

次世代ロボット知能化技術開発プロジェクト
移動知能(社会・生活分野)の研究開発

モジュール仕様書

基本台車モジュール
+
ゲームパッドモジュール

セグウェイジャパン株式会社

2010/10/8

改版履歷

[illegible]

基本台車モジュール システム・動作 仕様

システム概要

速度指示に従って移動する台車向けモジュール。SWG 共通化検討中のデータ型である速度指示値 IIS::TimedVelocity2D を受け取り、台車制御の結果の速度 IIS::TimedVelocity2D と結果の位置 IIS::TimedPose2D を出力する台車モジュール。共通の IF を採用することにより、共通の IF を備えた他の台車モジュールとの交換性をもつ。共通 I F に関しては、「移動 SWG 共通 IF 案.pdf」を参照ください。※OpenHRP3 向けのモジュールは、各 openhrp3 フォルダの doc を参照ください。

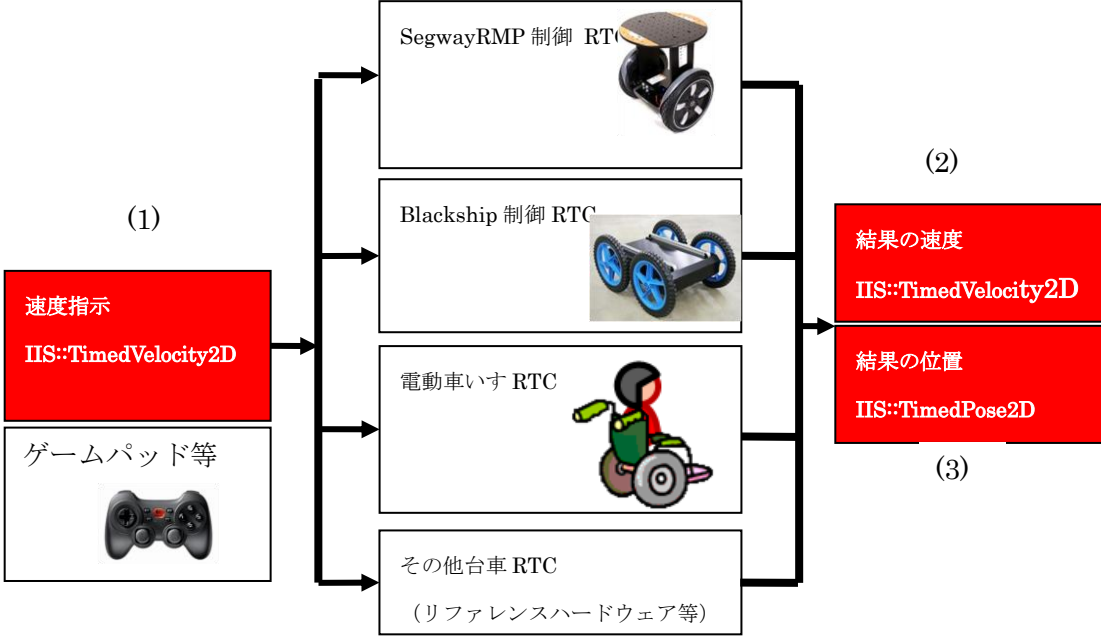


表 コネクタ情報

| NO | データタイプ | 説明 |
|-----|----------------------|---------------------|
| (1) | IIS::TimedVelocity2D | SWG にて共通化検討中の速度指示値 |
| (2) | IIS::TimedVelocity2D | SWG にて共通化検討中の速度結果出力 |
| (3) | IIS::TimedPose2D | SWG にて共通化検討中の位置結果出力 |

表 モジュール一覧

| NO | 名称 | 種別 | 備考 |
|----|------------------|-----|----|
| ① | SegwayRMP 制御 RTC | RTC | |
| ② | Blackship 制御 RTC | RTC | |
| ③ | ゲームパッド RTC | RTC | |

① SegwayRMP 制御 RTC

概要

SegwayRMP 向けの台車モジュール。

速度指示値に対する台車ハードウェアへの速度指示や、制御結果の速度の出力、台車固有のステータスの出力を行う。

動作説明

USB にて SegwayRMP と制御 PC を接続し、SegwayRMP ハードウェアを制御するコマンドを PC 側の当 RTC が生成し、USB 経由で送受信する。



基本情報

| | |
|-----------|--------------------------------------|
| 種別 | RTC |
| 動作 OS | Linux(Ubuntu10.04) / Windows(VC2008) |
| RT ミドルウェア | OpenRTM-aist-1.0.0-RELEASE |
| 開発言語 | C++ |
| 依存ライブラリ | なし |

注意 : Ubuntu10.04 のカーネルには libftd2xx を動作させるために必要な usbcore usbfs がデフォルトでは入っていない、/usr/proc/usb が存在しません。

対処として、/boot/config-2.6.32-21-generic をカーネルソースのルートに .config という名前でコピーして CONFIG_USB_DEVICEFS=y に変更してカーネル再構築をする必要があります。

もしくはリアルタイムカーネルパッケージ 2.6.31-11-rt を導入する手段もあります。

```
> apt-get install linux-rt
```

ポート情報

DataIn Port

| ポート名 | 型 | 説明 |
|---------------|----------------------|------|
| Velocity2DIIS | IIS::TimedVelocity2D | 速度指示 |

DataOut Port

| ポート名 | 型 | 説明 |
|----------------|----------------------|----------------------|
| RobotVel2DIIS | IIS::TimedVelocity2D | 制御結果の速度情報 |
| RobotPose2DIIS | IIS::TimedPose2D | 制御結果の位置情報（内部オドメトリ情報） |
| EncoderL | TimedLong | 左車輪エンコーダ値 |
| EncoderR | TimedLong | 右車輪エンコーダ値 |
| Battery | TimedFloat | バッテリー電圧 |
| UiBattery | TimedFloat | UI バッテリー電圧 |
| PitchAngle | TimedFloat | 本体角度ピッチ |
| PitchRate | TimedFloat | 本体角速度ピッチ |
| RollAngle | TimedFloat | 本体角度ロール |
| RollRate | TimedFloat | 本体角速度ロール |
| YawRate | TimedFloat | 本体角速度ヨー |

Configuration 情報

| 名前 | 型 | 説明 |
|----------------|--------|---------------------------------------|
| KSpeed | int | 入力速度に乘じられる係数 |
| KSpeedA | double | RMP 速度制御の係数 ※ 1 参照 |
| KSpeedB | double | RMP 速度制御の係数 ※ 1 参照 |
| KTurn | double | 入力角速度に乘じられる係数 |
| Kodoxy | double | 内部オドメトリ xy に乘じられる係数 |
| Kodotheta | double | 内部オドメトリ theta に乘じれる係数 |
| wheelwidth | double | 内部オドメトリ計算に用いされるホイールトレッド幅 |
| wheelradiusL | double | 内部オドメトリ計算に用いされる左ホイール半径 |
| wheelradiusR | double | 内部オドメトリ計算に用いされる右ホイール半径 |
| wheelcount1rot | int | 内部オドメトリ計算に用いられるタイヤ 1 回転で生ずるエンコーダカウント値 |
| MaxV | int | 最大速度 |

| | | |
|------|-----|-------|
| MaxW | int | 最大角速度 |
|------|-----|-------|

※1 $V = V * (K_{speedA} * |V| + K_{speedB}) * K_{speed}$

RMP 台車の生成速度が線形でないため、調整係数を用いて線形に近い指令値に変換します。

[環境設定 Linux]

FTDI のライブラリをインストールする必要があります。

RMP のマニュアル「Interface Guide for Segway RMP 2.0.pdf」 page11 に従い、

<http://www.ftdichip.com>

<http://www.ftdichip.com/Drivers/D2XX.htm>

より、FTD245BM 用の libftd2xx の tar ボールを取得、展開し、

libftd2xx.so.****

を PATH の通った場所に置きます。

(/usr/local/lib 等)

さらに、PATH の通ったところにシンボリックリンクを作成しておきます

```
>ln -s /usr/local/lib/libftd2xx.so.0.4.13 libftd2xx.so
```

[環境設定 Windows]

ftd2xx.dll を RmpComp.exe と同じフォルダにおいてください。

操作・使用方法 RMP

・ RMP のセッティング (RMPの詳細はUserGuideを参照してください)

- 仮にゲームパッドでRMPを動作させることを想定します
- PCとゲームパッドを接続します。RMPとPCを接続します。
- E-Stopケーブルがついていることを確認してください。
- RMPを起動します。下図左からボタンを入れていきます。



電源ボタン



モーター起動ボタン

トラクターモードを使用する場合は、そのまま

倒立モードを使用する場合はBalanceModeボタンを押します。

一度BalanceModeにしても、TractorModeボタンを押すとTractorモードになります

OpenHRP3の場合には、常時TractorMode状態で3点接地キャストモデルになります。

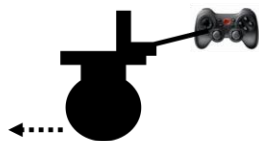
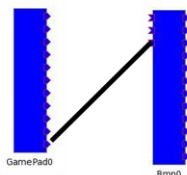


TractorMode



BalanceMode

GamePadのVelocity出力をRmpのVelocity入力に接続し、Activateします。
左右のアナログスティックでRmpを操作することができます。



E-Stopケーブル

ケーブルを抜くとモーターがシャットダウンします。
倒立モードの際は本体が倒れますので注意してください。

UserGuideに注意事項が書いてあります。
必ず読むようにしてください。

倒立モードにする場合は、
本体がまっすぐ立っていることを確認してください。
本体を支える際は、本体の横に立つようにしてください。

ソフトウェアを試す際や変更を加えた際、その他の確認の際には、
必ず最初にRMPをトラクターモード(倒立しないモード)にて
台座を設けるなどをして、タイヤを浮かして空転できる状態で十分な確認をしてください。
再利用センターではオレンジ色のリフトがありますのでそれを使ってください。

② Blackship 制御 RTC

概要

Blackship 向けの台車モジュール。

速度指示値に対する台車ハードウェアへの速度指示や、制御結果の速度の出力、台車固有のステータスの出力を行う。

動作説明

シリアルケーブルにて Blackship と制御 PC を接続し、Blackship ハードウェアを制御するコマンドを PC 側の当 RTC が生成し、USB 経由で送受信する。



基本情報

| | |
|-----------|--------------------------------------|
| 種別 | RTC |
| 動作 OS | Linux(Ubuntu10.04) / Windows(VC2008) |
| RT ミドルウェア | OpenRTM-aist-1.0.0-RELEASE |
| 開発言語 | C++ |
| 依存ライブラリ | なし |

ポート情報

DataIn Port

| ポート名 | 型 | 説明 |
|---------------|----------------------|------|
| Velocity2DIIS | IIS::TimedVelocity2D | 速度指示 |

DataOut Port

| ポート名 | 型 | 説明 |
|----------------|----------------------|----------------------|
| RobotVel2DIIS | IIS::TimedVelocity2D | 制御結果の速度情報 |
| RobotPose2DIIS | IIS::TimedPose2D | 制御結果の位置情報（内部オドメトリ情報） |
| EncoderL | TimedLong | 左車輪エンコーダ値 |
| EncoderR | TimedLong | 右車輪エンコーダ値 |

Configuration 情報

| 名前 | 型 | 説明 |
|----------------|--------|---------------------------------------|
| KSpeed | int | 入力速度に乘じられる係数 |
| KTurn | double | 入力角速度に乘じられる係数 |
| Kodoxy | double | 内部オドメトリ xy に乘じられる係数 |
| Kodotheta | double | 内部オドメトリ theta に乘じれる係数 |
| wheelwidth | double | 内部オドメトリ計算に用いされるホイールトレッド幅 |
| wheelradiusL | double | 内部オドメトリ計算に用いされる左ホイール半径 |
| wheelradiusR | double | 内部オドメトリ計算に用いされる右ホイール半径 |
| wheelcount1rot | int | 内部オドメトリ計算に用いられるタイヤ 1 回転で生ずるエンコーダカウント値 |
| MaxV | int | 最大速度 |
| MaxW | int | 最大角速度 |

[環境設定 Linux]

configuration の str:port に Blackship の接続ポートを指定します。

シリアルポート接続の場合には

`/dev/ttyCOM*`

シリアル USB 変換ケーブルで接続している場合は

`/dev/ttyUSB*`

になります。

[環境設定 Windows]

configuration の str:port に Blackship の接続ポートを指定します。

`COM*`

になります。

COM 番号は、

コントロールパネル → システム → デバイスマネージャ → ポート
にて確認できます。

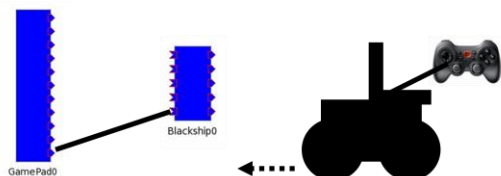
操作・使用方法 Blackship

• Blackshipのセッティング

- 仮にゲームパッドでRMPを動作させることを想定します
- PCとゲームパッドを接続します。BlackshipとPCを接続します。
- 主電源をいれてBlackshipを起動します。
- OpenHRP3の場合には、常時主電源が入っている状態になります。



GamePadのVelocity出力をBlackshipのVelocity入力に接続し、Activateします。
左右のアナログスティックでBlackshipを操作することができます。



③ ゲームパッド RTC

概要

PC 用ゲームパッド向けのデバイスモジュール

動作説明

PC 用ゲームパッド向けデバイスモジュール。ゲームパッドの入力値を出力します。

基本情報

| | |
|-----------|--------------------------------------|
| 種別 | RTC |
| 動作 OS | Linux(Ubuntu10.04) / Windows(VC2008) |
| RT ミドルウェア | OpenRTM-aist-1.0.0-RELEASE |
| 開発言語 | C++ |
| 依存ライブラリ | なし |

ポート情報

DataOut Port

| ポート名 | 型 | 説明 |
|---------------|----------------------|--------------------|
| Button | TimedULong | ボタン状態 (各 bit) |
| StickLX | TimedFloat | 左アナログスティックの x 方向値 |
| StickLY | TimedFloat | 左アナログスティックの y 方向値 |
| StickRX | TimedFloat | 右アナログスティックの x 方向値 |
| StickRY | TimedFloat | 右アナログスティックの y 方向値 |
| StickLXd | TimedDouble | 左アナログスティックの x 方向値 |
| StickLYd | TimedDouble | 左アナログスティックの y 方向値 |
| StickRXd | TimedDouble | 右アナログスティックの x 方向値 |
| StickRYd | TimedDouble | 右アナログスティックの y 方向値 |
| Velocity2DIIS | IIS::TimedVelocity2D | 左右アナログスティックからの速度生成 |



基本台車モジュール、ゲームパッドモジュール 使用手順

環境準備

1) OpenRTM-aist-1.0.0-RELEASE (C++版 UNIX) のインストール

【マニュアルの場所】 OpenRTM-aist Official Web Site ⇒ ドキュメント ⇒ インストール
<http://www.openrtm.org/openrtm/ja/node/999>

【インストール方法】 マニュアルに従い、インストールを行う。

2) RTSystemEditor のインストール

【マニュアルの場所】 OpenRTM-aist Official Web Site ⇒ ドキュメント ⇒ ツール ⇒ RT System Editor-1.0.0
<http://www.openrtm.org/openrtm/ja/content/rtsystemeditor-1>

【インストール方法】 マニュアルに従い、UNIX 用インストールパッケージのインストールを行う。

ハードウェア準備

各種ハードウェアの電源を入れ、準備可能にする。

起動手順

1) OpenRTM-aist RT / System Editor の起動

【マニュアルの場所】 OpenRTM-aist Official Web Site ⇒ ドキュメント ⇒ ツール ⇒ RT System Editor-1.0.0
<http://www.openrtm.org/openrtm/ja/content/rtsystemeditor-1>

【起動方法】 マニュアルに従い、RT System Editor の起動を行う。

2) Comp ファイルの実行

【Comp ファイルの場所】 各モジュールの src ディレクトリの中

【実行方法】 各 Comp ファイルを実行。

3) モジュールの接続

【接続方法】 以下の図のようにゲームパッドモジュールを接続する。IIS::TimedVelocity2D 同士をつなぐのがよい。

デバイスポートの設定等の各コンフィグレーション設定を行う。

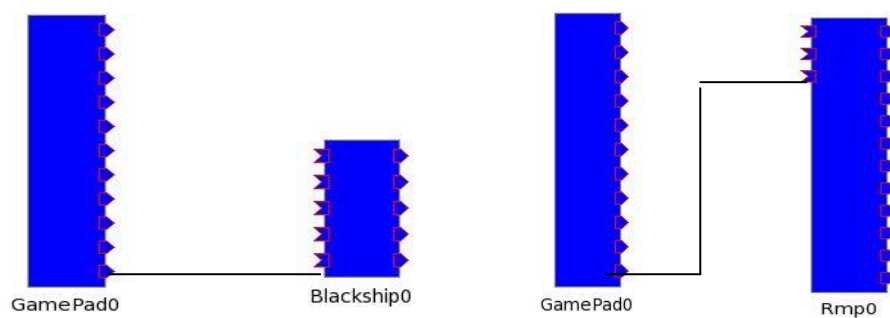


図 基本台車モジュール接続図

4) モジュールの起動

RT-SystemEditor にて、各モジュールを Activate します。

使用方法

ゲームパッドのアナログスティックを倒すことで出力される速度指示に従い、台車ロボットが移動します。