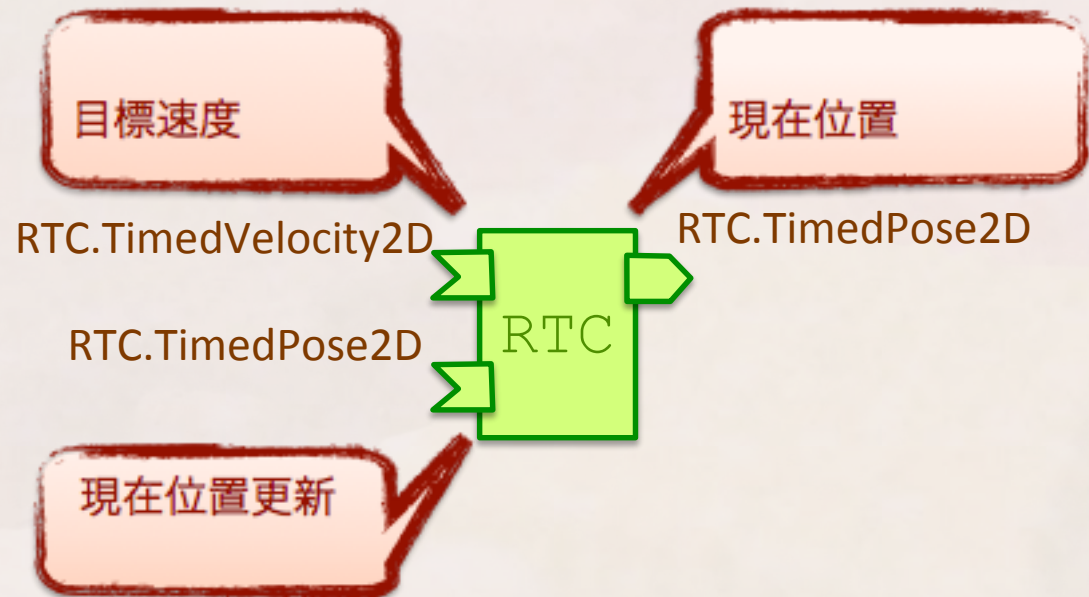


# 有用なRTCの紹介

菅 佑樹(早大, SSR)

# 移動ロボット

- Kobuki
- Pioneer3 DX



- 目標速度を与えて現在位置を返す.
- 現在位置は、駆動モータに着いたエンコーダの値の積算なので、スリップなどの影響で狂いやすい
- 現在位置を外部のRTCから更新できる

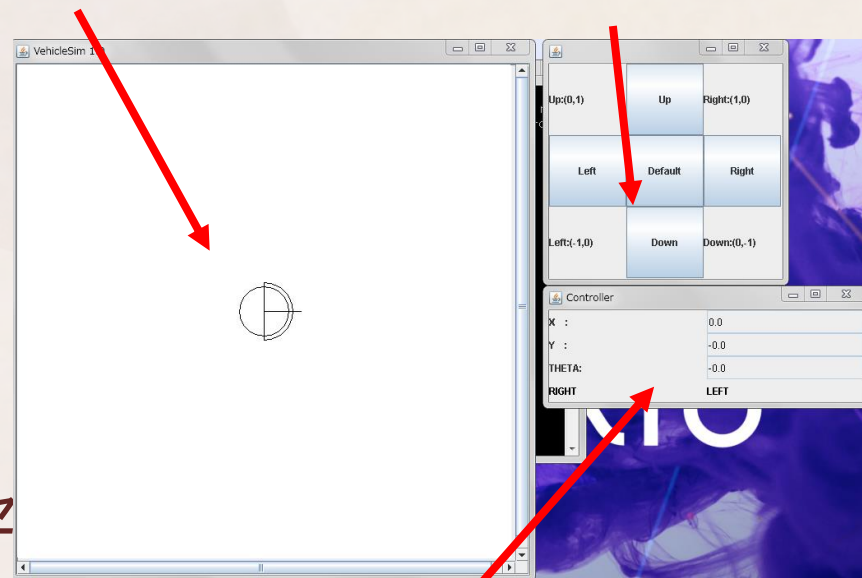
# OpenRTM-aist学習用台車シミュレータ

<http://ysuga.net/?p=133>

- Loader.batを実行
  - シミュレータ
  - 仮想ジョイスティック
  - コントローラ
- すべて接続してACTIVATE
- ジョイスティックで操作
- コントローラGUIで位置や接触セ

シミュレータ

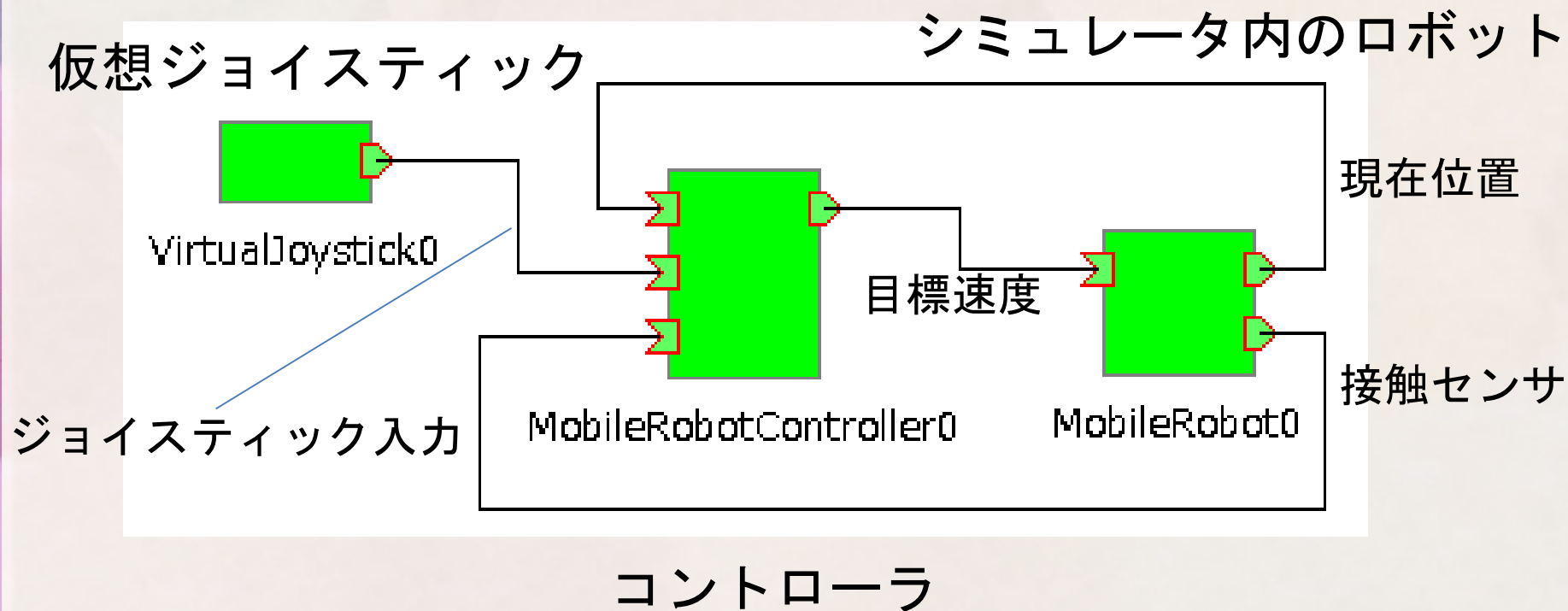
仮想ジョイスティック



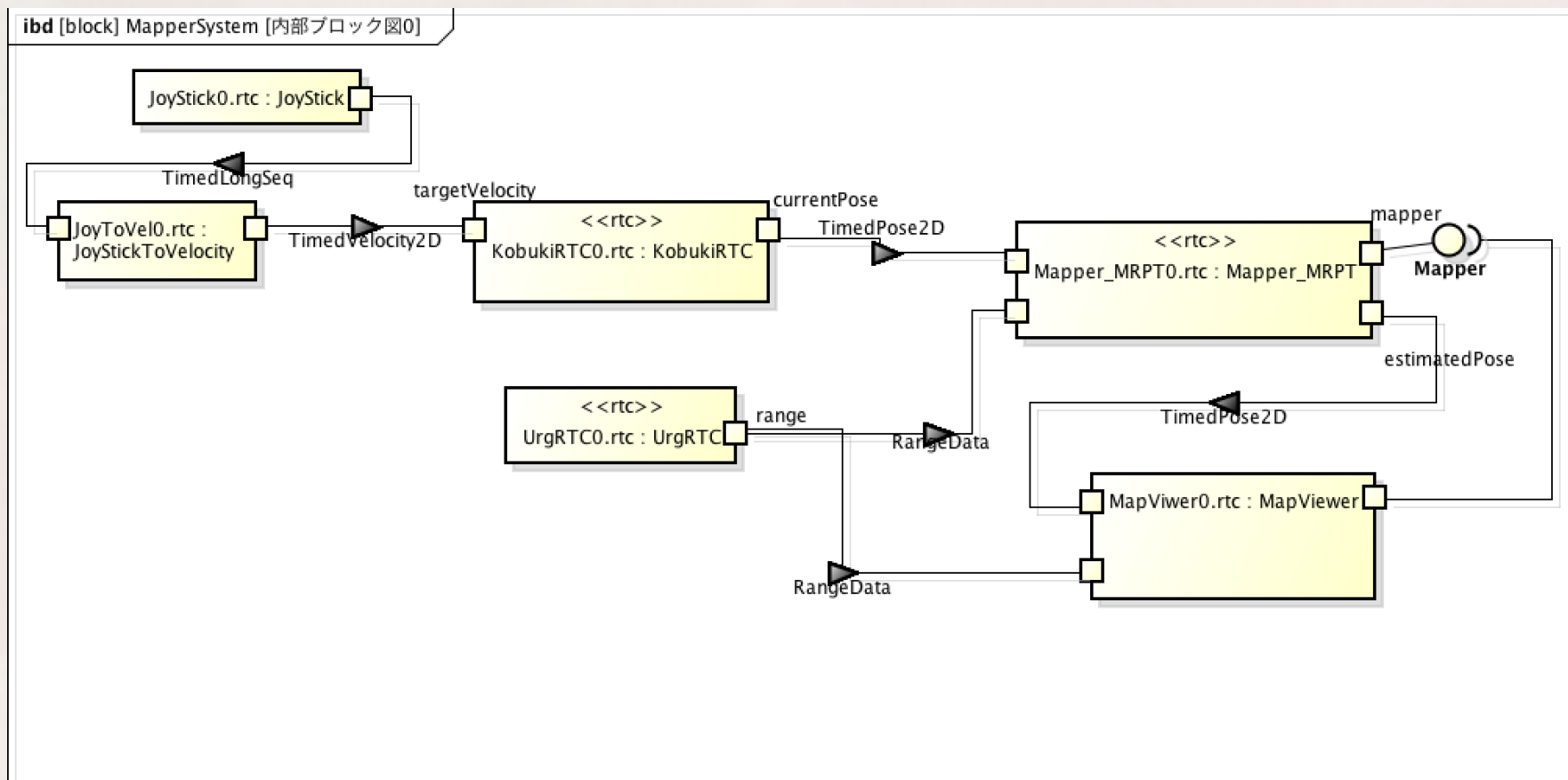
コントローラGUI

# 台車シミュレータの中身

- RT System Editorで表示

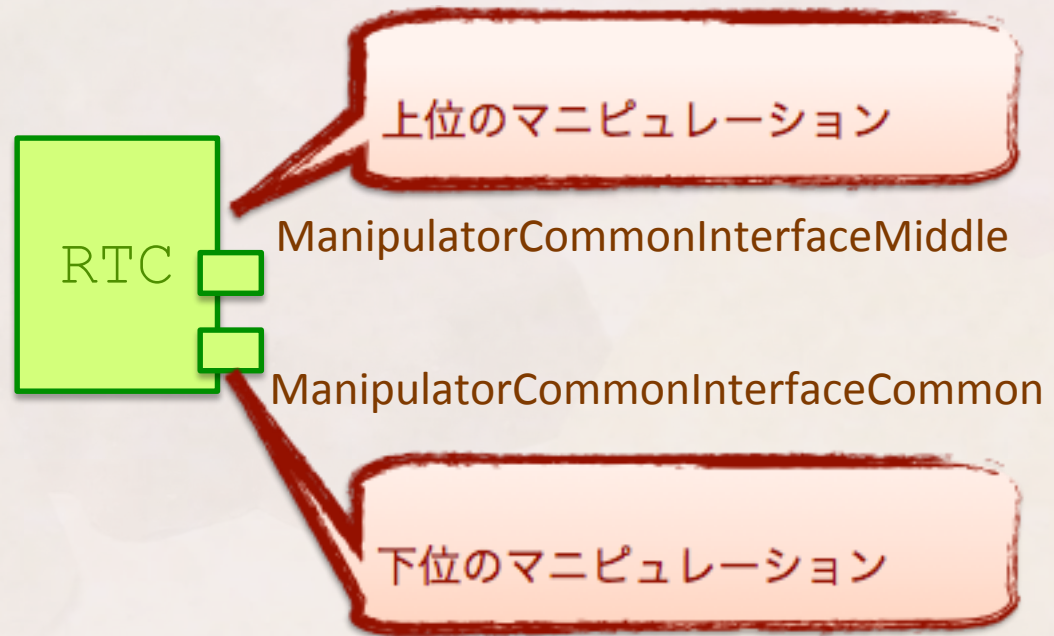


# 応用例: Mapperシステム



# ロボットアーム

- OROCHI



- サービスポートで動作
- 手先の目標位置を指定するとそちらに直線で動く

# IDLを見る

```
interface ManipulatorCommonInterface_Common {
    RTC::RETURN_ID clearAlarms();
    RTC::RETURN_ID getActiveAlarm(out RTC::AlarmSeq alarms);
    RTC::RETURN_ID getFeedbackPosJoint(out RTC::JointPos pos);
    RTC::RETURN_ID getManipInfo(out RTC::ManipInfo manipInfo);
    RTC::RETURN_ID getSoftLimitJoint(out RTC::LimitSeq softLimit);
    RTC::RETURN_ID getState(out RTC::ULONG state);
    RTC::RETURN_ID servoOFF();
    RTC::RETURN_ID servoON();
    RTC::RETURN_ID setSoftLimitJoint(in RTC::LimitSeq softLimit);
};
```

```
interface ManipulatorCommonInterface_Middle
{
    RTC::RETURN_ID closeGripper();
    RTC::RETURN_ID getBaseOffset(out RTC::HgMatrix offset);
    RTC::RETURN_ID getFeedbackPosCartesian(out RTC::CarPosWithElbow pos);
    RTC::RETURN_ID getMaxSpeedCartesian(out RTC::CartesianSpeed speed);
    RTC::RETURN_ID getMaxSpeedJoint(out RTC::DoubleSeq speed);
    RTC::RETURN_ID moveLinearCartesianAbs(in RTC::CarPosWithElbow carPoint);
    RTC::RETURN_ID moveLinearCartesianRel(in RTC::CarPosWithElbow carPoint);
    RTC::RETURN_ID movePTPJointAbs(in RTC::JointPos jointPoints);
    RTC::RETURN_ID movePTPJointRel(in RTC::JointPos jointPoints);
    RTC::RETURN_ID openGripper();
    RTC::RETURN_ID setMaxSpeedCartesian(in RTC::CartesianSpeed speed);
    RTC::RETURN_ID setMaxSpeedJoint(in RTC::DoubleSeq speed);
    RTC::RETURN_ID setMinAccelTimeCartesian(in double acclaim);
    RTC::RETURN_ID setMinAccelTimeJoint(in double acclTime);
    RTC::RETURN_ID setSoftLimitCartesian(in RTC::LimitValue xLimit,
    RTC::RETURN_ID setSpeedCartesian(in RTC::ULONG spdRatio);
    RTC::RETURN_ID setSpeedJoint(in RTC::ULONG spdRatio);

    ....
};
```



# 組み込み

- Raspberry-Pi
  - Linux
  - SSHでログインして使う

# ご清聴ありがとうございました

菅 佑樹 (ysuga@ysuga.net)

株式会社SUGAR SWEET ROBOTICS代表取締役

早稲田大学ヒューマノイド研究所招聘研究員

芝浦工業大学SIT研究所客員研究員

東京大学情報理工学系研究科特任研究員(JSK-START)プロジェクト

<http://vsuga.net>

<http://sugarsweetrobotics.com>

<http://opensource-robotics.tokyo.jp>

尾形 哲也 (ogata@waseda.jp)

早稲田大学基幹理工学部表現工学科 教授

科学技術振興機構さきがけ研究「情報と人」領域研究員

<http://ogata-lab.jp>

本研究の一部はJSTさきがけ領域「情報環境と人」及び文科省科研費新学術領域研究「構成論的発達科学」(No. 24119003)の助成を受けています。ここに感謝の意を表します。