevEnvMac](http://dev.toppers.jp/trac_user/ev3pf/wiki/DevEnvMac)



**TOPPERS**  
Toyohashi Open Platform  
for Embedded Real-time Systems

Search

Login | Preferences | Help/Guide | About Trac

Wiki | View Tickets | Search

wiki: DevEnvMac

Start Page | Index | History

## OS Xでの環境構築方法

### 前提となる環境

予め以下のソフトウェアがインストールされているものとします。

- gcc
  - Xcodeをインストール(App Store から入手可能)or [macport](#) でインストール

### mkimageのインストール

- ⇒ [ここ](#)からバイナリファイルをダウンロードし、パスの通ったディレクトリに置く

```
cp mkimage /usr/local/bin/
```

- ソースコードからインストールする場合は、[こちら](#)を参照してください。

### GNUツールチェーンのインストール

- ⇒ Mac用バイナリが、[GCC ARMのサイト](#)からダウンロードすることができます





# ここがポイント!! (Mac 編) - 1/3



## ■ 前提となる環境

- gcc がインストールされているか確認します

```
$ make -v
```

## ■ mkimage のインストール

- “/usr/local/bin” にパスが通っているか確認します

```
$ echo $PASH
```

表示された文字列に “/usr/local/bin” が含まれていることを確認します

- “mkimage” ファイルを “/usr/local/bin” に置きます

```
$ cd ~/Downloads
```

“mkimage” ファイルをダウンロードしたディレクトリ

```
$ cp mkimage /usr/local/bin/
```



# ここがポイント!! (Mac 編) - 2/3



## ■ GNUツールチェーンのインストール

- 動作確認済みバージョンを使用しましょう  
→ gcc-arm-none-eabi-4\_8-2014q3-20140805-mac.tar.bz2
- Finder でダブルクリックして解凍します
- 特にこだわりがなければ、“/usr/local” に置きましょう

```
$ mv gcc-arm-none-eabi-4_8-2014q3 /usr/local/
```

- パスを通します。bash を使っていない方は、ツールに応じた設定を行ってください

実際は1行で記述します

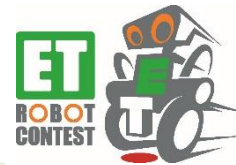
```
$ echo "export PATH=/usr/local/gcc-arm-none-eabi-4_8-2014q3/bin:¥$PATH" >> ~/.bash_profile
```

- 以下のコマンドで、設定を反映します

```
$ source ~/.bash_profile
```



## ここがポイント!! (Mac 編) - 3/3



- EV3RT のコンフィギュレータ (cfg) のインストール
  - 「2.開発プラットフォーム EV3RT のダウンロード & インストール」の“ev3rt”フォルダーに配置します
  - ダウンロードした“cfg-osx-static-1\_9\_4.tar.gz”を Finder でダブルクリックで解凍します  
→ 拡張子のない“cfg”というファイルができます
  - “cfg”ファイルを“~/ev3rt/hrp2/cfg/cfg/ (※)”に置きます  
→ Finder でドラッグ & ドロップで移動します  
※“cfg.exe”や“cfg.cpp”などがあるフォルダーです

# 4. サンプルプログラムのダウンロード

[https://github.com/ETrobocon/etroboEV3/wiki/sample\\_program](https://github.com/ETrobocon/etroboEV3/wiki/sample_program)



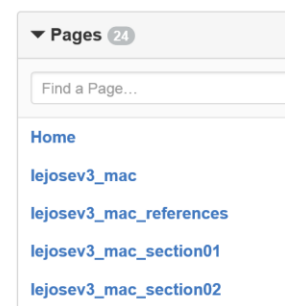
## sample\_program

Jiro Tanahashi edited this page on Mar 6 · 3 revisions

## サンプルプログラム

**【重要】**ここで提供されているEV3RTのサンプルプログラムは、β5-2用です。β6では動作しません。β6は現在検証中ですので、もうしばらくお待ちください。

EV3way-ET走行体についてサンプルプログラムを提供します。ETロボコン2016でEV3用として公式にサポートする開発環境、EV3RT、MonoBrick、leJOS EV3用のプログラムを用意しています。（mruby on EV3RT + TECSのサンプルプログラムは、配布パッケージに同梱されています。）





## ここがポイント!! (Windows 編)



- β6-2は検証済みですので、以下は問題ありません  
【重要】ここで提供されているEV3RTのサンプルプログラムは、β5-2用です。β6では動作しません。β6は現在検証中ですので、もうしばらくお待ちください。
- etroboEV3-master.zipを解凍してできたフォルダーを移動します
  - 移動元  
etroboEV3-master¥SampleCode¥  
EV3way\_EV3RT\_sample¥sample\_c4
  - 移動先  
32bit 版 OS . . . C:¥cygwin¥ev3rt¥hrp2¥sdk¥workspace  
64bit 版 OS . . . C:¥cygwin64¥ev3rt¥hrp2¥sdk¥workspace

EV3way 用  
サンプルプログラム



## ここがポイント!! (Mac 編)



- etroboEV3-master.zipをFinder でダブルクリックで解凍します
  
- 解凍してできたフォルダーを移動します
  - ※ドラッグ & ドロップで OK です
  - 移動元  
etroboEV3-master¥SampleCode¥  
EV3way\_EV3RT\_sample¥sample\_c4
  - 移動先  
ev3rt/hrp2/sdk/workspace

# 5. サンプルアプリケーションのビルドと実行

[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/SampleProgram](http://dev.toppers.jp/trac_user/ev3pf/wiki/SampleProgram)  
#サンプルアプリケーションのビルドと実行



[Login](#) | [Preferences](#) | [Help/Guide](#) | [About Trac](#)

[Wiki](#) | [View Tickets](#) | [Search](#)

[Start Page](#) | [Index](#) | [History](#)

wiki: [SampleProgram](#)

## サンプルアプリケーションのビルドと実行

### 実行形式とビルド方法

本プラットフォームのアプリケーションは、以下の2つの方法で実行することができます。

- 動的ローディング形式: OSとアプリケーションの実行ファイルを分け、OS実行中に、ダイナミックローディング機能を使ってアプリケーションの実行ファイルだけをロードして実行する。
- スタンドアローン形式: ダイナミックローディング機能を使わず、OSとアプリケーションを1つの実行ファイルにして単独(スタンドアローン)で実行する。

### 動的ローディング形式(ダイナミックローディング機能を利用)

#### 動的ローディング形式とは?

動的ローディング形式は、ダイナミックローディング機能をもつアプリケーションローダ(apploader)を使って、SDカードに入れた、またはBluetooth等のシリアル通信で受信したアプリケーションの実行ファイル(ELF形式)をロードして動的に実行する形式です。

**Table of Contents**  
[TOPPERS/EV3RT とは?](#)  
[ダウンロード](#)  
[開発環境構築のマニュアル](#)  
[サンプルアプリケーションのビルドと実行](#)  
[ユーザマニュアル \(アプリケーション開発者向け\)](#)  
[EV3RT C APIリファレンス](#)  
[EV3RT C++ APIリファレンス](#)  
[FAQ](#)  
[お問い合わせ](#)



# ここがポイント!! - 1/2



## ■ 本日は動的ローディング形式でビルドします

### 1. ビルドは、“workspace” フォルダで行います

- Windows + Cygwin の場合

```
$ cd /ev3rt/hrp2/sdk/workspace
```

- Mac の場合

```
$ cd ~/ev3rt/hrp2/sdk/workspace
```

### 2. サンプルプログラムをビルド (コンパイル) します

```
$ make app=sample_c4
```

EV3way 用  
サンプルプログラム





# ここがポイント!! - 2/2



## ■ 本日は動的ローディング形式でビルドします

### 3. SD カードにコピー

- ✓ 動的ローディング形式の場合、初回のみアプリケーションの他にアプリケーションローダーもコピーする必要があります

※ただし、EV3RT のバージョンが変わった場合は、そのバージョン用のアプリケーションローダーをコピーし直してください

- “ev3rt/sdcard” フォルダの中にある “ev3rt” フォルダと “ulImage” ファイルを SD カードにコピーします
- SD カードにコピーした “ev3rt” フォルダの中の “apps” フォルダに、先ほどビルドした “app” ファイルをコピーします
- ✓ “USB 経由” と “Bluetooth 経由” は、各自でお試してください

## 6. サンプルコースの試走

走行体上でサンプルプログラムを実行してみましよう



# ここがポイント!!



## ■ EV3way のサンプルプログラム

- ✓ タッチセンサーを押したときにジャイロセンサーのリセットを行います
- ✓ そのため、走行体をできるだけ揺らさないようにタッチセンサーを押すことがうまく走らせるコツです

## ■ HackEV のサンプルプログラム

- ✓ HackEVのサンプルプログラムは提供していません
- ✓ しかし、光センサーとタイヤの配置がEV3wayと似ているため、EV3wayのサンプルプログラムの倒立制御と尻尾モーターの制御を無効にするとそれなりのものを作れます
- ✓ 使用するポートに合わせてソースを修正する必要があります

# EV3 開発環境の構築

## 【参考文献】

- ◆ ETロボコン EV3開発環境構築ガイド -github.com  
<https://github.com/ETrobocon/etroboEV3/wiki>
- ◆ TOPPERS/EV3RT ウェブサイト - TOPPERS  
[http://dev.toppers.jp/trac\\_user/ev3pf/wiki/WhatsEV3RT](http://dev.toppers.jp/trac_user/ev3pf/wiki/WhatsEV3RT)
- ◆ Mac OS X でETロボコンに出よう！ (EV3RT 環境構築編) - Qiita  
<http://qiita.com/yamanekko/items/3cee3991ea9472a1f6b6>