

RWL016 Toton エンジニアリングジャーナル

概要

私たち「Toton」は、都立小石川中等教育学校の4年生（高校1年生相当）3人によって構成されたチームです。2024年度1月に発足し、全国大会優勝を目指して活動してきました。本エンジニアリングジャーナルでは、その開発の日々の記録を紹介します。LEGOとArduinoの機体で参加して関東大会で敗退したチームが、完全自作機体で全国大会に挑むまでの軌跡を、ぜひご覧ください。

HP

<https://toton-rcj.jp/>

GitHub

<https://github.com/toton-RCJ>

YouTube

https://www.youtube.com/@toton_RCJ

X(Twitter)

https://x.com/toton_RCJ

メンバーの役割

・五十嵐 栄司

回路・ソフトウェア担当。キャプテン。

部品選定・回路設計・はんだ付け等を担当。また、ライントレースのソフトウェアを担当。

・高津 謙大

ハードウェア担当。

機体設計・組み立て等を担当。3Dプリントやシリコンの成型も行う。

・光本周一郎

ソフトウェア担当。

レスキューのソフトウェアを担当。機械学習・画像処理・レスキューアルゴリズム等を開発。

日々の記録

目次

1. 模索期

2024/01/07～2024/03/24

2. 東東京ノード機体「ポチ」開発期

2024/03/2～2024/11/16

3. 関東ブロック機体「ミケ」開発期

2024/11/17～2025/01/12

4. ジャパンオープン機体「シロ」開発機

2025/01/13～2025/03/19

1. 模索期

2024/01/07

旧チーム「—・—・—　・—・—・　・—・・　・・・—」が2024 関東ブロック大会で敗退。

光本が加入し、新チーム「ツートン」(後に Tutan に改名) 発足。

X (Twitter) アカウントを開設。チーム用の Scrapbox のページを作成。

メンバーの役割分担を決定。

2024/01/08

Teensy4.1 をメインマイコン、Pololu 製品を駆動部に使用する部品計画を作成。(後に大きく変更)

| | 部品 | 機能 | 単価 (円) | 個数 | 金額 | 用途 |
|-----|------------|----------|-----------|----|-------|-------------------|
| センサ | LBR-127HLD | フォトリフレクタ | 50 | 14 | 700 | ライントレース・レスキュー・キット |
| | S11059 | カラーセンサ | 500 | 3 | 1500 | ライントレース・避難所判別 |
| | SS-10GL13 | マイクロスイッチ | 160 | 6 | 960 | 前、後ろ、レスキュー前に2個ずつ |
| | VL53L1X | ToF センサ | 1320 | 8 | 10560 | 障害物回避、レスキューでの探索 |

| | | | | | | |
|-------|------------------|-------------|------------|----------|-------|------------------|
| | BN0055 | ジャイロセンサ | 2450 | 1 | 2450 | 坂、シーソーの検知、90°旋回 |
| | TCA9548A | I2C マルチブレクサ | 1070 | 1 | 1070 | VL53L1X を接続 |
| モータ関連 | SG90 | サーボモータ | 550 | 5 | 2750 | アーム上下、開閉、かご開閉 |
| | CYTRON-MDD3A | モータドライバ | 1265 | 2 | 2530 | モータの制御 |
| マイコン | Teensy4.1 | メインマイコン | 7528 | 1 | 7528 | ロボットの頭脳 |
| | | | | 合計 | 30048 | |
| | | | | | | |
| | | | 単価 (ドル) | | | |
| | Pololu 20D 195:1 | ギヤモータ | 29.95 | 4 | 119.8 | 駆動用。 |
| | Pololu エンコーダ | エンコーダ | 8.95 | 4 | 35.8 | モータのエンコーダ。 |
| | Pololu ブラケット | ブラケット | 7.95 | 4 | 31.8 | ブラケット |
| | Pololu マウンティングハブ | マウンティングハブ | 8.49 | 4 | 33.96 | マウンティングハブ |
| | Pololu ホイール | ホイール | 7.95 | 4 | 31.8 | 直径 90mm の大きなホイール |
| | | | 合計 | 253.16 | | |
| | 1 ドル= | 144.33 | 円換算 | 36538.58 | | |
| | | | | | 総額 | 66586.58 |

2024/01/14

モータ、モータソケット、ねじ、ナットなどの 3D モデルを作成。

2024/01/17

部品在庫管理システムを Excel のマクロを使用して作成したが、以後使うことはなかった。

2024/01/18

ESP32 を入手する。Teensy4.1 に代わってロボットのメインマイコンとして使用する方向で検討を進める。

2024/01/19

ESP32 で Bluetooth の動作確認。無線デバッグの実用性を確認した。

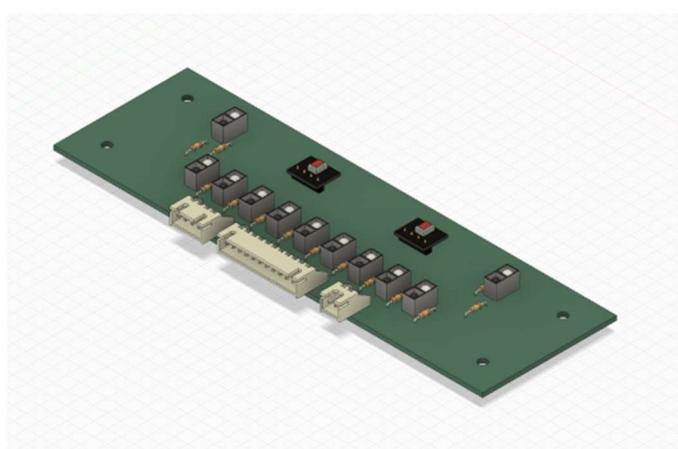
2024/01/20

ESP32 でアナログ入力の動作確認。

ブログの初投稿、SNS 運用開始

2024/01/23

ラインセンサ基板を仮作成。フォトリフレクタ 11 個、カラーセンサ 2 個を使用。



2024/01/24

ジャイロセンサについて、BN0055 と MPU6050 のどちらを使うかを議論。どちらも搭載することになった。

2024/01/27

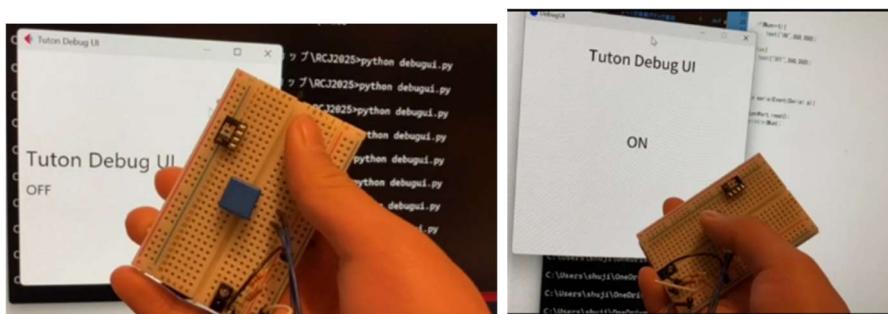
使用する ESP32 は 1 つの予定だったが、ピン割り当てを考えていたところピン数が足りないことに気づいたため、ESP32 を 2 台と Raspberry Pi Pico を 1 つ使用する計画に変更した。

| 割り当て | 属性 | ピン | メイン ESP32 ピン割り当て | ピン | 属性 | 割り当て | | |
|----------------|------------|--------|------------------|--------|------------|-----------|--|-----------------|
| | | 3.3V | | GND | | | | |
| 三軸加速度センサ | ADC0 | GPIO36 | ～入力電圧、PULLUP 不可 | GPIO23 | SCL | Wire_SCL | Wire_I2C S11059, BN0055, MPU6050, リープESP32 | Wire_I2C S11059 |
| 九軸センサ | ADC1 | GPIO39 | ～入力電圧、PULLUP 不可 | GPIO22 | SDA | Wire_SDA | | |
| レスキュー フォトリフレクタ | ADC6 | GPIO34 | ～入力電圧、PULLUP 不可 | GPIO1 | | | | |
| 前方 フォトリフレクタ | ADC7 | GPIO35 | ～入力電圧、PULLUP 不可 | GPIO03 | | | | |
| motor1_1 | ADC4, PWM | GPIO32 | | GPIO21 | SDA | Wire_SDA | | |
| motor1_2 | ADC5, PWM | GPIO33 | | GPIO0 | GND | UART0 | | |
| motor2_1 | ADC18, PWM | GPIO29 | | GPIO13 | SCL | Wire1_SCL | UART1 | |
| motor2_2 | ADC19, PWM | GPIO28 | | GPIO10 | SDA | Wire1_SDA | UART2 | モスクライ |
| motor3_1 | ADC11, PWM | GPIO27 | | GPIO05 | | motor4_2 | | |
| motor3_2 | ADC16, PWM | GPIO14 | | GPIO17 | U2_TX | U2_RX | | |
| motor4_1 | ADC15, PWM | GPIO12 | | GPIO16 | U2_RX | U2_TX | | |
| | | 3V | | GPIO04 | ADC10, PWM | motor1 | | |
| | | | | GPIO03 | ADC11, PWM | motor2 | | |
| | | | | GPIO12 | ADC12, PWM | motor3 | | |
| | | | | GPIO15 | ADC13, PWM | motor4 | | |
| | | | | GPIO08 | | | | |
| | | | | GPIO10 | | | | |
| | | | | GPIO07 | | | | |
| | | | | GPIO6 | | | | |

| 割り当て | 用途 | ピン | ノールピ32ピン割り当て | ピン | 用途 | 割り当て |
|-----------|-------|--------|-----------------|--------|----------|------------------------------|
| | | 3.3V | | GND | | |
| | | EN | | GPIO23 | servo1 | |
| アナログ入力 | ADC0 | GPIO35 | - 入力電圧、PULLUP不可 | GPIO22 | SCL | Wire_SCL メインESP (as a slave) |
| アナログ入力 | ADC3 | GPIO39 | - 入力電圧、PULLUP不可 | GPIO1 | Wire_SDA | Wire_I2C_S11059 |
| アナログ入力 | ADC6 | GPIO34 | - 入力電圧、PULLUP不可 | GPIO3 | servo2 | |
| アナログ入力 | ADC7 | GPIO35 | - 入力電圧、PULLUP不可 | GPIO21 | SDA | Wire_SDA |
| アナログ入力 | ADC4 | GPIO37 | | GPIO2 | | UART0 |
| アナログ入力 | ADC5 | GPIO23 | | GPIO19 | SCL | Wire_SCL |
| アナログ入力 | ADC18 | GPIO25 | | GPIO18 | SDA | Wire_L_SDA |
| アナログ入力 | ADC19 | GPIO26 | ラインセンサモモに接続します | GPIO5 | | UART1 |
| アナログ入力 | ADC17 | GPIO27 | | GPIO17 | U2_TX | |
| アナログ入力 | ADC16 | GPIO14 | | GPIO16 | U2_RX | |
| ADC15使えない | | GPIO12 | | GPIO4 | ADC10 | servo4 |
| | | GND | | GPIO0 | ADC11 | servo5 |
| アナログ入力 | ADC16 | GPIO13 | | GPIO5 | ADC12 | 基準电压 |
| | | GPIO9 | | GPIO15 | ADC13 | |
| | | GPIO10 | | | | |
| | | GPIO11 | | | | |
| | | 5V | | GPIO6 | | |

| 割り当て | ピン | Raspberry pi picoピン割り当て | ピン | 割り当て |
|----------|------|-------------------------|-----------|----------------|
| UART0 TX | GP0 | VBUS | | |
| UART0 RX | GP1 | VSS | | |
| | GND | GND | | |
| XSHUT6 | GP2 | 3V3_EN | | |
| XSHUT7 | GP3 | 3V3_OUT | | |
| I2CO_SDA | GP4 | ADC_VREF | | |
| I2CO_SCL | GP5 | | | |
| | GND | GND | | |
| | | ディスプレイ処理と、ToF&スイッチ | | |
| XSHUT8 | GP6 | GP28 | XSHUT1 | UART0 無線デバッガ |
| XSHUT9 | GP7 | GP27 | XSHUT2 | UART1 メインESP |
| UART1_TX | GP8 | RUN | | SPI0 タッチディスプレイ |
| UART1_RX | GP9 | GP22 | XSHUT4 | SPI1 |
| | GND | GND | | |
| マイクロスイッチ | GP10 | GP21 | XSHUT5 | I2CO ToF |
| マイクロスイッチ | GP11 | GP20 | SPI0_CS_2 | I2C1 |
| マイクロスイッチ | GP12 | GP19 | SPI0_TX | |
| マイクロスイッチ | GP13 | GP18 | SPI0_SCK | |
| | GND | GND | | |
| マイクロスイッチ | GP14 | GP17 | SPI0_CS_1 | |
| マイクロスイッチ | GP15 | GP16 | SPI0_RX | |

XIAO ESP32C3 を取り外しできる無線デバッガとして使用することを計画しており、その際の PC 側の UI を何で開発するか検討する。Flet (Python) と Processing が選択肢としてあがり、両方について簡単なプロトタイプを作成して実現可能性を確認した。



2024/01/28

ESP32 と Raspberry Pi Pico を使用する計画を立てていたが、ピン数不足により複雑な構成となってしまっていたため、マイコンボードを使用するのではなく、チップマイコンを表面実装する方針を考える。チップマイコンとしてロボカッパーに多く採用されていて性能が高い STM32 マイコンを検討。

STM32 マイコンを表面実装する際、周辺回路設計のハードルが高いという懸念が出たが、時間的に余裕があるため技術を身に着けて設計できるようになることを目指した。

2024/02/02

全体的な仕様を固めるオンラインミーティングを開催。

マイコンは STM32 を採用することに決定。

駆動モータには連続回転シリアルサーボの STS3032 を採用することに決定。このモータの方が Pololu の DC モータよりも小型であり、シリアル通信で簡単に正確な制御ができる。トルクや回転数も十分であり、STS3032 が適していると考えた。

レスキューで銀と黒の検知はフォトリフレクタと導通確認の二重でチェックする方針に決定。ボールをつかむ部分は金属で作ることになった。

ディスプレイの搭載は断念。無線デバッガとして XIAO ESP32C3 を搭載し、値を無線で垂れ流す方針に決定。

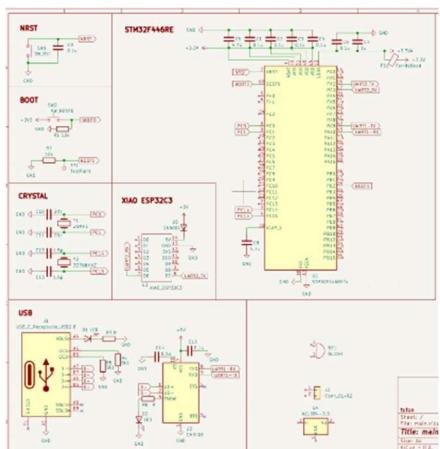
2024/02/03

STM32 マイコンの回路設計を開始。

USB コネクタには Type-C を採用することを決定。

書き込みは UART 書き込みにすることを決定。

周辺回路の設計方法を習得、STM32 周りの回路設計が完了。



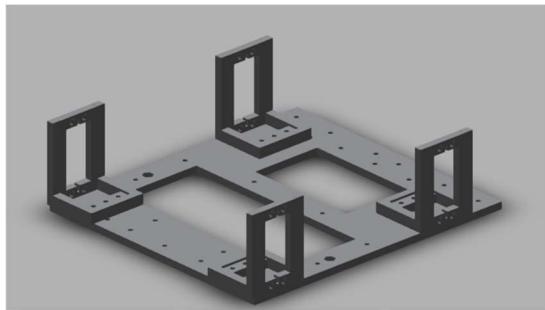
2024/02/04

コストカットのため、STS3032 の代替として GWS S35 STD を検討。

2024/02/05

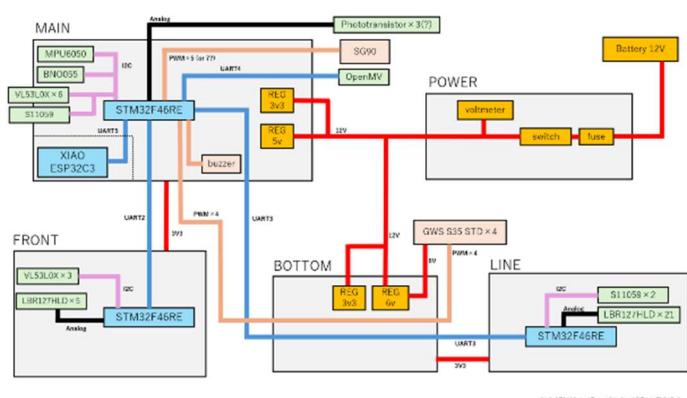
名古屋大会に見学に行くことを検討。

GWS S35 STD を使用した一層目の設計を作成。



2024/02/07

回路の構成図を作成。



電源は eneloop 8 本を想定。リポバッテリも視野に入れていたが、安全性の観点から保留。

2024/02/09

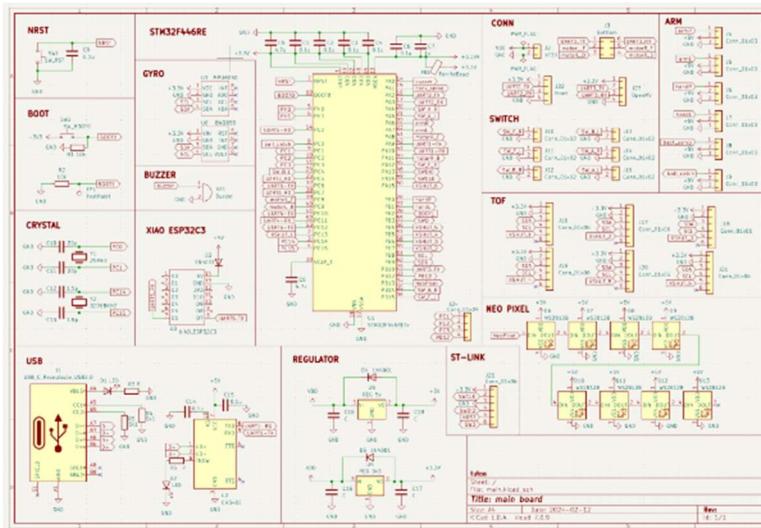
Nucleo F446RE を入手し、STM32 の開発方法を確認。

ソフトウェアの開発は PlatformIO と Arduino フレームワークの組み合わせで進めることを決定。記事にまとめる (<https://qiita.com/shuji4649/items/deaafccbd064dc9000a>)

2024/02/12

メイン基板の回路設計が完了。

この回路図ではコネクタの数が 20 個を超えており危機感を持つ。回路構成を変更する必要性を感じる。

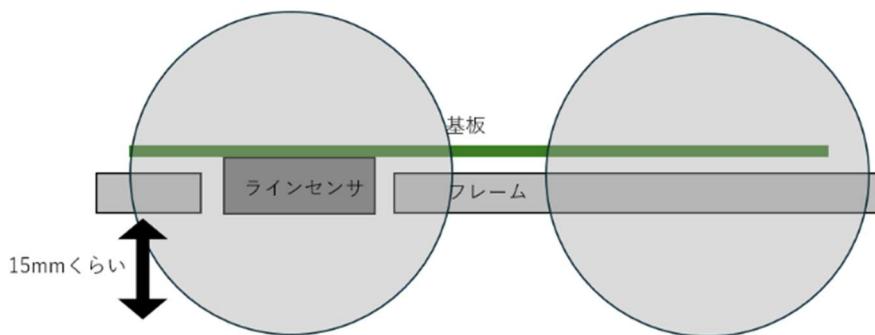


2024/02/13

ソフトウェア担当がやることがないため、2024 関東に出場した LEGO 機体の開発に手を付ける。

乾電池 8 本からリポバッテリを 1 つ搭載する方針に変更。

フォトリフレクタと床の理想の距離は 15mm 程度だが、現状の設計では 25mm であることを発見。

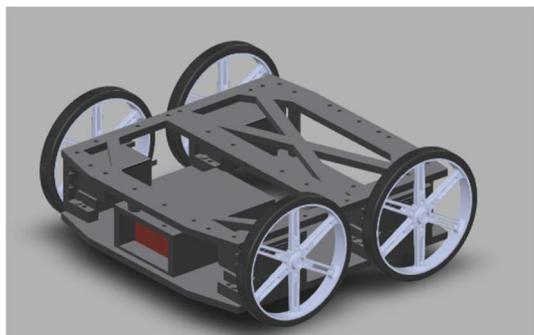


モータを一層目の板の上面に乗せていたが、それを少し浮かせて、その下にラインセンサを入れ込むことでラインセンサの幅を拡張すると同時にフォトリフレクタと床の距離を縮めることを提案。

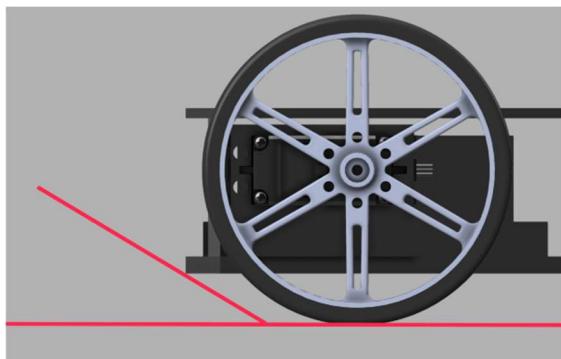
2024/02/14

リポバッテリは Kypom 7.4V 1300mAh を使用することを決定。

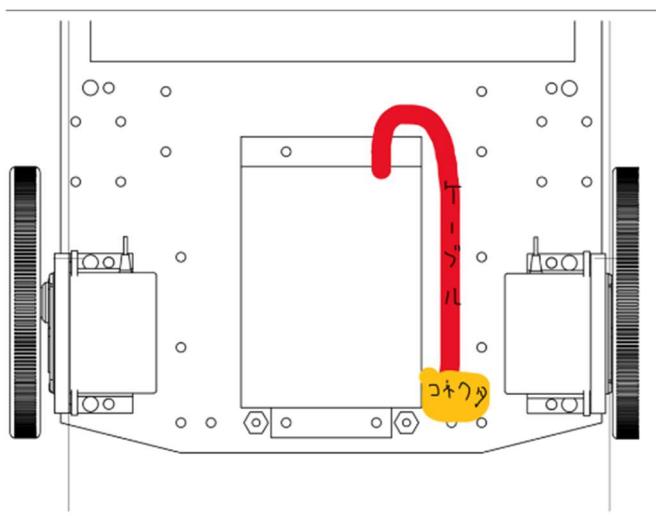
モータ、バッテリ、一層目、二層目のハードウェアの基本設計が完了。



現状の設計では坂道で登れない可能性を指摘。



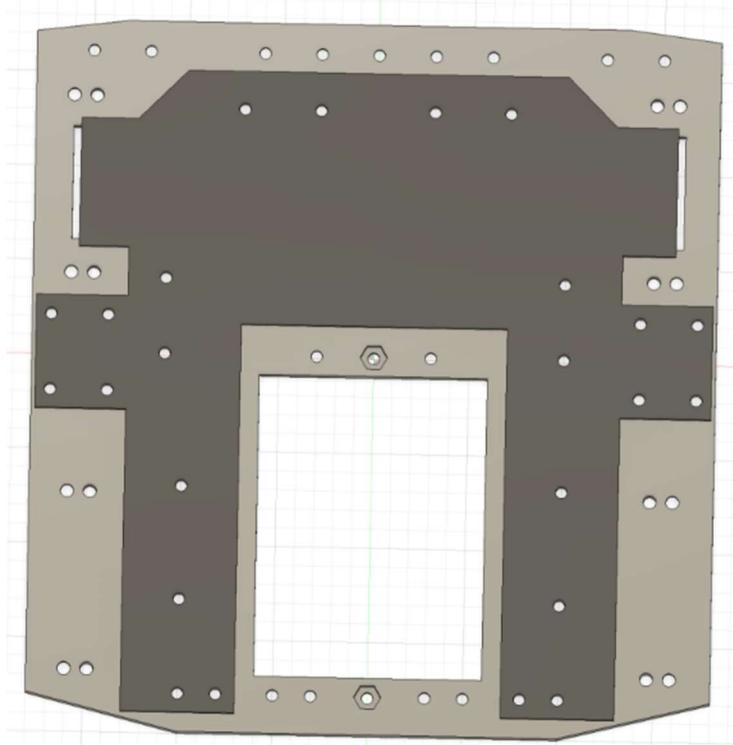
リポバッテリの交換時にケーブルの取り回しが面倒になる可能性を指摘。延長ケーブルを作成することで解決する方針を決定。



ピン数をより拡張するため、STM32F446RE から STM32H723ZG に変更することを検討。

2024/02/19

メイン基板の外形を決定。



ToF センサは I2C なためケーブルを介しての接続ではなくピンヘッダで基板に直付けすることを決定。

被災者発見にカメラを両側 1 つずつ、合計 2 つ搭載することを決定。

2024/02/22

使おうとしていた STM32H723ZG が PlatformIO に無いことに気づく。

Nucleo ボード用のものから移植することで開発できる方法を発見。記事にまとめる

(<https://qiita.com/shuji4649/items/6f86d65fc1442101d06d>)。

2024/03/08

タッチセンサにロードセルを採用することを決定。

2024/03/014

ラインセンサ基板とメイン基板を別の基板にし、連結ピンヘッダで接続することを提案。

2024/03/19

レスキュー用のサーボモータとして SG90・SG92R を考えていたが、耐久性の懸念から、金属ギアの MG90D に変更。

2 層目置く部品を全て 1 つの基板にまとめることを提案。(前の方に ToF 系、左後ろにタクトスイッチ・ト

グルスイッチ、かごの下にディスプレイ・電圧計、右後ろに電源スイッチ・ヒューズ)

後ろの電源周りの基板は一層目からスペーサを挟んで取り付けることを提案。スペーサに電流を流すことで配線を減らすことを提案。走行するための機能が一層目で完結するという利点を確認。

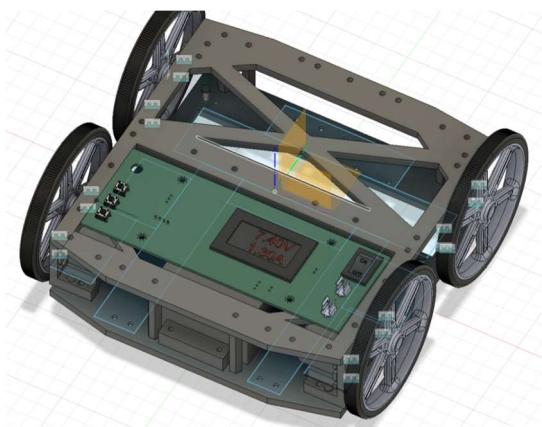
JLCPCB 様にスポンサーになっていただくことが決定。

2024/03/20

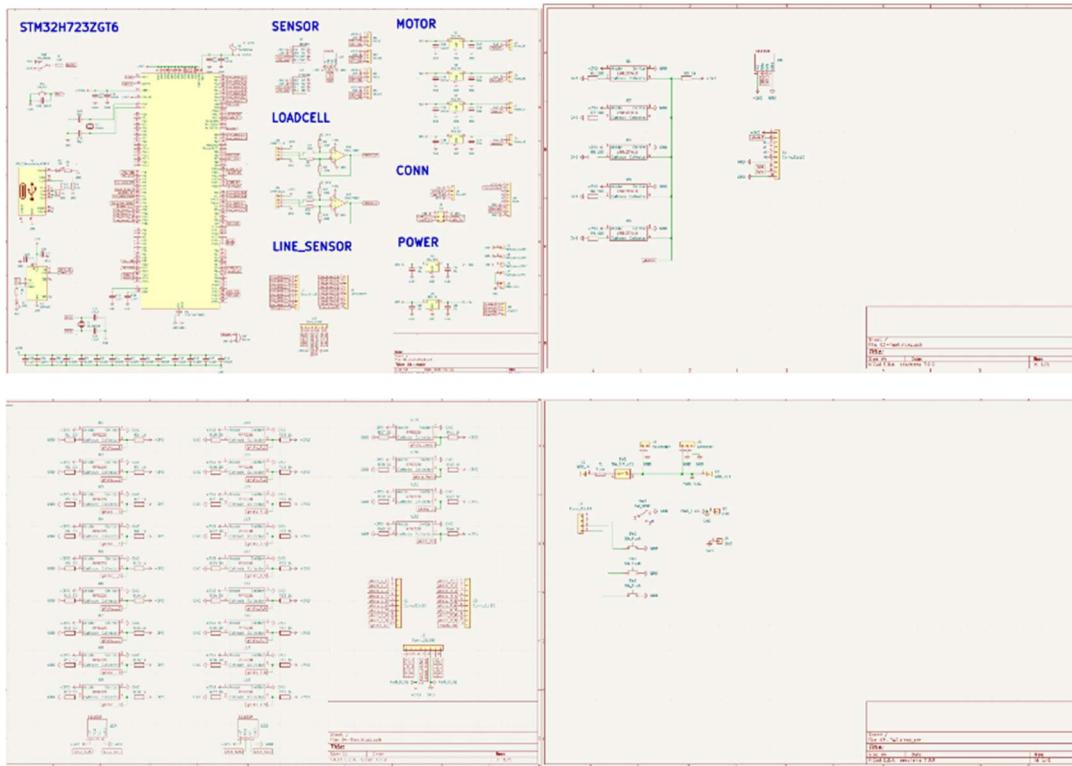
被災者回収機構を試作。ボールを確実につかんでホールドすることのできる機構を目指とした。



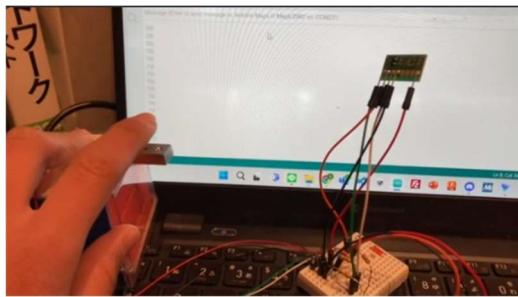
電源基板を試作。



メイン基板、正面基板、ラインセンサ基板、電源基板の回路設計が完了。



オペアンプを介してロードセルの値を読み取る実験を行う。1000 倍程度に増幅するとちょうどよいことが分かった。



2024/03/23, 24

ジャパンオープン 2023 に見学しに行く。

全国大会優勝レベルのロボットを見てモチベーションが爆発的に高まる。

レスキューライン競技の走行を見て、全国大会レベルの戦いにおいて何がポイントとなるのかについて学ぶ。

来年実現したいことを書き出す。

2024年3月24日

shuji 2024/03/24 14:50

強いレゴ機体を作つておいて安心して自作機体を作りたい
レスキューで正確に動きたい
縁検知ミスりたくない
初心者マーク買わない
メンテナンス性良くしたい
プログラム綺麗にしたい
操作性よくしたい
坂道パンプも軽々と乗り越えたい
マツケンサンバ流したい
全部オープンソースにしたい
かっこいいレスキュー機構つけたい
速くてきれいなライントレースしたい
カメラでライントレースの先を見たい
かっこいい開発ツール作りたい
LEDビカビカさせたい
満点走行して全国優勝したい

2025 ルールで被災者の作り方が変わることを知る。

これらを受け、ロボットの構想を 0 から考え直すことにする。

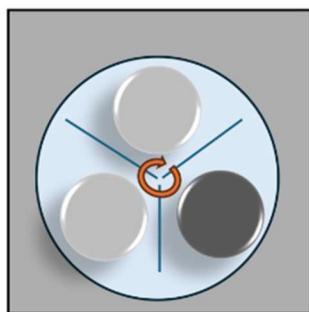
2. 東東京ノード機体「ポチ」開発期

2024/03/25

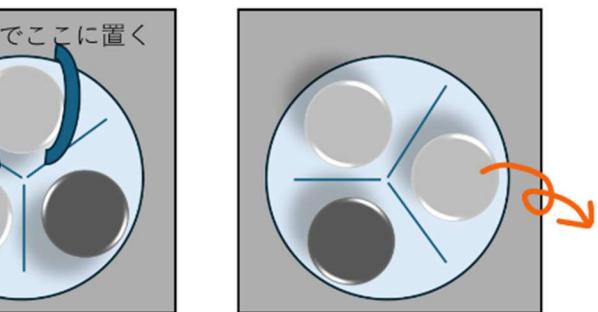
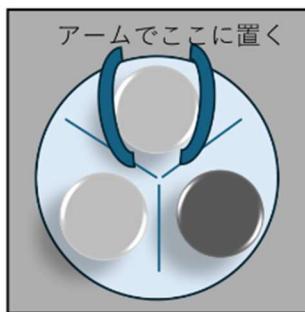
ジャパンオープンの走行を見て、小型化が重要であることを再認識し、ロボットを 15cm 四方のサイズで作ることに決める。

3D CAD 上で草案を作成する。

メイズのロボットから図のようなレスキュー機構の着想を得る。(後にスペースの問題から断念)

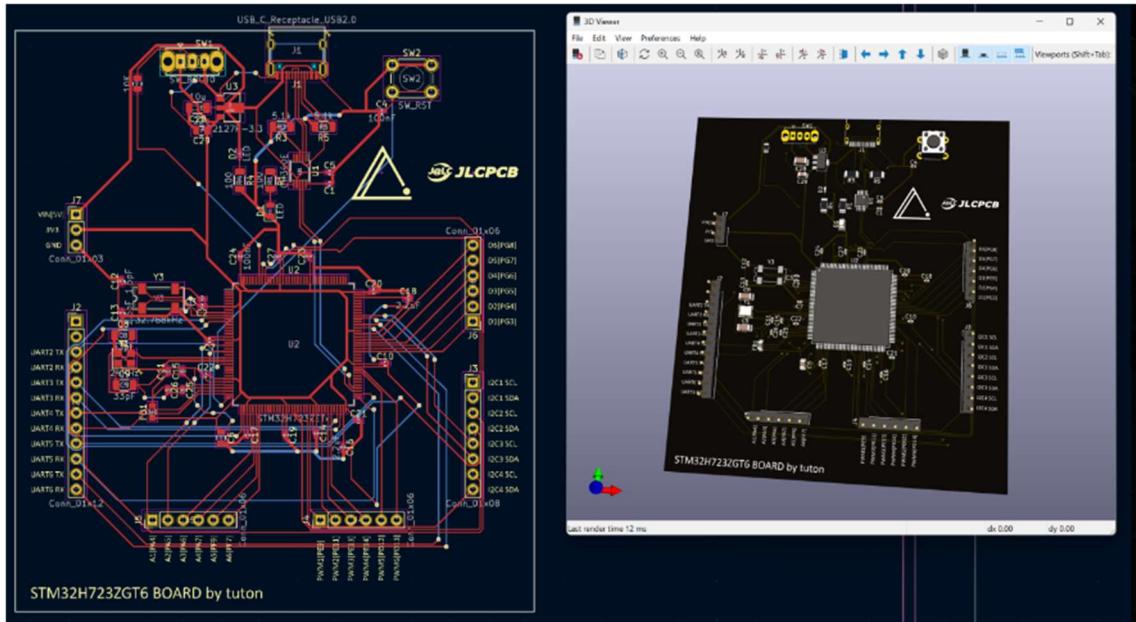


円形の板がぐるぐる回る



捨てたいものを横に持ってきて何かしらで落とす

STM32H723ZG の実験用ボードを設計、発注。



2024/03/26

必要となるトルクを計算した結果、STS3032 と GWS S35 STD はどちらも余裕があることを確認。

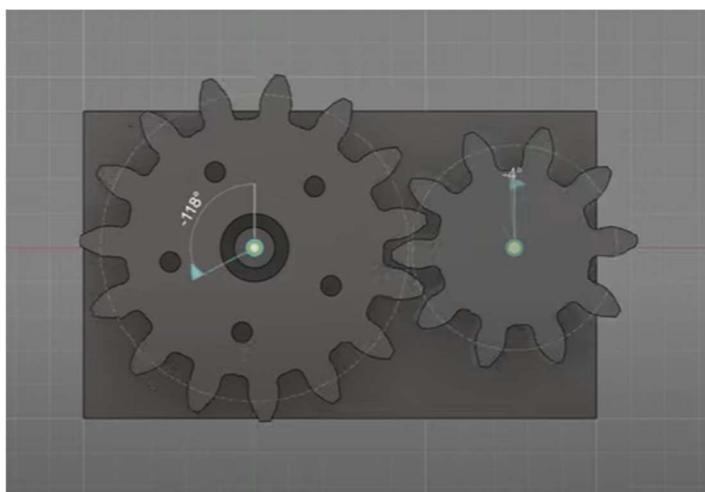
2024/03/27

ピンポン玉に米を入れて 2025 年の被災者を作成。坂道は転がることを確認。

サイズと性能から、足回りのモータは STS3032 を使用することに決定。

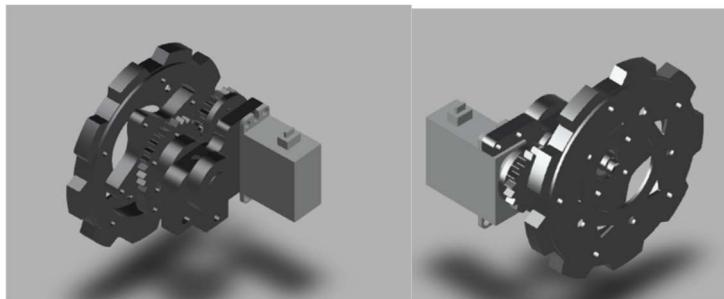
2024/03/29

ギア周りを作成。



2024/03/30

足回りのモジュールの設計が完了



EV3Lab (EV3 教育版ソフトウェア) で、Arduino と UART 通信する手法を新たに確立 (自分調べで世界初)。I2C では安定性に不安があり、MicroPython は動作が遅いため、これは大きな成果である。予備機として作成する LEGO 機体に採用することを予定。

(後に GitHub にて公開 : <https://github.com/shuji4649/EV3LabUART>)

2024/04/02

タイヤの素材として、シリコン・TPU フィラメント・HP フィラメント・高弾性ラバーレジン (光造形 3DP) のどれがよいかを検討。

シリコンが望ましいが、耐久性について懸念がある。

カラーセンサに TCS34725 を使用することを検討。

STM32H723 基板が到着。動作確認を行う。

2024/04/10

フォトリフレクタをコンパレータによるデジタル読みにすることを検討。

2024/04/13

基板の省スペース化のために STM32 の書き込み回路を書き込み基板として別にすることを検討。ピンヘッダだけ出しておくことでスペースの削減を狙う。

基板の構成を整理。

メイン基板 一層目後方に配置

- メインマイコン
- モータコネクタ・駆動回路
- ジャイロセンサ
- ブザー
- 横・後ろToF
- 後ろカラーセンサ
- ロードセル用オペアンプ

ラインセンサ基板 一層目前方に配置

- サブマイコン
- フォトリフレクタ
- カラーセンサ
- ToF(床までの距離測定、いらんかも)

おまけ (といいつつ必要)

- 書き込み基板
- OpenMVシールド

二層目基盤 二層目前方に配置

- サブマイコン
- サーボコネクタ
- 横ToF
- ディスプレイ
- 無線デバッガ
- ✨フルカラーLED✨

前方基盤 二層目基板から垂らす

- ToFたくさん
- カラーセンサ

電源・UI基盤 二層目後方に配置

- 電圧計・ヒューズ
- 電源スイッチ
- UIスイッチ (プッシュ・トグル)

カメラを OpenMV と UnitV のどちらを使用するか議論。視野角的にはどちらも問題ないことを確認。

2024/04/14

書き込み基板にマグネットコネクタの採用を検討するが、接触の信頼性の観点から却下。

2024/04/16

TCS34725 の使用を以前検討したが、応答速度が遅いという情報を見つけたため却下。S11059 で設計を進める。

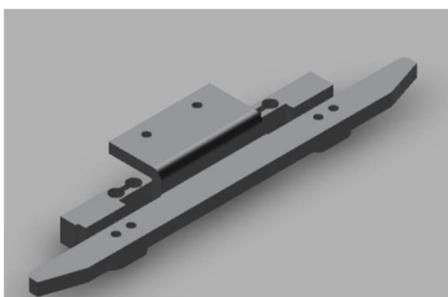
2024/04/18

レスキューに LiDAR が必要かどうかを検討。

コストと有効性を考えた結果必要でないと判断。

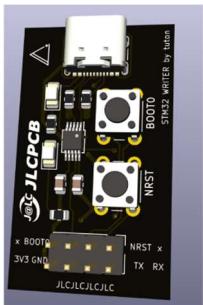
2024/04/21

ロードセルを使用したバンパーの設計を行う。



2024/04/25

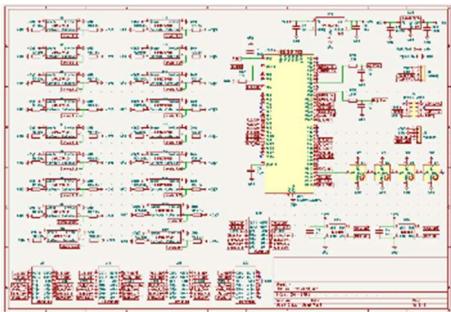
書き込み基板の設計が完了。



2024/04/26

PWM 平滑化の CR 回路の抵抗・コンデンサの定数値を計算。

コンパレータを使用したラインセンサの回路設計が完了。



2024/04/28

スペースの削減のために、ガラス管ヒューズではなく低背ヒューズを使用することを決定。

多くのサーボモータを制御するため、I2C で制御できる PCA9685PW を表面実装で搭載することを決定。

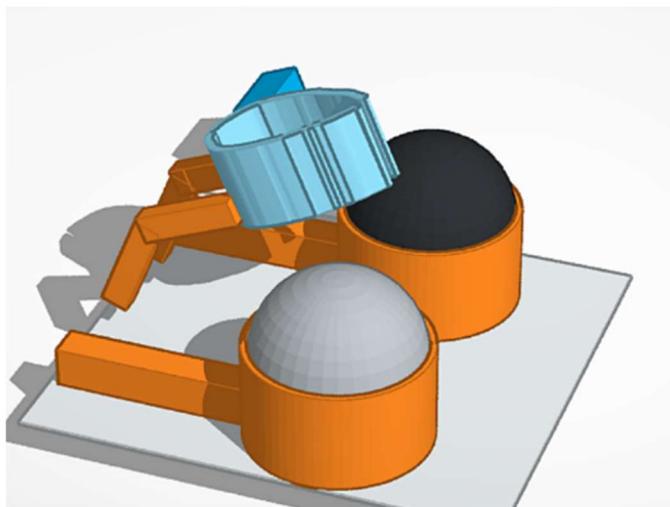
2024/05/03

レスキュー機構について、ボールは高々3つであり、1つはアームで保持をしておけばいいので、かごが保管するボールの数は2つで十分であることを発見する。

CNC で発注したタイヤの軸が到着。十分堅牢に固定できていることを確認する。



新しいレスキュー機構が提案される



複雑な機構であることに対してメリットが小さいため却下。

2024/05/05

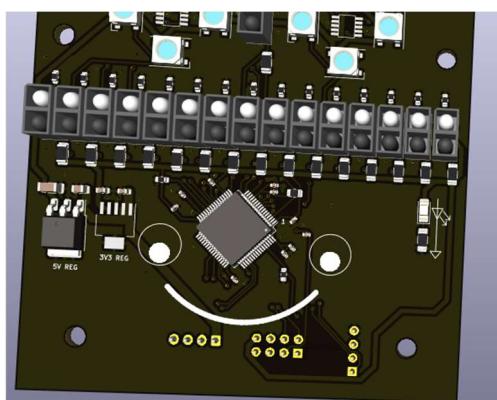
電源基板と UI 基板は一体化させ、スペースと操作性を考え、それをロボット後方に取り付けることを決定。

2024/05/06

ラインセンサ基板について、STM32F446RE の ADC のピン数は 16 であり、使用するフォトリフレクタ 16 個の値をアナログで読み取れるため、コンパレータの使用をやめる。

2024/05/07

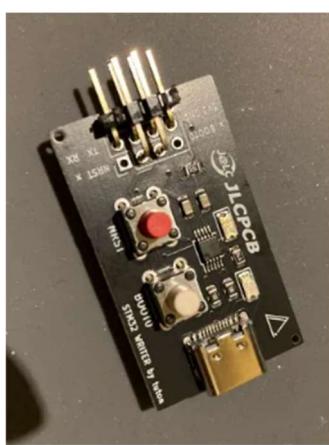
ラインセンサ基板の設計が完了。



2024/05/24

発注した書き込み基板とラインセンサ基板が到着。

当初使用しようとしていたカラーセンサである S11059 が在庫なしのため互換性のある S13683 に変更。

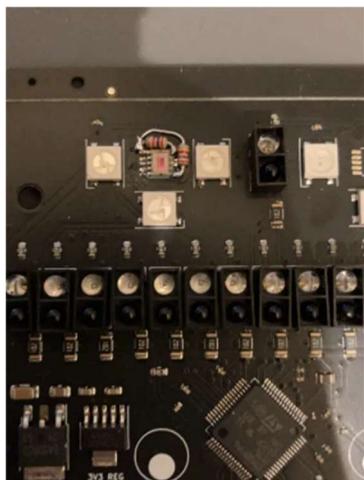


ラインセンサ基板への書き込みが正常にできることを確認。

LED が正常に動作することを確認。



I2C のプルアップ抵抗を忘れていたため、修正。カラーセンサが正常に動作することを確認。



2024/05/26

Processing でソフトウェアを作成し、フォトリフレクタとカラーセンサの動作確認。

2024/05/27

ラインセンサ基板が発熱により故障。三端子レギュレータの放熱処理をしていなかった。

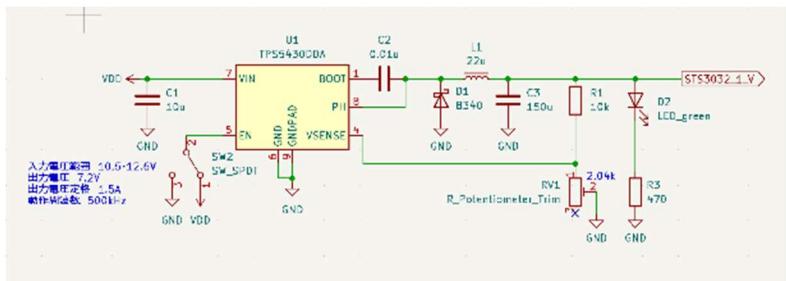
三端子レギュレータではなく DCDC コンバータを使用することに方針変更。

2024/05/30

STS3032 の電源用の DCDC コンバータを TPS5430 で自作することを検討。回路設計を学ぶ。

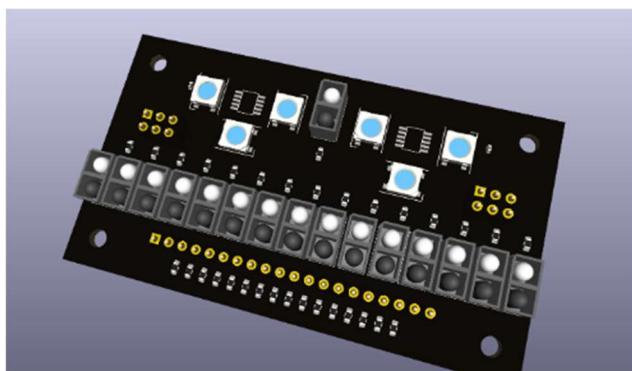
2024/06/02

TPS5430 の周辺回路の設計を完了。



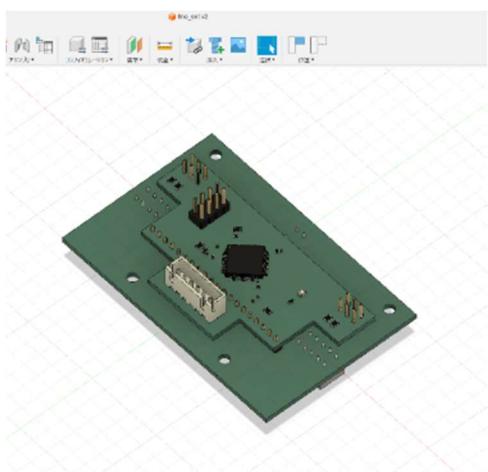
2024/06/05

新しいラインセンサの基板設計を開始。



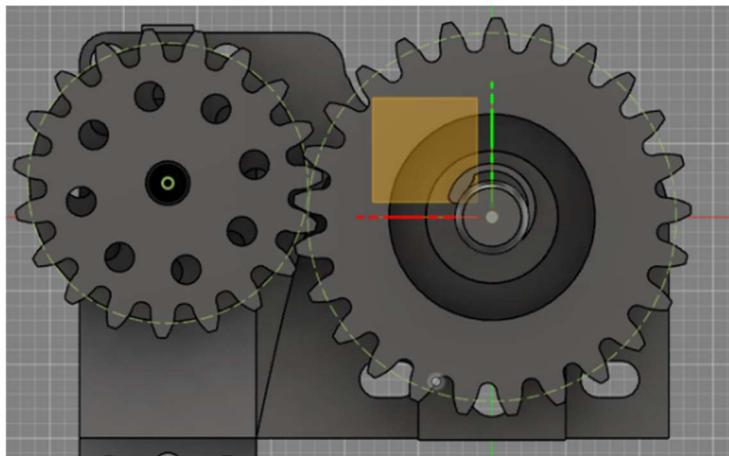
2024/06/08

ラインセンサ基板を、センサ基板とマイコン基板の 2 つに分けてピンヘッダで接続する方針で設計を進め
る。スルーホール部品の真上に表面実装のマイコンを置けるためスペースを有効活用できると考えた。



2024/06/17

ギアの形状を単純な鋸歯状からインボリュート曲線に基づく形状に変更した。



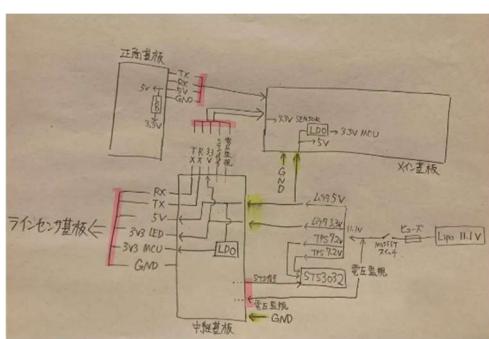
2024/06/18

5V と 3.3V を生成する DCDC コンバータに OKL-T/6-W12N-C を採用することを検討。6A まで生成することができ、ロボットの中の 5V・3.3V を全てこれでまかなうことができる。

2024/06/20

バッテリの電圧をマイコンから監視する手段について議論する。抵抗で分圧はマイコンを壊すリスクがあるため、電圧監視 IC ・ M51957B を使用することを検討。

ロボットの基板構成を整理。



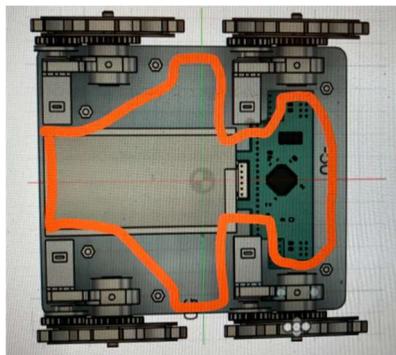
独自のステッピングレギュレータモジュールの設計完了。



2024/06/22

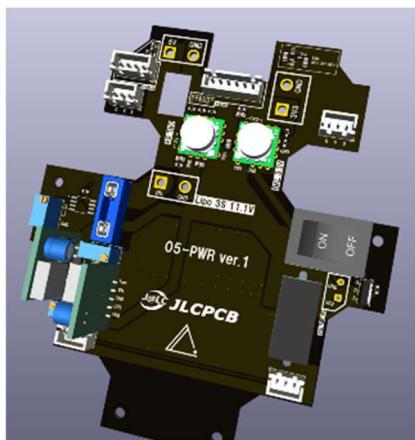
中継基板と電源基板を一体化させることを提案。

電源基板の部品を置くスペースが足りない、中継基板までケーブル出すとごちゃごちゃする、コネクタの数少ない方が信頼性上がるという理由から。



2024/06/25

電源基板の設計が完了



2024/06/28

カメラでライントレースをする実験を行う。



2024/07/04

発注していた電源基板が到着。動作確認を行う。

2024/07/06

電源基板の三端子レギュレータのピン配置を間違えて設計していたことが判明。ラインセンサ基板のマイコンを壊す。

2024/07/12

二層目の前方にアームを取り付ける都合上、二層目の高さをあげることを提案。

2024/07/15

3.3V の DCDC コンバータに負荷を接続すると電圧が低下する現象を発見。DCDC コンバータの端面スルーホールのはんだ付け不良が原因であることが判明。はんだ付けの部分で 32Ω ほどの抵抗があったため、はんだ付けし直したところ正常に動作した。端面スルーホールの基板側をスルーホールにしていたが、表面実装のパッドにした方がよいと判断した。

ラインセンサ基板が正常に動作することを確認。



2024/07/20

シリコンタイヤの製造を進めるも、失敗が続く。

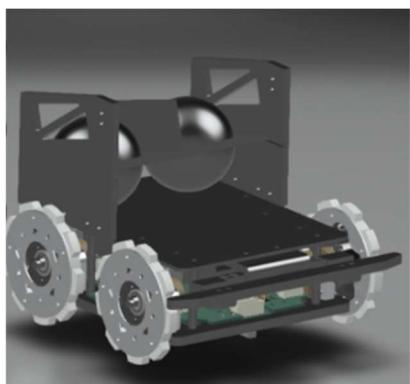
2024/08/06

タイヤを含めた駆動部のモジュールの組み立てが完了。



2024/08/07

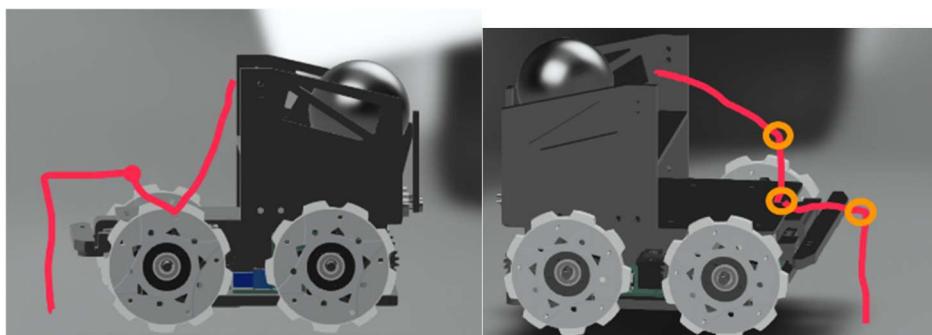
3D CAD 上でかごを試作。



2024/08/09

アームの軸数を検討する。

最も単純なのは左のような1軸だが、かっこよくしたいため右のような2軸を検討。



2024/08/10~2024/08/16

NEST ロボコンに向けて LEGO ベースの機体の開発を進めるも、準備時間不足により出場は断念。

2024/08/17

自作 ToF モジュールの設計が完了。



2024/08/19

アームの根本にステッピングモータを取り付け、アームの自由度を挙げることが提案されるが、コストと

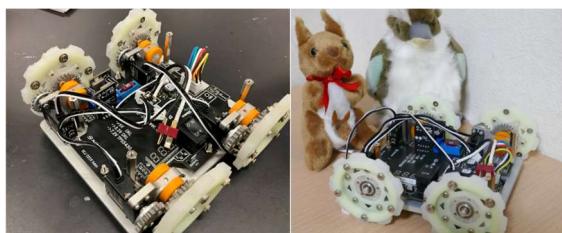
安定性と効果について検討した結果、却下。

2024/08/20

在庫管理・タスク管理・Wiki 管理に Notion を導入。

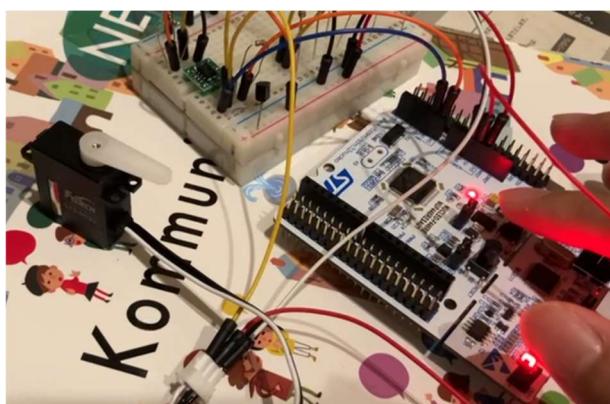
2024/09/03

STS3032 を 4 つ入手し、一層目の組み立てが完了する。



2024/09/04

ブレッドボード上で UART-RS485 変換回路を構築し、STS3032 を制御することに成功。

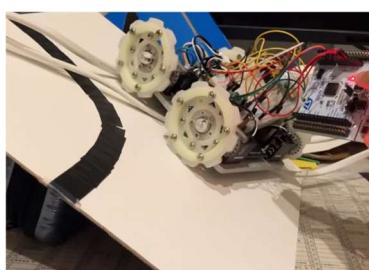


2024/09/05

ギアとタイヤを固定しているねじともう一方のギアが干渉していることを確認。

2024/09/07

ブレッドボードで構築した STS3032 駆動回路を機体に搭載し、走らせてみる。40 度程度の傾斜であれば走行できる性能を確認。



それぞれのモータの回転開始タイミングにずれが生じる現象を確認。

2024/09/13

Nucleo32 マイコンを使用して STS3032 駆動回路をユニバーサル基板に実装し、ロボット単体で走らせることに成功。メイン基板が完成するまでの実験用に使用する。



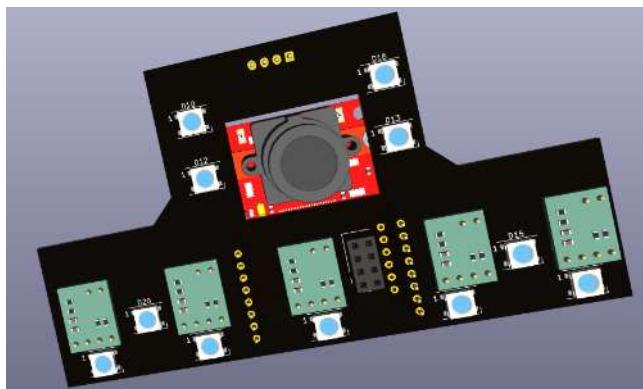
2024/09/14~15

ロボットを学校の文化祭（創作展）で展示。

どこかがショートし、Nucleo 上のレギュレータが燃える。

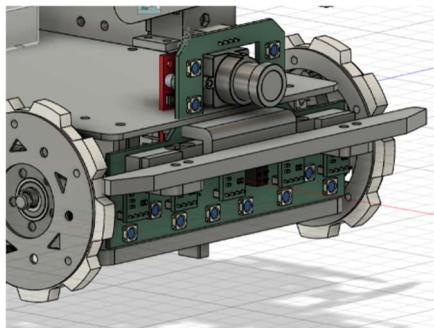
2024/09/17

カメラを基板の穴からぞかせる形にすることに決定。正面基板を設計。



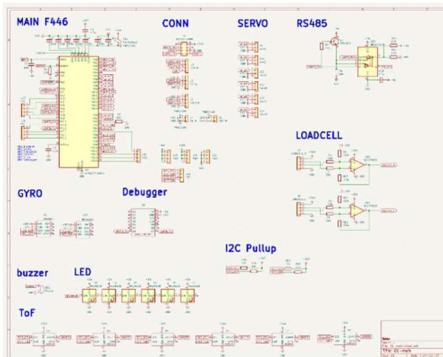
2024/09/18

正面基板をロボットの 3D モデルと組合せて固定方法や干渉の有無について確認する。

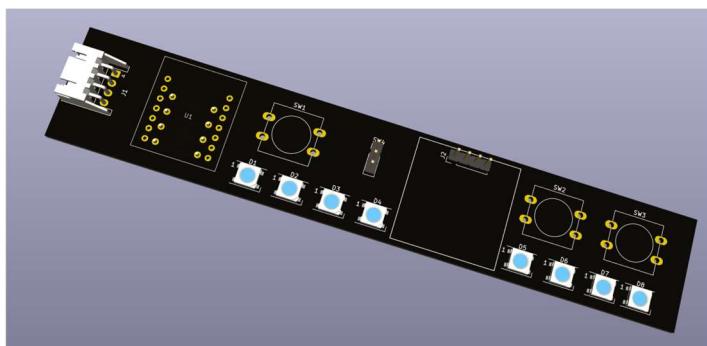


2024/09/19

メイン基板の回路設計が完了。

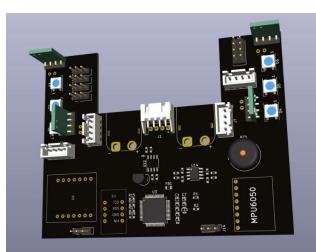


ディスプレイ・UI・スイッチを搭載した UI 基板を作成。



2024/09/21

メイン基板の設計が完了。

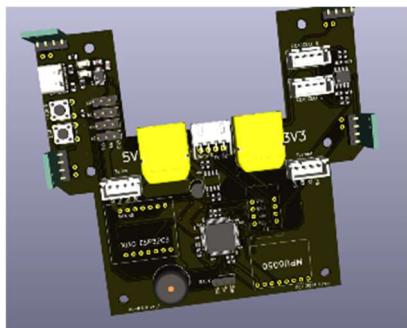


ロボットを持ち運びするケースを購入。



2024/09/28

メイン基板の設計を修正。



2024/09/30

電子部品を整理。



2024/10/10

発注していた基板が到着。メイン基板にミスを見つける（電源コネクタの極性が逆）



2024/10/14

メイン基板にプログラムを書き込めないことが発覚。

フットプリントを変更後に DRC (デザインルールチェッカー) を回さなかったため、配線がずれていたことが原因と分かった。

ジャンパなどによって修正。

2024/10/15

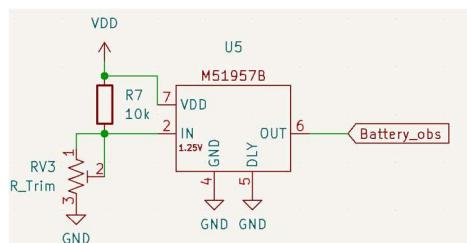
正面基板の動作確認を行う。



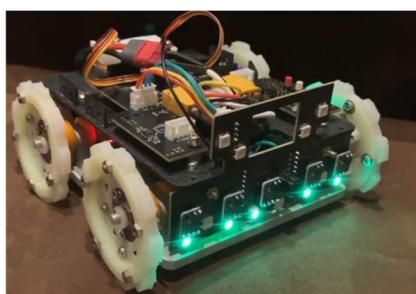
OpenMV をさして動かしたところ、OpenMV のレギュレータが壊れた。正面基板へ送る 5V の電圧降下が見られたが、詳細な原因は不明。以降カメラなしで使用。

2024/10/17

電源基板に電圧監視 IC を実装しようとした時、ふと思い IC のデータシートを見ると、この回路ではマイコンに 11V が加わることが判明（OUT からは VDD と同じ電圧が出力される）。実装は取りやめになった。

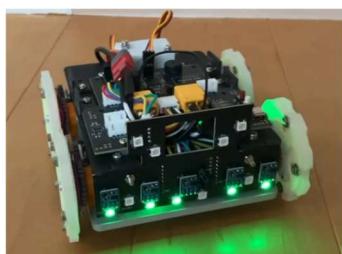


一通り組み立てが完了。



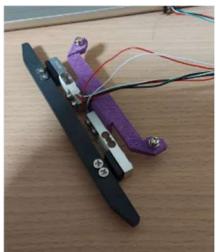
2024/10/20

実際に使用する基板で走らせるに成功。



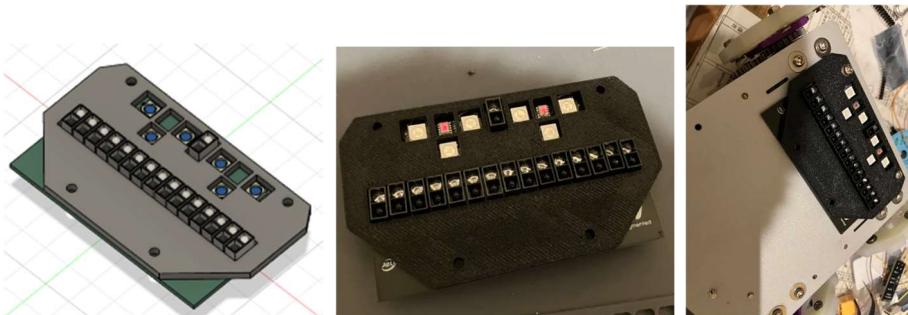
ライブラリの中のコードを編集することで、4つのタイヤの回転の開始タイミングに差が生まれる現象を解消。時間がかかっていたACKの処理を取り除いた。

バンパーが完成する。



2024/10/23

ラインセンサ基板の保護とLEDとカラーセンサの間を仕切ることを目的としたカバーを設計、印刷。



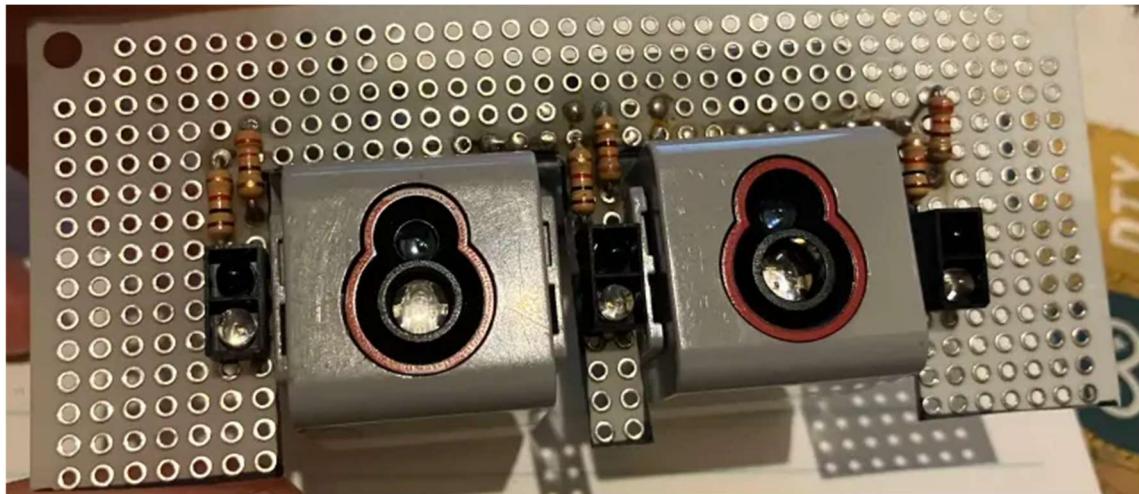
2024/10/26

かごが付く。



2024/10/27

予備機である LEGO 機用のカラーセンサ基板を作成。加工が簡単である薄型ユニバーサル基板を使用した。



2024/10/31

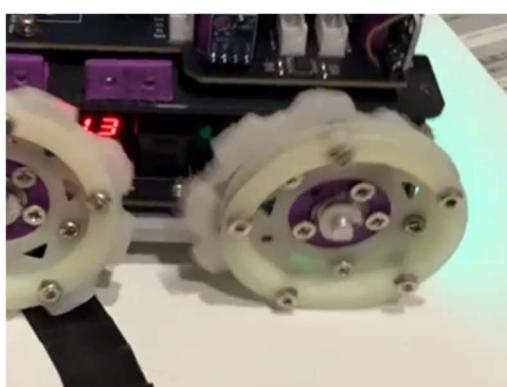
メイン基板と正面基板の UART 通信がうまくいっていないことが判明。

無線デバッガ用に生やしていた UART を正面基板にジャンパし、無線デバッガは使用しないことにした。

(後に原因是回路ではなくソフトウェアだったことが判明。)

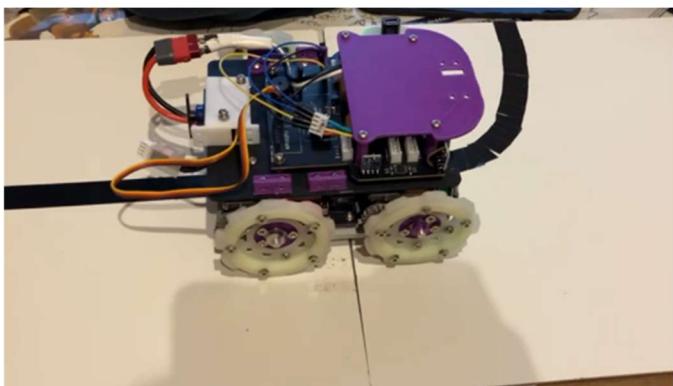
2024/11/01

旋回するとタイヤが取れることが判明。タイヤの厚みを 5mm から 10mm に変更することにする。

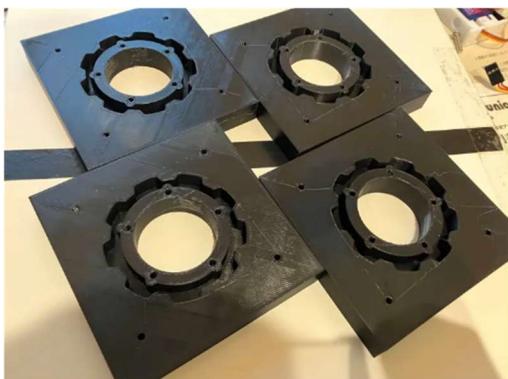


2024/11/02

基本的なライントレースのプログラムを書く。自作機で初のライントレース。

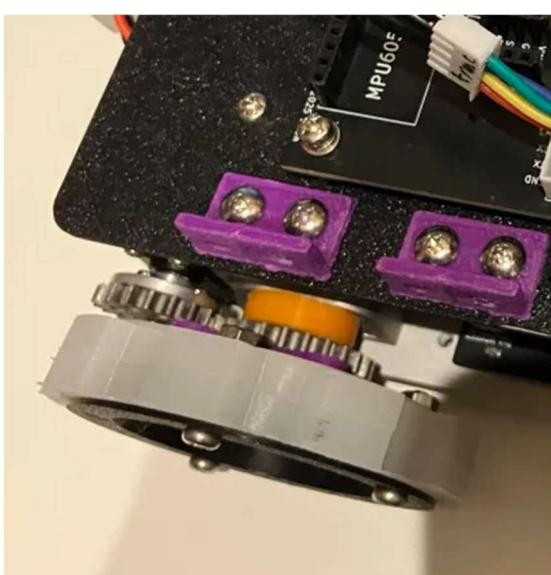


10mm のタイヤ用の型を印刷。一度に多く生産するために 4 つ印刷。



2024/11/04

10mm のタイヤに換装。旋回しても取れなくなった。



2024/11/06

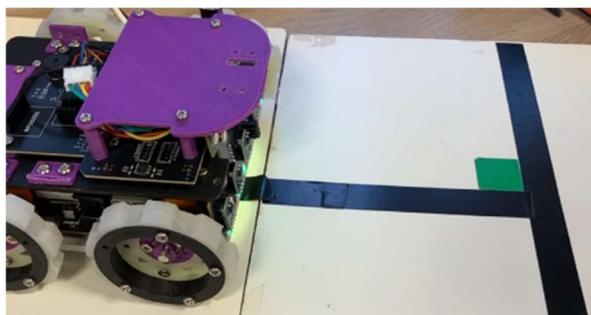
発注していた ToF モジュールが到着。



2024/11/10

大会前最後の休日なので b-lab (地域の施設) に集まって丸一日活動。

交差点制御を実装。



2024/11/11

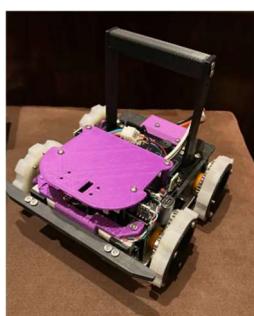
障害物回避を実装。

ライントレースのプログラムは完成。

レスキューはハード・ソフト共に間に合わないため断念。ライントレースで満点を取ることを目指す。

電源スイッチが押しにくいところにあったため、毎回しっかりと押せておらず、接触不良を起こしていることが発覚。スイッチを買いにいくことを決断。

ハンドルが付く。



2024/11/13

プレゼンポスターが完成。

2024/11/14

スタート/ストップスイッチを実装。

2024/11/15

スポンサー様のロゴのシールを貼付。



2024/11/16

東東京ノード大会参加。

【競技動画】

一走目：<https://youtu.be/mHmy6ow2aJI>

二走目：<https://youtu.be/Brlag7j284c?si=YrfkRYcvI0XdY4DP>

【競技結果】

順位：2位

一走目点数：286

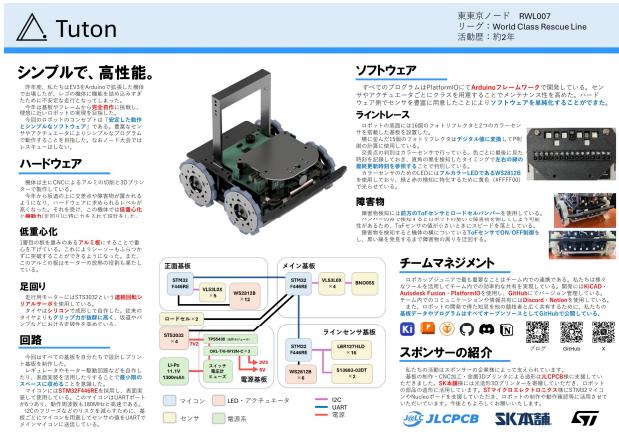
二走目点数：345

進行停止：8

時間：07:51

救助：0

ベストプレゼンテーション賞



「講評」

・一走目

緑の誤検知、障害物の回避の失敗を一度ずつてしまい、予定より二回多く進行停止をしてしまった。

もっと、閾値等の調整をするべきであった。細かい部分の調整が結果に大きくかかわることを学んだ。

・二走目

一走目で発覚した問題点を調整時間内に修正することができたため、ライントレースを完璧にこなすことができた。

・その他

他のチーム(特に BoingDGK, おしゃんてい)から機体の機構などを教えてもらい、大会でしか得られない学びを得ることができた。

3. 関東ブロック機体「ミケ」開発期

2024/11/17

大会翌日に反省会(12 時間)を行い以下のことを決めた。

【関東の目標】

- ライントレースで 2 回満点を取る。

【基板】

- ラインセンサ基板：下側にはみ出さないようにする。
- メイン基板：STM32F446 を使う。
- 正面基板：ToF とネオピクセルのみ搭載する。カメラは 2 層目に置く
- ターミナル基板：サーボモータをまとめる。STM32F446 を使う。
- 電源基板：F446 を使う。かごの後ろにのせる。スペーサを通してメイン基板に電源送る

【ハード】

- タイヤのギア部分を変更。
- 一層目の設計を基板の形状変更に伴い変更。
- インサートナットを活用し整備性向上を図る。
- かごにリンク機構を採用しモータ数を削減。
- ねじを数本外せば、一層目と二層目を分離させることのできる構造にする。

【ソフト】

- 低レイヤのドキュメント管理を行う
- 自作機レスキュー担当光本
- 自作機ライントレース担当五十嵐（たぶん移行してちょっといじるだけ）
- レゴ予備機プログラム担当五十嵐

【レスキュー】

- かごには2つに入る設計にする
- 先に銀を優先していれる
- 捜索にカメラで探して拾う&捨てる

2024/11/18

ノード大会でポスター賞を獲得していることが判明。

関東ブロック大会への推薦をいただく。

他チームのポスターを読み込み、機構などを学んだ。

2024/11/21

YouTubeにノード大会の走行風景を公開。

[ブログ](#)で、当日の様子や反省点について公開。

2024/11/25

バッテリーケースに磁石を用いることを決定した。

2024/12/01

関東大会に向けて車輪のギアを発注した。

2024/12/08

関東大会でのチーム番号がRWL011だということが判明した。

2024/12/10

発注していたパーツが届いた。



[ブログ](#)で発注した物に関する解説記事を書いた。

2024/12/19

関東大会用の基板(メイン基板、ライン基板、電源基板)を発注した。

2024/12/24

レスキューのアーム用の自作サーボホーンを3Dプリンターで作成した。

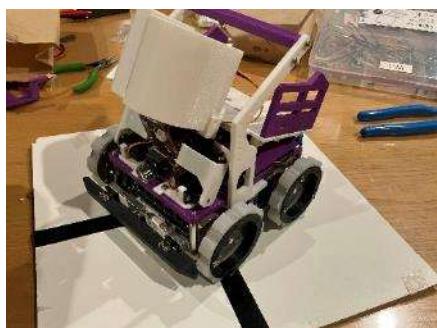


2024/12/31

発注していた三種類の基板が届く。はんだ付け・組み立てを行う。

2025/01/01

はんだ付けを行った基板をロボット本体に組み込んだ。また、全体の動作確認を行った。

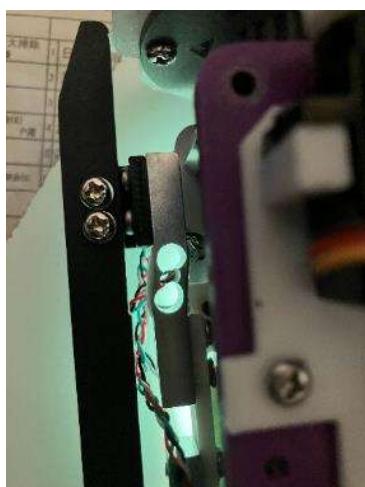


2025/01/02

P制御によるライントレースをすることができた。

2025/01/06

ロードセルがひしゃげた。予備パーツに交換。



2025/01/09

ロボットがレスキューゾーンの四隅にいた時でもかごを開けるように、かごの形を改善した。



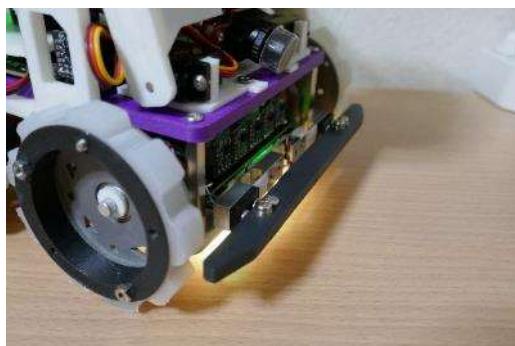
2025/01/10

関東大会用のポスターが完成。



2025/01/11

大会前日の追い込みのために、チームで集まって作業をした。



2025/01/12

ロボカップジュニア関東大会に出場。

【競技動画】

1走目：<https://youtu.be/uhHvAZY3oSU>

2走目：https://youtu.be/EKjE-MT_gL4

【競技結果】

順位：2位

点数：一走目 389 点 二走目 420 点

(レスキュー0個)

タイム：08：34 (二回合わせて)

【講評】

・ライントレースについて

一走目で、縁を誤検知してしまったことにより、ライントレースで満点を取るという目標は叶わなかった。

二走目では、閾値を改善しライントレースを完璧にこなすことができた。全体的に安定していたが調整が足りなかった。

・レスキューについて

正面に搭載されている ToF センサが、バンパーと干渉してしまい被災者の正確な位置を把握することができず、ボールを拾うことができなかった。前回大会とは異なり今回はレスキュー機構を搭載していたにも関わらず、うまくそれを活用することができなかった。この経験は次回に必ず生かさなければならない。

4. ジャパンオープン機体「シロ」開発期

2025/01/13

大会翌日に前大会同様に反省会(12 時間)を行い以下のことを決めた。

【全国の目標】

- ・レスキューをする。満点走行をして優勝する。

【ハードウェア】

- ・着脱可能で整備性の良いレスキュー機構を次期機体に向け製作
- ・全体的に整備性を向上させ、機能ごとにパーツをユニット化し機体が壊れた際の対応が行いやすい機構を考える
- ・閾値調整等に使えるディスプレイを搭載することも決定

【回路】

- ・全国大会に向けて、前回基板のミスを修正し小型化を行った。
- ・整備性向上のために、スペーサに 11.1V を流す機構を本格採用。
- ・レスキュー用のカメラを OpenMV から変更する場合の代替え案の提示

【ソフトウェア】

- ・すべての被災者検知に、機械学習を用いるのではなくアルゴリズムを用いて性能の低いマイコンでも実行できるようなアルゴリズムの作成準備を行う。
- ・それによる銀の検知率上昇を期待

2025/01/22

関東大会の動画をアップロード。[ブログ \(https://tuton-rcj.jp/20250122/\)](https://tuton-rcj.jp/20250122/) の記事を更新。

2025/01/27

開発につかれたメンバーがメイズ世界大会出場チームを分析し始めた。

2025/01/28

関東ブロックから全国大会への推薦チームが 2 チームであったことが判明し、ほかのブロックに強豪がいることを察知し焦る。

2025/01/30

ローカル環境でロボカップジュニアの CMS を動かせることを発見。練習の際に活用するために回路担当が頑張って導入した。

2025/02/01

3 日前にローカルに導入した CMS をさらに活用するために、OCI 上に CMS 環境を構築した (<http://cms.tuton-rcj.jp/>)。これによってどの端末からでもアクセスできるようになった。



2025/02/06

全国大会に向けて基板を追加発注。



機体のほとんどが完成した。

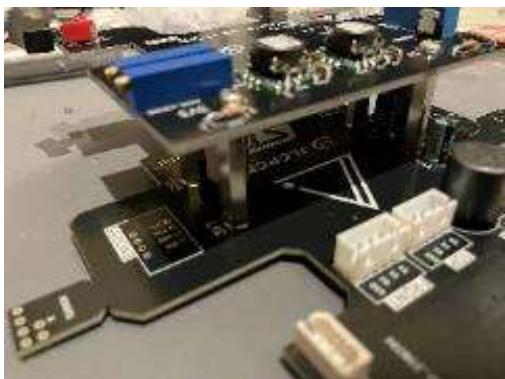
2025/02/14

ロボカップジュニアジャパンから被災者の作り方が発表される。

同時に同じものを制作するために必要なものを準備する。

2025/02/14

基板の動作確認を行った。



2025/02/20

ロボカップジュニアレスキュー競技実施要項を確認した。

また、ロボットを組み立てた。



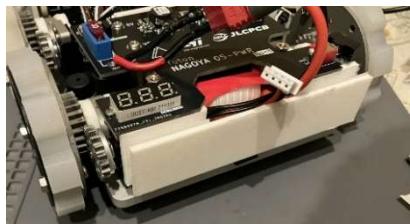
2025/02/21

機体を横向きにおいても倒れないことが判明。重心を低く作れていることが示された。



2025/02/25

バッテリーケースの蓋にマジックテープを採用した。



2025/03/07

被災者の部品が届く。被災者を作成。



2025/03/08

ある程度のレベルのライントレースができるようになった。

2025/03/16

進捗を互いに発表しあった。

2025/03/17

OpenMV カメラとメイン基板との通信方法の決定・動作確認。

2025/03/19

ジャパンオープン機体の名前を 30 分ほどかけて「シロ」に決定。

ポスター・ドキュメントの完成、提出。