

手順5 ハードウェアジョイスティック（Ministick）で Kobuki を操  
作する

## 目次

1	Raspberry Pi に Ministick Sensor を接続する。 .....	2
2	PC に接続した Raspberry Pi に Ministick コンポーネントをダウンロード.....	3
3	PC に接続した Raspberry Pi で NameServer と Ministick コンポーネントを起動 .....	4
4	Kobuki に接続した Raspberry Pi で NameServer と Kobuki 用コンポーネントを起動 .....	5
5	PC で RTSystemEditorRCP を起動し各コンポーネントを接続し Active にする。 .....	6
6	ハードウェアジョイスティックで Kobuki を操作できることを確認 .....	9

※ 文中の「x.y」や「x.y.z」の表記は使用環境の OpenRTM-aist のバージョンに読み替えてください。

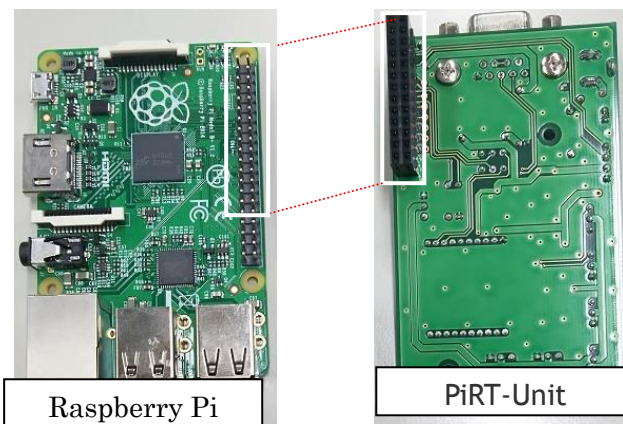
当ドキュメントは下記ページを参考にしています。

・ PiRT-Unit を利用した IO プログラミング

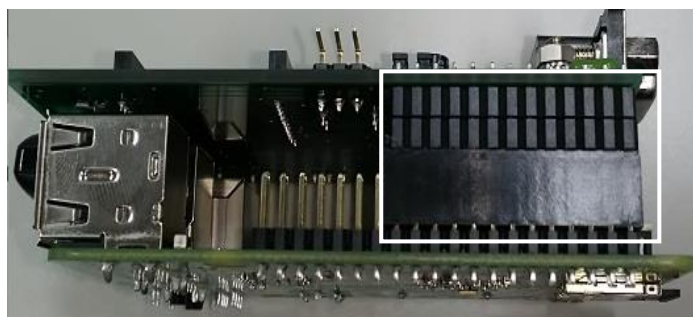
<http://www.openrtm.org/openrtm/ja/content/pirt-unit-programming> (2016/1/20 アクセス)

### 1 Raspberry Pi に Ministick Sensor を接続する。

Raspberry Pi に PiRT-Unit を接続します。上を合わせて線で囲まれた部分同士を接続します。



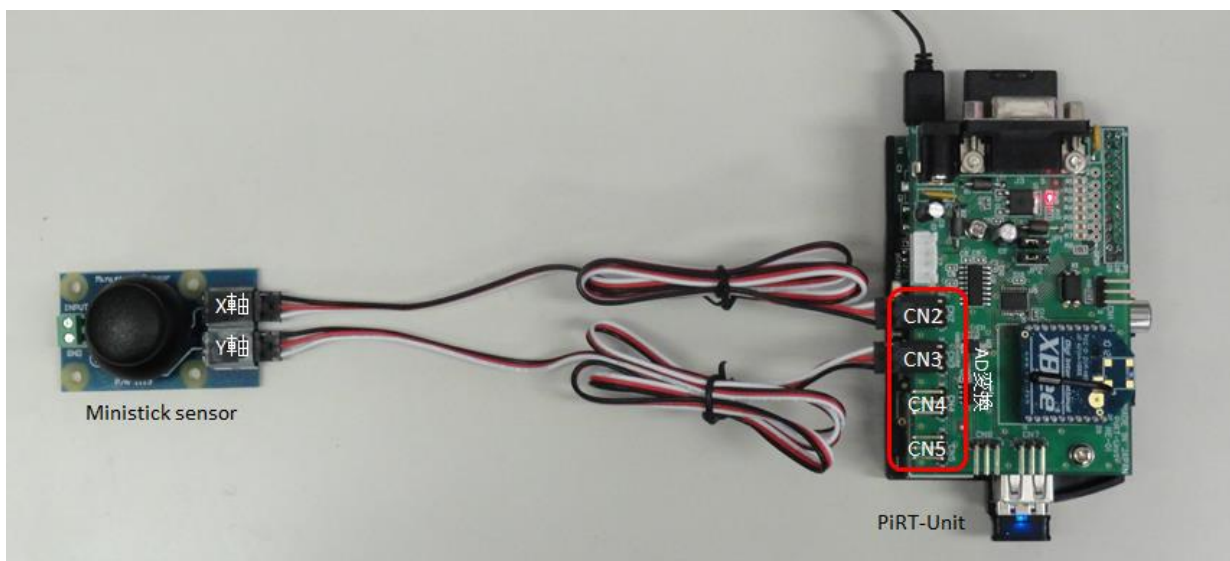
下図が接続後、横から見た状態になります。



Ministick Sensor と Raspberry Pi に接続された PiRT-Unit を以下のように接続します。

※Ministick Sensor は十分な数が準備できない可能性がありますのでテーブル内で順番に確認してください。

- X 軸方向: CN2
- Y 軸方向: CN3



## 2 PC に接続した Raspberry Pi に Ministick コンポーネントをダウンロード

PiRT-Unit の使用のためのパッケージをサイトからダウンロードし、セットアップするのに以下のように実行します。

```
$ wget http://svn.openrtm.org/Embedded/trunk/RaspberryPi/tools/rpi.sh
$ chmod 755 rpi.sh
$ sudo ./rpi.sh [Raspberry Piのホスト名] --type rtunit_only
$ sudo ./rpi.sh [Raspberry Piのホスト名] --type rtunit_examples
```

wget : ファイルをダウンロードする。

chmod : ファイルやディレクトリのパーミッションを変更する。

sudo : 指定したユーザーでコマンドを実行する。デフォルトでは root 権限で実行する。

上記のコマンドの実行により、PiRT-Unit を利用するために行っている設定、パッケージのインストール、サンプルのインストールが自動で行われます。完了まで時間がかかりますのでお待ちください。

ネットワーク環境により、上記のインストールが正常に完了しない場合があります。

以下のコマンドを実行してください。

```
$ ls
```

ls : ファイル・ディレクトリ情報を表示する。

実行結果に `py-spidev` というディレクトリが存在しない場合はインストールが正常に完了していませんので下記のコマンドを実行してください。

```
$ git clone https://github.com/doceme/py-spidev
$ cd py-spidev
$ chmod 755 setup.py
$ sudo ./setup.py install
$ cd
$ sudo chown -R pi py-spidev
```

git clone : 既存のリポジトリの複製を作成する。

cd : カレントディレクトリを変更する。

chown : ファイルやディレクトリの所有者を変更する。

### 3 PC に接続した Raspberry Pi で NameServer と Ministick コンポーネントを起動

Raspberry Pi に TeraTerm など接続後、コンポーネントを起動します。ネーミングサービスを起動してから Ministick コンポーネントを起動します。

```
$ rtm-naming  
$ cd PiRT-Unit  
$ python Ministick.py
```

rtm-naming : ネーミングサービスの起動

rtm-naming の際に[y/n]を聞かれた場合[y]を選んでください。

4 Kobuki に接続した Raspberry Pi で NameServer と Kobuki 用コンポーネントを起動  
デバイスファイル /dev/ttyUSB0 へのアクセスには root 権限が必要ですので、sudo を使って起動して  
います。

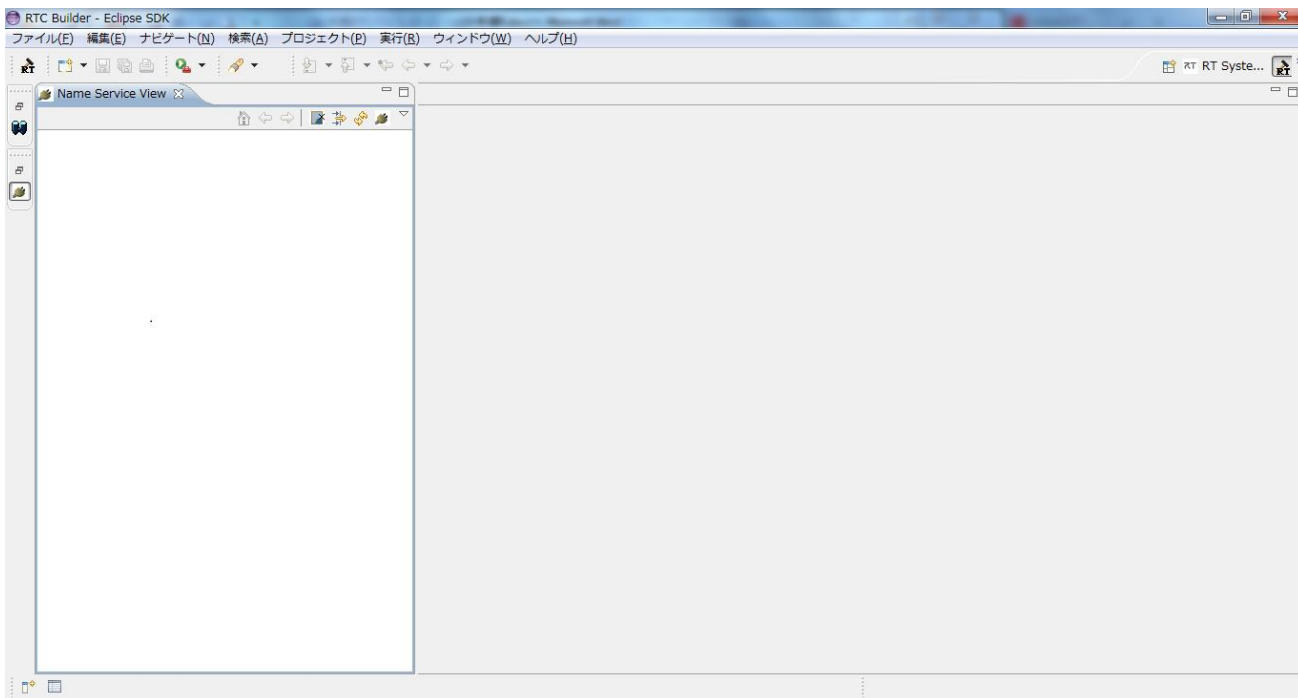
※Kobuki は 1 テーブルに 1 台ですので順番に動作確認を行ってください。

```
$ rtm-naming  
$ sudo /usr/lib/openrtm-1.1/rtc/KobukiAISTComp
```

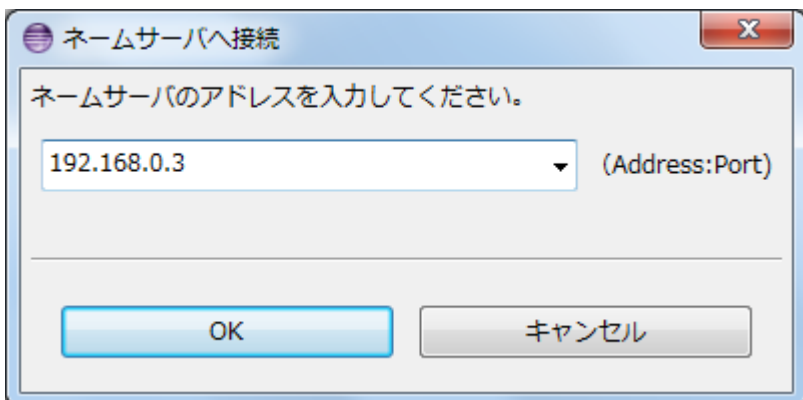
## 5 PC で RTSystemEditorRCP を起動し各コンポーネントを接続し Active にする。

OpenRTP を起動していない場合、スタートメニューから 「OpenRTM-aist x.y」 → 「tools」 の下の OpenRTP から起動します。

RTSystemEditor の左側の NameService View のコンセントアイコンをクリックし、ネームサーバに接続します。

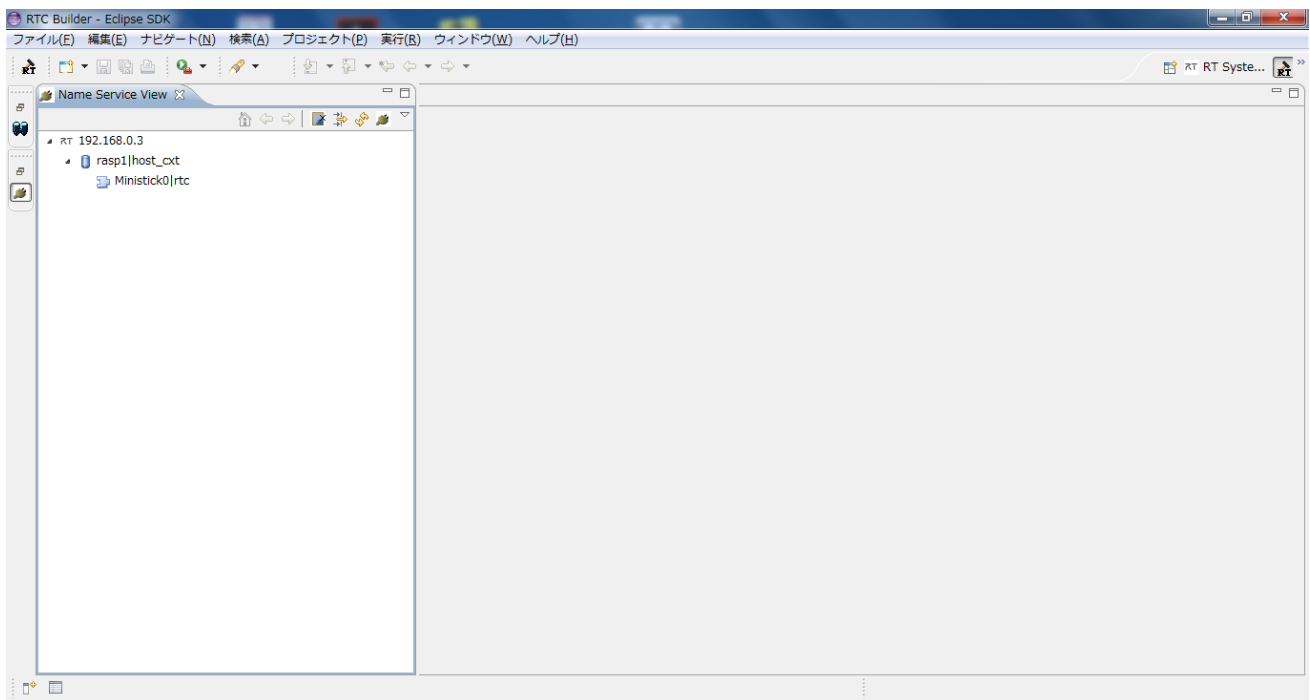


まず、PC に接続した Raspberry Pi に接続します。接続ダイアログに PC に接続した Raspberry Pi の IP アドレス、または、ホスト名を入力します。

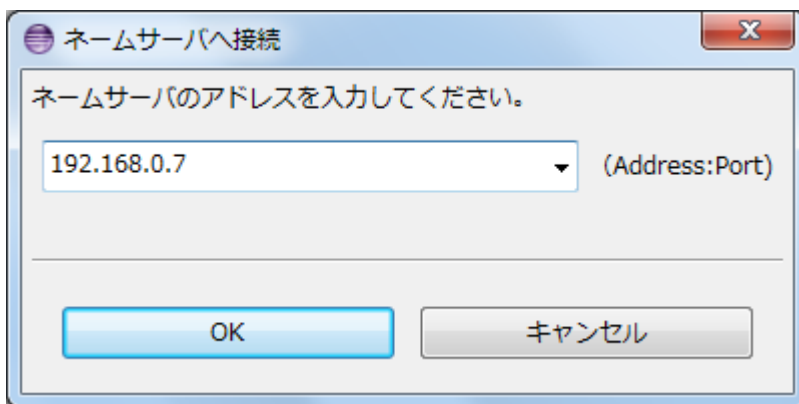


手順5 ハードウェアジョイスティック (Ministick) で Kobuki を操作する

ネームサービスビューに PC に接続した Raspberry Pi が表示されます。



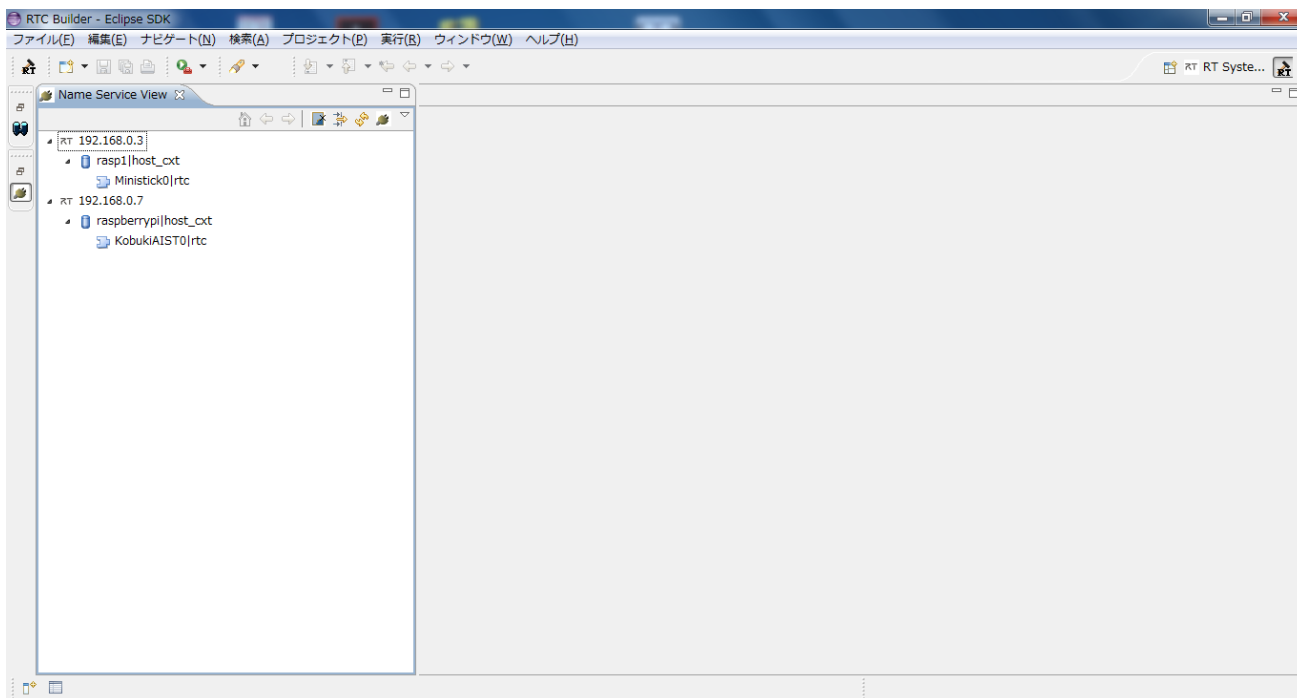
次に、Kobuki に接続した Raspberry Pi に接続します。接続ダイアログに Kobuki に接続した Raspberry Pi の IP アドレス、または、ホスト名を入力します。



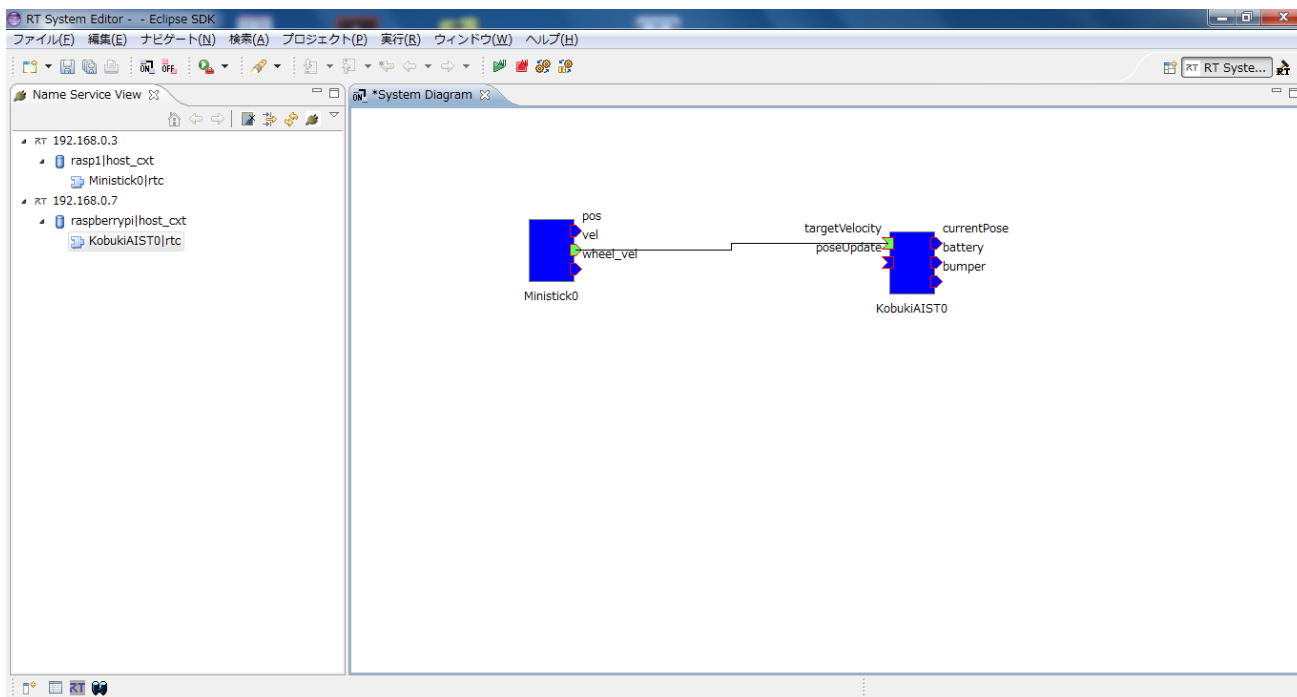


手順5 ハードウェアジョイスティック (Ministick) で Kobuki を操作する

ネームサービスビューに Kobuki に接続した Raspberry Pi が表示されます。



RTSystemEditor のメニューバーの online エディタアイコン (ON と書かれたアイコン) をクリックし、SystemEditor を開きます。NameService View から Ministick0 と KobukiAIST0 をそれぞれ SystemEditor 上にドラッグアンドドロップし、Ministick0 の 2 次元速度ベクトルと KobukiAIST0 の 2 次元速度ベクトルのデータポートを接続します。



## 6 ハードウェアジョイスティックで Kobuki を操作できることを確認

RTSystemEditor のメニューバーの緑の再生ボタンをクリックすると、Ministick0、KobukiAIST0 というコンポーネントがアクティベートされ、ハードウェアジョイスティックで Kobuki が操作できることを確認してください。

操作できることの確認が終了しましたら、OpenRTP の画面の赤の停止ボタンをクリックしてディアクティベートし、Kobuki に接続している Raspberry Pi の TeraTerm の画面で Ctrl キー + C キーを押して Kobuki 用コンポーネントを終了してください。

PC に接続している Raspberry Pi の TeraTerm の画面で Ctrl キー + C キーを押して Ministick コンポーネントを終了します。