

SUSTAINABLE DEVELOPMENT GOALS

4 質の高い教育を
みんなに



9 産業と技術革新の
基盤をつくろう



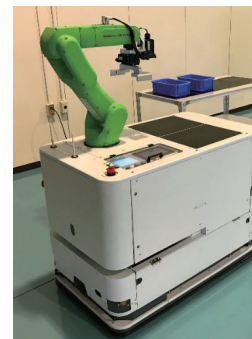
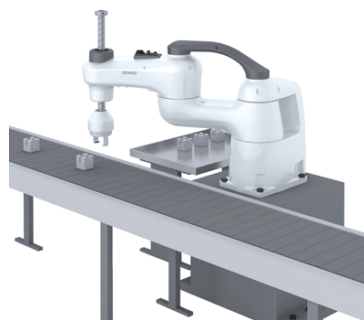
ビジュアルプログラミング言語 で動かせる移動ロボットの製作

生産エンジニアリング科 本間 康介
指導教員 新居 徹哉

1.はじめに

近年,生産の多様化によりベルトコンベアに代わって省スペースで使える自立走行ロボットと協働ロボットの併用が提案されている。

コンベアの代わりに無人移動車を採用することで,移動先を指定することができ,積み下ろし用のロボットを積むことで,柔軟な対応が可能になる。



無人移動車とは?

人が操作しないで搬送などの仕事が行えるロボット車両のことである。

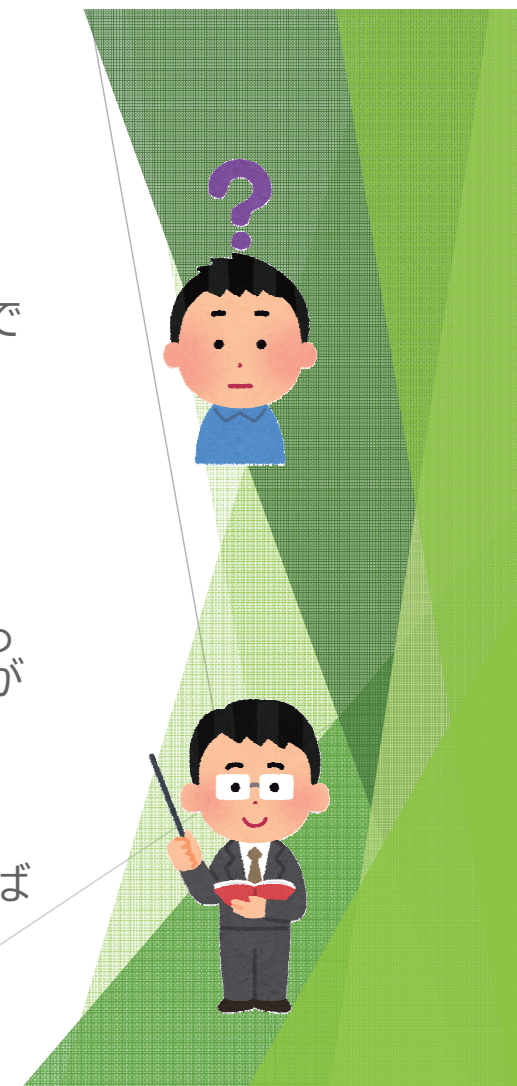
制御方式によって2種類ある。

- AGV

AGVは無人搬送車(「Automated Guided Vehicle」の頭文字をとってAGV)と呼ばれる無人移動車で,移動に目印(ガイド・マーカ)が必要である。

- AMR

AMRは自律走行搬送ロボット「Autonomous Mobile Robot」と呼ばれる無人移動車であらかじめ地図を作って与える必要がある。



AGVとAMRの違い



ガイドやラインが必要だが、人や物を検知する
センサが不要。
製品コストが低い。

AGVの例



事前に地図の作成をしなければならない。
対人対物センサーといった、多くのセンサが必要。
プログラムが簡単で、運用コストが低い。(導入コストが高い)

AMRの例

2. 今回の研究の目標

AMRはコストが高く過剰スペックである。

一般に提案されている「自立走行ロボット(AMR)+協働ロボット」のスタイルではなく,AGV+協働ロボットの移動搬送ロボットを製作する。

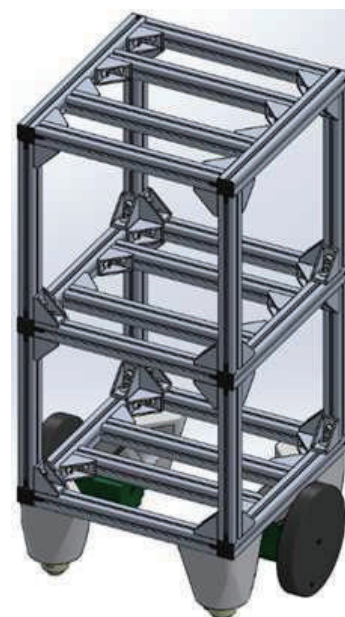
AGVの導入ハードルを下げるために,プログラム作成の容易さに特化したビジュアルプログラミング言語と組み合わせる。

低コストで導入・運用可能な移動搬送ロボットシステムを実現する。

3.構造

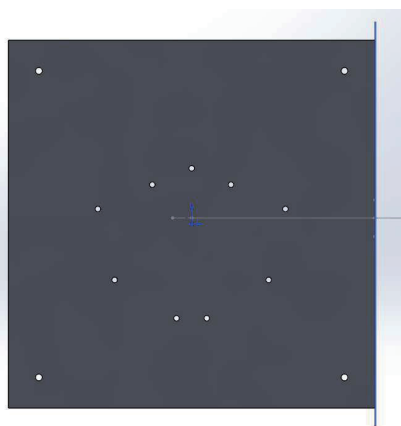
④昨年度までの進捗状況

- ・ AGV本体の完成
- ・ マイコンの選定
- ・ モーター駆動での試走



今年の進捗状況

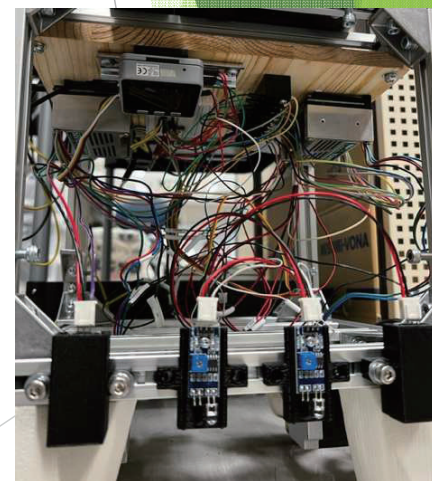
- ・プログラムの製作と部品化
- ・センサの搭載
- ・回路の固定
- ・協働ロボットの固定板の設計・製作



ロボットの固定板



実際の写真



中段の裏面

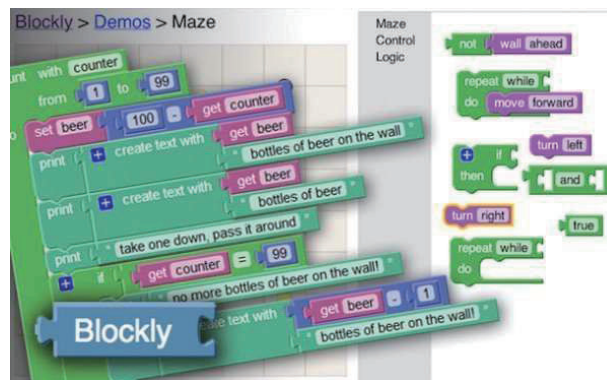
4.AGV制御用のマイコン

- 今回M5Stack社のマイコンモジュール「M5Stack gray」を使用した。
- 専用拡張モジュール「Extend I/O Unit2」を使用しライントレースで走行させるためのセンサを複数取り付けた。



5.M5Stackのプログラミング

- v ビジュアルプログラミング言語Blocklyが使用できるためプログラムが簡単である.
- v Blocklyはインタプリタ型のプログラミング言語なので,製作途中でもプログラムの試行ができ,現場での作業に向いている.
- v 自作部品を作成できる.



プログラム概要

v今回はラインレースをするプログラムを作成した。

「ラインレース動作」

センサーがラインを検知すると検知した側の車輪の速度を下げラインから外れると両輪とも同じ速度に戻るという繰り返して動作する。

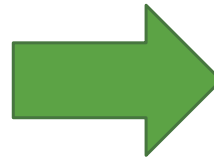
直角の曲がり角に来ると一番外側にあるセンサが反応し、内周側のタイヤが停止し外周側のみのタイヤが回転することで、直角の曲がり角でも動作が可能である。

作成したプログラムと部品化

The screenshot shows a long and repetitive Arduino program. The 'Setup' function configures four digital input pins (0, 1, 2, 3) and four digital output pins (2, 5, 3, 1). The 'Loop' function contains four large 'if' blocks, each corresponding to one of the input pins. Each 'if' block contains a 'do' loop with 'Set' blocks for the output pins, 'Wait' blocks, and 'repeat' loops. This structure is highly repetitive and difficult to maintain.

部品化前プログラム

部品化



The screenshot shows a much simpler and more organized Arduino program. The 'Setup' function is followed by an 'initialize' block. The main logic is now contained within four separate, stacked function blocks: 'forward_L', 'forward_R', 'forward_LL', and 'forward_RR'. This modular approach makes the code easier to read, understand, and maintain.

部品化後プログラム

6.おわりに

v 今後は以下の機能の実装を目標に進めていく予定である。

- (1)実際に,黒テープのライン上を走行させる.
- (2)走行時の安定性を向上させる.
- (3)複数の制御用部品を作り,誰でも走行プログラムを作れるようにする.
- (4)移動先で協働ロボットに作業指令を送り実際に作業させる.

ご清聴ありがとうございました。