

要素技術とモデルを開発に使おう

ETロボコン実行委員会 - er-info@etrobo.jp - 2.0, 2016-05-21 12:56:41 | 2016年用

この教材について

目的

- この教材は、ETロボコンに参加されるみなさんに、モデル作成に必要な知識やスキル取得の機会を提供することを目的に作成しています

著作

- この教材はETロボコン実行委員会が作成したものです
- この教材の著作権は、ETロボコン実行委員会に帰属します

使用について

- ETロボコンの参加資格（企業・大学・個人）の範囲内に限り、ご自由に活用していただいてもかまいません

はじめに

本コースの想定する受講者

C言語/C++言語、あるいはこれらに類似の言語のプログラミング経験はあるが、モデルを使った開発経験のないエンジニア

トレーニングのゴール

- モデルとコードのつながりを知っている
- 要素技術をモデルに組み込んで使える

トレーニングの進め方

- コードの状況をモデルで表すことで、モデルが表すものを実感します
- モデルとコードを並行して編集し、モデルとコードの対応づけを学びます
- 要素技術を実験してからモデルに組み込み、そのモデルからコードを作成します

目次

text00.html

- 準備

text01.html

- sample00（ウォーカー）を動かしてみよう
- sample01:システムの処理を担当するクラスを追加する
- sample01の構造をクラス図に表す
- sample01のコードを作成する
- ここまでのまとめ

text02.html

- sample03（ライントレーサ）を動かしてみよう
- sample03の振舞いの図を描いてみよう
- sample03の課題を検討しよう
- sample04:比例制御の実験をやろう
- sample05:検討結果をモデル図に反映する
- sample05の構造をクラス図に表す
- sample05のコードを作成する
- ここまでのまとめ

text03.html

- sample_c4（倒立振子のサンプル）を試す
- sample_4cの構造をクラス図に表す
- sample_cpp4:倒立振子ライブラリの呼び出し方を考える
- sample_cpp4の構造をクラス図に表す
- sample_cpp4のコードを作成する
- ここまでのまとめ
- トレーニングのまとめ

準備 1：ロボットキットのチェック

必ず、2016年版の「EV3way-ET組み立て手順書」を確認しましょう。

最初は倒立振子を使わないで演習します。

そのため、このトレーニングで使う「トレーニング用補助脚」がないと困ります。

1. ロボットの前部に「トレーニング用補助脚」を取りつけてありますか

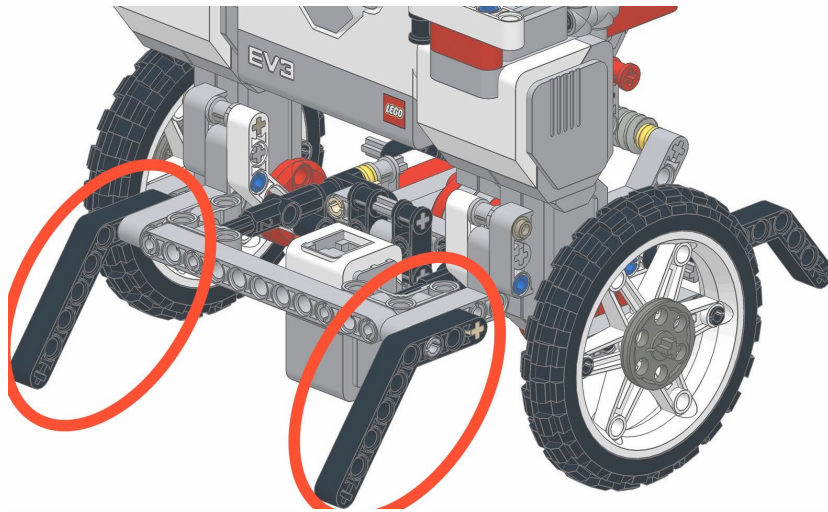


図 1. トレーニング用補助脚（前部）

2. ロボットの後部にも「トレーニング用補助脚」を取りつけてありますか

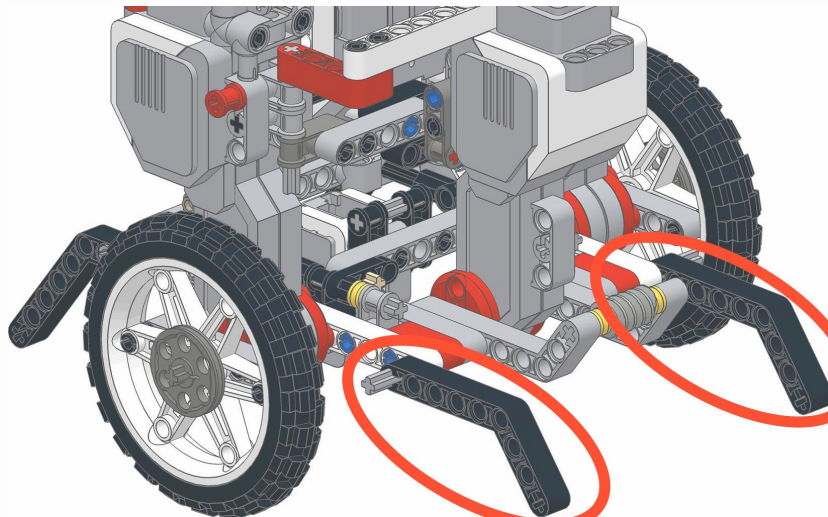


図 2. トレーニング用補助脚（後部）

3. ロボットの前後でリフトアームを取り付ける向き（長短）が逆なことに気をつけます

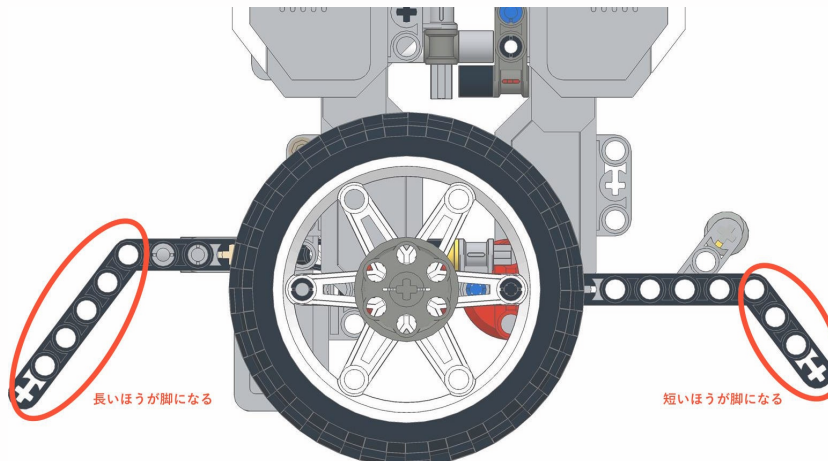


図 3. 「トレーニング用補助脚」の装着状態を横からみた図

準備 2：EV3用の環境構築

Mindstorms EV3を使う場合は次の構築手順に従います。

- ETロボコン EV3開発環境構築ガイド
 - <https://github.com/ETrobocon/etroboEV3/wiki>
- 開発環境構築の課題と対処
 - https://github.com/ETrobocon/etroboEV3/wiki/problem_and_coping

準備 3：ターミナルの使い方の確認

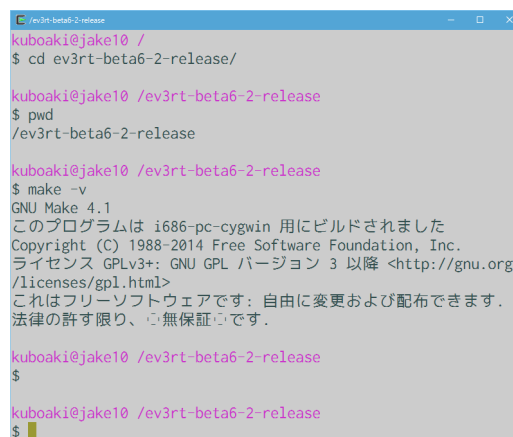
Cygwinやターミナルが動作することを確認します。

Windowsユーザー:

- Cygwinのシェルは起動しますか
- Unixの基本コマンド（cd, ls, makeなど）は動作しますか

Mac OSXユーザー:

- ターミナル（Terminal.app）は使えますか
- コマンド使えることを確認しておきましょう



```

kuboaki@jake10 /
$ cd ev3rt-beta6-2-release/

kuboaki@jake10 /ev3rt-beta6-2-release
$ pwd
/ev3rt-beta6-2-release

kuboaki@jake10 /ev3rt-beta6-2-release
$ make -v
GNU Make 4.1
このプログラムは i686-pc-cygwin 用にビルドされました
Copyright (C) 1988-2014 Free Software Foundation, Inc.
ライセンス GPLv3+: GNU GPL バージョン 3 以降 <http://gnu.org/licenses/gpl.html>
これはフリーソフトウェアです: 自由に変更および配布できます.
法律の許す限り、無保証です.

kuboaki@jake10 /ev3rt-beta6-2-release
$

kuboaki@jake10 /ev3rt-beta6-2-release
$
```

図 4. cygwinのシェル画面

Unix のコマンドに慣れておきましょう

Unixの基本コマンド（cd, ls, cp, makeなど）や、シェルの操作方法、ファイルシステムの表現方法を確認しておきましょう。

モデリングツールを用意する

astah* Professionalの参加者向けライセンスが提供されるので、モデリング演習にはこのツールを使います。

Windows PCだけでなく、MacやLinuxでも利用できます。

1. 参加者向けの入手方法に従ってツールをダウンロードします
2. インストール手順に沿ってインストールします



モデリングツールの利用は、演習の円滑を意図したもので、提出モデルの作成に使用するツールを強制しているわけではないことに注意しましょう。

参加者向けライセンスを設定します

同じライセンスで、astah* SysMLや、astah* GSNも利用できます。

1. 参加者向けの入手方法に従ってライセンスを入手します
2. 提供される手順に従って、astah* Professionalを起動し、ライセンスを設定します
 - Windows PCの場合、ライセンスの設定時だけは「管理者で実行」で起動する必要があります

テキストエディタを用意する

「メモ帳で済ませよう」なんて、ダメ、ダメ、絶対！

プログラミング用のテキストエディタや、Eclipseのような統合環境（IDE）を使いましょう。

テキストエディタの例

Windows用:

- Notepad++、TeraPad、サクラエディタ、Atom、Brackets、Emacs、Vim、Sublime Text、Visual Studio Codeなど

Mac OSX用:

- CotEditor、mi、TextMate、Atom、Brackets、Emacs、Vim、Sublime Text、Visual Studio Codeなど

テキストエディタに慣れておく

このトレーニングでは、テキストエディタとコマンドラインツールで演習します。

トレーニング中は、みなさんで同じやり方で進めましょう。

基本操作を確認しましょう

- ファイルの作成、削除、保存、別名での保存、扱えるファイル拡張子の確認
- 文字単位、単語単位、行頭・行末、文頭・文末などへのカーソルの移動
- テキストの入力、削除、コピーやペースト、検索、置換
- その他提供されているコマンドの使い方

ワークスペースを作成する

EV3RTの開発環境では、プログラムを作る場所をワークスペースと呼びます。
この演習で使うためのワークスペースを作成しましょう。

EV3RTをインストールしたディレクトリ中の **hrp2/sdk** ディレクトリに移動します。

```
$ cd (ev3rtのインストールディレクトリ) /hrp2/sdk
$ ls -l ①
common/
LICENSE
Makefile.sdk.conf
workspace/
```

- ① **ls** はファイルリストの表示コマンド、**-l**（ハイフン・イチ）は、縦1列に出力表示するオプション

このディレクトリの中に **beginners** ディレクトリを作成します。

```
$ mkdir beginners ①
$ ls -l
beginners/ ②
common/
LICENSE
Makefile.sdk.conf
workspace/
```

- ① beginnersディレクトリを作成する
② beginnersディレクトリが追加されている

サンプルコードの準備

サンプルコードは、**beginners.zip** の中の **sample_code** ディレクトリに含まれています。

ワークスペースとして作成した **beginners** ディレクトリの中に、**sample_code** の中に含まれる **Makefile** と **sample00** ディレクトリとその中身すべてをコピーします（エクスプローラーやFinderで操作してもかまいません）

コピーしたら、コピー先のディレクトリの内容を確認しましょう。

```
$ cd (ev3rtのインストールディレクトリ) /hrp2/sdk/beginners
$ ls
sample00/ Makefile
```

```
$ ls sample00
Makefile.inc  app.cfg  app.cpp  app.h  util.cpp  util.h
```

モデル図作成の準備

モデル作成に使うクラスが定義されているテンプレートは、 `beginners.zip` の中の `asta-template` ディレクトリに含まれています。

内容を確認しておきましょう。

```
$ cd (`beginners.zip` を展開した場所)
$ ls asta-template/
README.md
ev3rt-ver2.asta ①
テンプレートの使い方について.md*
JudeP.properties ②
JudeP.propertiesについて.md*
```

- ① EV3RT用C++ライブラリのテンプレートファイル（配布時期によってファイル名は変わることも）
- ② `astah*`をC++用に設定するためのプロパティファイル

2. テンプレートファイルをモデル保存用ディレクトリにコピーします
3. コピーできたかどうか確認します

```
$ cd (モデル保存用ディレクトリ)
$ ls
ev3rt-ver2.asta
```

モデル図を保存する場所が準備できました。

環境の使い方に慣れておく

Cygwin環境（Macではターミナル）の操作方法:

- ファイルやディレクトリの配置の理解
- Unixのシェルとコマンドの使い方

テキストエディタの操作方法:

- ファイルを作る、開く、保存する、閉じる
- 文章やコードを入力する、削除する、コピー&ペーストする

- 文字列を検索、置換する

モデリングツールの操作方法:

- プロジェクトファイルの作成、保存
- モデル図を追加、複製、削除する
- モデル図の要素の追加、移動、修正、削除

モデリング言語の基本:

- UMLのよく使う図や構成要素の名前や役割

プログラムの実行形式について

EV3のプログラムをビルド・実行する方法は2つあります。

- 動的ローディング形式（ローダを使って指定したプログラムを起動する方式）
- スタンドアローン形式（プログラム自身がブートイメージとなる方式）

このトレーニングでは「動的ローディング形式」を使用します。

動的ローディング形式の特徴

- EV3をリセットすることなく、アプリケーションプログラムを更新・追加できます
 - 加えて、USBケーブルを使って転送すれば、Bluetoothの再接続やSDカードの抜き差しなしに転送できます
- 1枚のSDカードに複数のアプリケーションが登録できます
 - ファイル名を変えて転送しておけば、実行時に選択して実行できます
- 短時間でビルドできます
 - OSやライブラリを一緒にビルドしないので、その分ビルド時間が短くなります

動的ローディング形式のビルド手順

1. Cygwinやターミナルを起動して、アプリケーションのソースコードが含まれるワークスペース（下記の例では **beginners**）へ移動します

```
$ cd (EV3RTのインストールディレクトリ) /hrp2/sdk/beginners
```

- 移動するのが、開発中のアプリケーションのディレクトリではないことに注意しましょう

2. 次のコマンドを実行します

```
$ make app=アプリケーションのディレクトリ名
```

ビルドに成功すると、**beginners** ディレクトリ中に **app** というファイル名のロードモジュール（EV3本体に転送するファイル）ができます

アプリケーションローダの準備

動的ローディング形式のプログラムはアプリケーションローダを使って実行します。
みなさんの使っているEV3RTの開発環境が提供するアプリケーションローダでないとプログラムは起動しません。

micro SDカード（以降、短くSDカードと書きます）を使いまわすときには、SDカード上のアプリケーションローダを1度書き換えておいたほうがよいでしょう。

1. SDカードにアプリケーションローダを書き込みます
 - a. EV3RTインストールディレクトリの **sdcard** ディレクトリにある **uImage** を探します
これがアプリケーションローダのファイルです
 - b. SDカードをPCに接続（挿入）します
 - c. **uImage** を、SDカードのルートディレクトリにコピーします
 - d. SDカードをPCと切断（拔出）します
2. EV3本体をEV3RTを使って起動します
 - a. SDカードをEV3本体に差し込んで、EV3本体の電源を入れます
 - b. EV3RTのアプリケーションローダの画面が表示されたら起動は成功です

USBを使ったプログラムの転送・実行手順

1. SDカードをEV3本体に差し込んで、EV3本体の電源を入れます
2. アプリケーションをSDカードに転送します
 - a. USBケーブルでPCとEV3本体を接続します
 - b. PCがSDカードを認識するのを待ちます（USBメモリのような外部ドライブに見えるでしょう）
 - c. ビルドしたアプリケーション（通常は **app** というファイル名です）を **/ev3rt/apps** ディレクトリにコピーします
 - d. コピーが終わったら、SDカードとPCの接続を解除します

e. 解除を確認したら、EV3本体側のUSBケーブルを取り外します

3. アプリケーションを動かします

a. アプリケーションローダの画面に「Load App」が表示されたら中央ボタンでローダのメニューに切替えます

b. ロード元デバイスのリストが表示されたら、上下ボタンで「SD Card」に移動し、中央ボタンで選択します

c. 転送したファイルリストが表示されたら、 **app**（や自分が変えたファイル名）を中央ボタンで選択します

d. アプリケーションが起動します

本資料について

資料名：要素技術とモデルを開発に使おう（技術資料）

作成者：© 2016 by ETロボコン実行委員会

この文書は、技術教育「要素技術とモデルを開発に使おう」に使用するETロボコン公式トレーニングのスライドです。

2.0, 2016-05-21 12:56:41, 2016年用

2.0

Last updated 2016-05-21 12:54:57 JST