**自動走行ロボットの製作**

プロジェクト演習4

情報工学科　正元淳也　石橋尚之　馬場竣平　佐々木祥二

[年]

内容

[プロジェクト概要 2](#_Toc72232620)

[開発目的 2](#_Toc72232621)

[過去作品 2](#_Toc72232622)

[システム構成 3](#_Toc72232623)

[自動走行ロボット(ローバー) 3](#_Toc72232624)

[基地局 3](#_Toc72232625)

[RTK2GO 4](#_Toc72232626)

[クライアントアプリ 4](#_Toc72232627)

[仕様技術 5](#_Toc72232628)

[realSense D435i 5](#_Toc72232629)

[Tofセンサー 5](#_Toc72232630)

[バンパー（タッチセンサー） 5](#_Toc72232631)

[RTK法とは 6](#_Toc72232632)

[みちびきを利用した単独測位 6](#_Toc72232633)

[RTK法 7](#_Toc72232634)

[サスペンション 7](#_Toc72232635)

[スケジュール 7](#_Toc72232636)

[参考 8](#_Toc72232637)

# プロジェクト概要

GoogleMAPで指定した場所までGNSSの自己位置をもとに自立走行するロボットの製作をする。

# 開発目的

GPSを利用して自立走行を実現することで、みちびきによる高精度測位技術の習得・ロボットの自立走行に関する知見を得ること。

上記のものを製作しようとすると、LEVEL5になり現実的ではないので、行動や人通りが多いところでの運用は想定しない。

そして、公園等の簡単な障害物がある平地での自立走行を目指す。

# 過去作品

2019年度にRTK測位を用いた自走ロボットプロジェクトあり

基地局と移動局を使ったGPSシステムであったが、RTKLIBなどを利用した複雑なプログラムではないと感じた。

久保田先生からいただいた「プロジェクト演習4プログラム.zip/移動局/GPS\_Rover/gps.py」内の関数ENU\_CONV()から判断した。

# システム構成

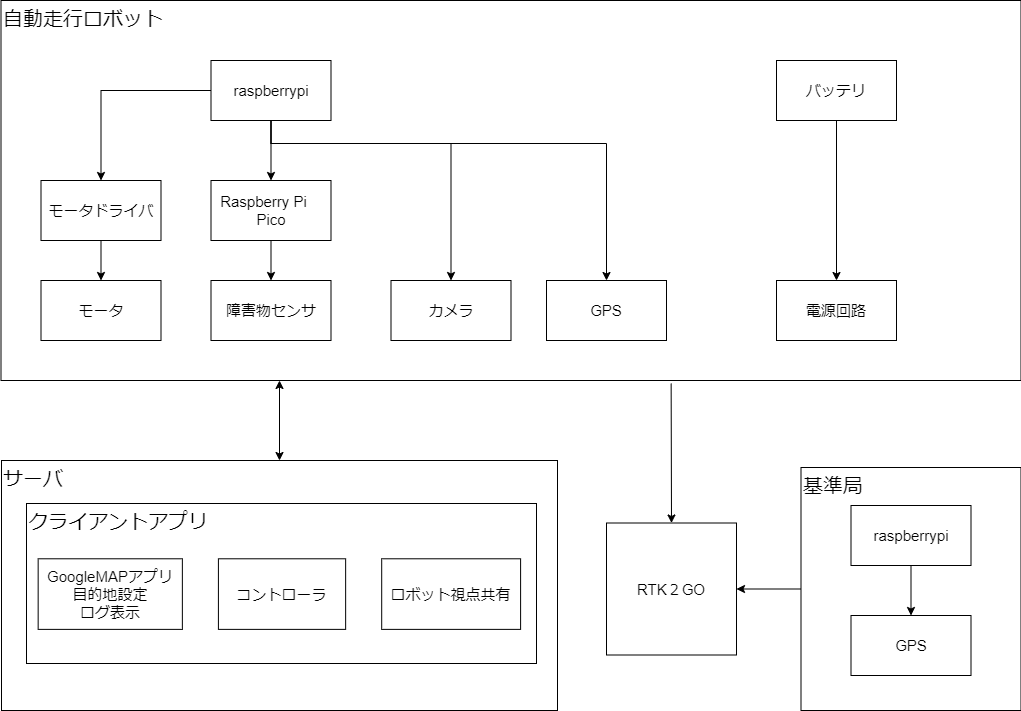


図1

システムは上記のように4つの要素によって製作されている。

## 自動走行ロボット(ローバー)

ロボットに実装予定の機能としては下記の通りである。

* 4輪走行体(駆動輪は2つにして制御を容易にする)
* 目標地点に移動
* RTK測位で自己位置推定
* 障害物回避(intel Real Senseやtofセンサーを利用して障害物を認識する)
* 防水・防塵設計(スタンチューブなどを利用するほどの防水性は必要ない。)
* サスペンションによる車体の安定化
* バンパー（タッチセンサーによって障害物に接触した状態から進み続けるなどといったことがないようにする。）

## 基地局

基準局の位置情報をもとに位置補正情報を生成する。

先輩方が設置した洛北校屋上の物を改良する。

基地局の正確な座標は高精度単独測位(PPP方式)をRTKLIBによって行うべきだと感じたが、詳細不明である。

## RTK2GO

基準局の測位情報配信サービスである。

駅前校の基準局は対応しているが、洛北校の基準局は非対応である。

そこで、洛北校の基準局も対応させることで、学校wifiが届かない公園などでもロボットが利用できるようにする。

参考6の京都駅前校の基地局についてのリンクがある。



上記の図が駅前校基地局の範囲だが、洛北校も10km内に入っている。しかし、洛北基地局を設置する理由は技術力向上が目的なので設置して、洛北基地局を利用したときと駅前校基地局を利用したときを比較したい。

## クライアントアプリ

クライアントアプリで実装したい機能は下記の通りである。

* 目標位置設定
* 視点共有
* マニュアル操作
* ロボット走行ログ
* MONACA(HTML・js)

MONACAを利用した場合、スマホアプリになりWindowsなどから操作が難しくなるので、どのように行うかは判断に迷っている。

# 仕様技術

## realSense D435i

深度の測定が28cm～2mなので、約30cm以下の障害物に対する対策が必要。

慣性測定センサーが付属しているので、ロボットの傾きや移動距離などを補正することができる。

深度はrs2\_depth\_frame\_get\_distance(frame,横座標,縦座標)より得られる。([参考プログラム](https://github.com/IntelRealSense/librealsense/blob/master/examples/C/distance/rs-distance.c)109行目より)

慣性測定システムの理想的な動作などを含めた[サンプル](https://github.com/IntelRealSense/librealsense/tree/master/examples/motion)も公開されている。

また、震度情報から、３D図への変換は[ここ](https://demura.net/robot/17459.html)が役に立つのではないかと感じた。

## Tofセンサー

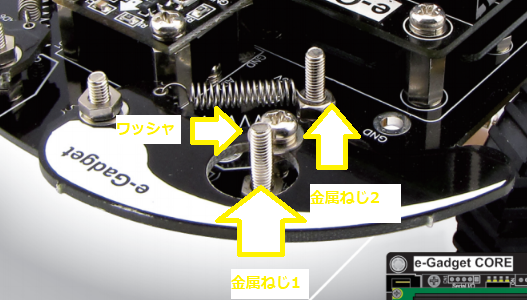
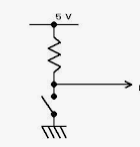
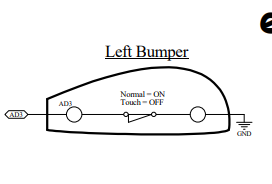
ペットボトルなどの透明なものの認識は苦手だが、曲面や斜面などの認識が超音波センサーなどと違って詳細に行うことができる。

[このセンサー](https://akizukidenshi.com/catalog/g/gM-12590/)は3cm~200cmが測定範囲である。

realSenseの範囲外の障害物を安価に認識することを期待する。

## バンパー（タッチセンサー）

realSenseやtofセンサーによって見落とした障害物に機体が接触したときに、障害物を回避するために利用する。

接続順としては下記の通り。

・GPIO\_a-金属ねじ1-ワッシャ-アルミホイル-ばね-金属ねじ2-GND

GPIO\_aは5Vに図1のように接続するとバンパーのワッシャと金属ねじ1が触れているときにGND。触れていないときに５V。

アルミホイルは「ばね」と「ワッシャ」の間に電気を通せば、何でもよい。

## RTK法とは

“”“「RTK」とは『リアルタイムキネマティック』の略で、地上に設置した「基準局」からの位置情報データによって、高い精度の測位を実現する技術のことです。「RTK」は、一般的に「RTK-GNSS」という表記がされています。この「GNSS」とは「汎地球測位航法衛星システム」のことで、GPSなど、衛星を用いた測位システムの総称のことです。

通常、GPSのみの場合、位置情報データは2メートル前後の誤差となりますが、このRTKを組み合わせることで、数センチ内の誤差に抑えることが可能になります。”””　…　参考1

また、基地局と移動局の距離は10km以下である必要がある。

トランジスタ技術2018年1月号に利用法の概要が記載されており、久保田先生からいただいた。

また、トランジスタ技術2019年2月号、10月号などにも情報が記載されている模様。

## みちびきを利用した単独測位

参考４のGPSセンサーを車の助手席に乗せてデータを集めたところ「NmeaLog\_20210516.nma」というファイルができたので、参考2に付属するpos2kmlを利用してgoogle earthで利用できるkmlファイルに変換すると図2のように大まかな測定に成功しているが、図3のように細かな測定には失敗している。

また、参考４のセンサーは出力が、NMEA形式なので、参考3、参考2のRTKLIBにおいてRTK測位を行うことができない。



図2 図3

よって、アンテナ付きセンサーを利用するか、RTK法を行う必要がある。

## RTK法

参考3は参考２のGUIをLINUX上で動作するものに変更したもの。

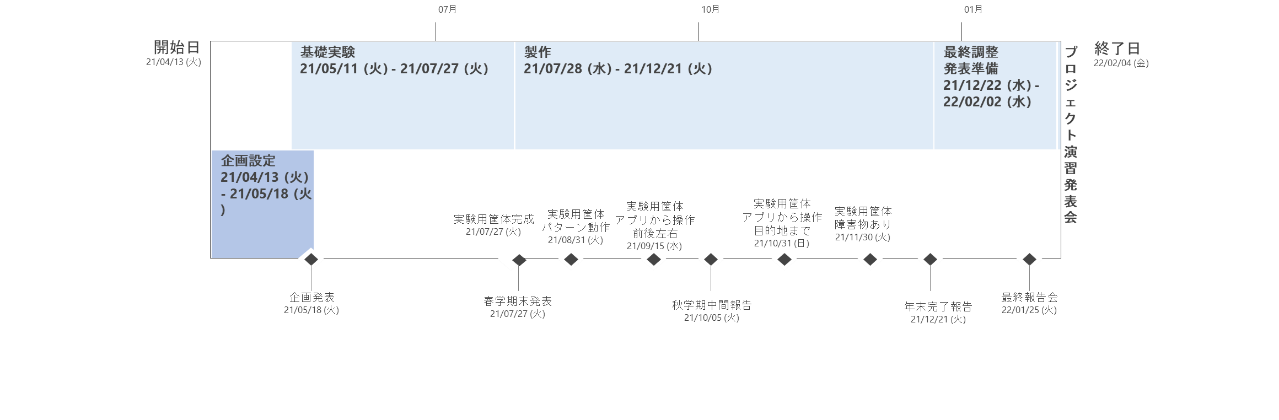
str2str(やSTRSVR)を利用して基準局のGPSデータを配信でき、RTK2GOなどに送信もできる。

RTK法はリアルタイム処理(RTKNAVI, RTKRCV)と後処理(RTKPOST, RNX2RTKP)があるが、収集したログデータがオプションなどで必要なデータが付いていなければ、違った結果になるので注意が必要である。

出力はシリアルやTCP、ファイルなどが複数選択できる。

## サスペンション

# スケジュール



# 参考

1. [驚異のセンチメートル測位！「RTK」の特徴とGPSとの違い、DJIドローンの使用方法など](https://atcl-dsj.com/work/2685/)
2. [tomojitakasu/RTKLIB](https://github.com/tomojitakasu/RTKLIB/tree/master/)
3. [rtklibexplorer/RTKLIB](https://github.com/rtklibexplorer/RTKLIB)
4. <https://akizukidenshi.com/catalog/g/gK-09991/>
5. [google earth](https://earth.google.com/web/)
6. [駅前校のGPS基地局の紹介ページ](https://www.kcg.ac.jp/news/2019/09/47018/)