



Team Name

Munako Aegis

Our School

Munakata High School

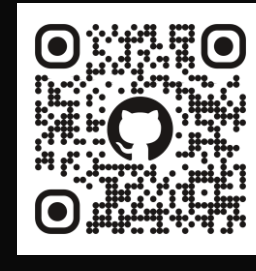


Region

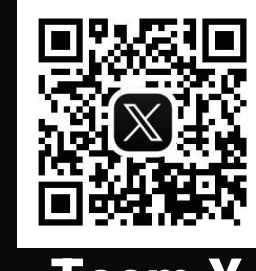
JAPAN

League

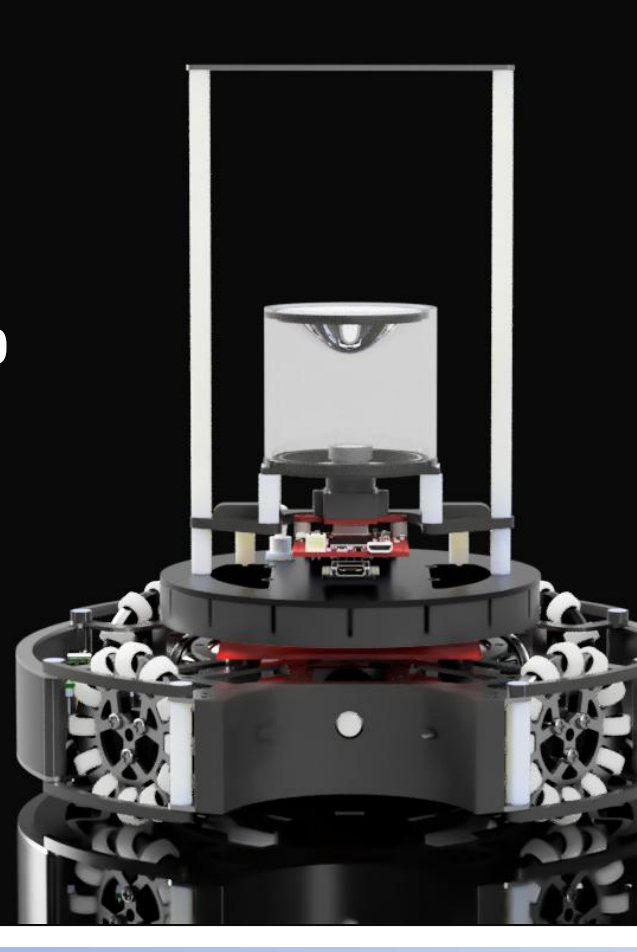
Soccer LWL



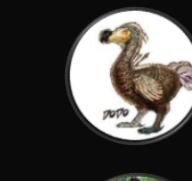
Team GitHub



Team X




Team Member Software



Koki Suehiro

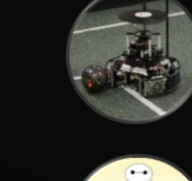
ゴールキーパープログラムマネージャー



Yuta Kurisaki

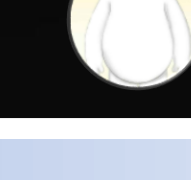
攻撃者プログラムマネージャー

Hardware



Umi Fujita

航空機および回路設計マネージャー



Maiki Kimura

回路設計、メディアマネージャー

Abstract

私たちは、ライト級リーグに出場する高校生4名からなるチーム「ムナコイージス」です。2023年シーズンより活動を開始し、日々のロボット製作を通して着実に技術を磨いてきました。活動開始当初は、特別なスキルを持ったメンバーは皆無でしたが、メカ設計、プログラミング、回路設計に日々取り組み、試行錯誤を繰り返しながら知識と経験を積み重ねてきました。初年度は予選会で1位を奪取し、ジャパンオープンへの出場権を獲得しました。しかし、本戦では64チーム中31位という残念な結果に終わりました。それでも諦めずに努力を続け、今年はジャパンオープンで優勝を果たしました。さらに、ベストプレゼンテーション賞も受賞しました。

メカ設計にはAutodesk Fusion、回路設計にはKiCadを主に使用しています。製作には3DプリンターとCNCフライス盤を活用し、効率的かつ高精度なロボット開発を行っています。プログラムはVisual Studio Codeで記述し、マイコン制御にはC++、カメラ制御にはMicro Pythonを使用し、Open MV IDEで開発しています。ロボット本体には経費かつ高い耐久性を実現するため、ジュラルミンを採用しています。バッテリーボックスやサイドカバーなどの外装部品は3Dプリンターを用いて自社製作することで、設計の自由度を高めています。回路基板は、スポンサーであるJLPCPBに高精度な製作を依頼し、安定した回路構成を実現しました。カメラシステムには、強化ビニルを熱圧着して自社製作した全方位ミラーを採用しました。これにより、1台のカメラで360度の視野を確保しながら、大幅なコスト削減を実現しています。さらに、TAKAHARA社から提供されたレンズドを用いてキッカー機構を搭載しています。

近年多くのチームが悩まされている「ラインアウト」の問題に対処するため、速度制御機能を搭載しました。ロボットは移動中にリアルタイムで速度を計測し、コートの位置に応じて最適な速度で走行することで、最高速でのラインアウトを防止します。さらに、オムニホイールは二重構造を採用しています。二重構造により床面との接地面積が拡大し、ブレーキ性能が向上しています。

ラインセンサーには赤色LEDを採用し、応答速度と精度を向上させました。これらの工夫により、ラインエリアの強化が図られています。

ボール検出には、限られたピン数でも複数の入力処理できるマルチプレクサを採用し、移動平均処理によってノイズの多い値を安定化しています。さらに、LCDモニターを搭載することで、PCに接続することなく、様々な値をリアルタイムで確認できます。

世界選手権に向けて、ロボットの設計を見直ししました。これまでアタッカー用とゴールキーパー用で異なっていたロボットの形状を統一しました。両方の役割の利点を兼ね備えた2体の同一ロボットを採用することで、試合中に即座に役割を切り替えることが可能になりました。さらに、センサーや基板も全て標準化することで、部品管理の効率化と互換性の向上を実現しています。私たちの活動はX（旧Twitter）でフォローできます。また、GitHubでもすべてのデータを公開しています。

チーム活動

日々の積み重ねと計画が大切

私たちのチームは昨シーズンよりロボカップサッカーライト級リーグに参加しており、今年で2年目となります。チームは4名で構成されており、高校時代からロボット競技に携わってきたメンバーが3名、中学時代から1名います。設立からまだ日が浅いチームですが、「世界選手権優勝」という大きな目標を掲げ、日々ロボット製作と技術力の向上に励んでいます。限られた時間の中で、メンバー一人ひとりが着実に成長できるよう、明確な活動方針を定めました。本発表では、どのような方針を掲げ、どのように活動を進めてきたのかを紹介します。

1年目 - H1	1年目 - 後半	2年目 - H1	2年目 - H2	世界大会準備	最終目標
各チームメンバーは独自のロボットを製作し、その構造や仕組みについての理解を深めました。	ロボットの製作プロセスを通して、私たちは自分たちの強みと弱みを認識し、それに応じてチームの役割を割り当てました。	ジャパンオープンでの経験を振り返り、実現したい仕組みを議論し、必要な技術を検討し、計画を立て、一つ一つのステップを明確にすることで、着実に技術力を高めていきました。	これまでの試合経験を踏まえ、今後さらに導入したい仕組みや戦略について話し合い、チームとして今後の大会に向けてどのような準備が必要かを話し合いました。	ジャパンオープンでの経験をもとに、世界選手権の環境に適応するために何を必要があるかを話し合いました。	世界選手権で優勝

私たちは技術力を高めるために、膨大な時間を費やし、たいゆめ努力を重ねてきました。週末は午前9時から午後5時まで、平日は放課後の午後4時から7時まで活動しています。そのグラフは、ロボットの開発に費やした時間を月ごとに累計で表したものです。縦軸はロボットの開発にかかった時間（分）、横軸は日付の推移を表しています。このグラフは、限られた時間の中で世界を目指し、たいゆめ努力を続けてきた私たちの記録です。

私たちの総活動時間は約18万分、つまり約3,140時間に及びます。

この膨大な時間は、「毎日の小さな努力が、やがて大きな成果につながる」という私たちの強い信念の証です。私たちは、この道のりを通して、ロボットを作るだけでなく、チームとして粘り強く取り組む力そのものを育んできました。私たちは失敗と成功の両方を自らの手で経験したからこそ、継続的な努力が成長と成果につながり、最終的にはより強力なロボットを作ることにつながることを真に理解するようになりました。


方法 / 製造 / デザイン

ロボットのコンセプト

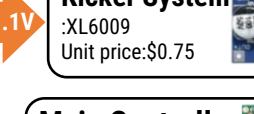
今シーズンは「力強く、正確に動くロボット」というコンセプトに着目し、センサー、構造設計、駆動性能の改良に取り組んでいます。機軸センサーでは、Open MVカメラと全方位ミラーを組み合わせたことで、1台のカメラでコート全体を認識できるようになりました。ボール検出には、16個のボールセンサーを円形に配置するシステムを採用し、より正確な角度検出を実現しました。特に、ボールセンサーシステムは、マルチプレクサを用いることでマイコンのピン数制限を解消しました。また、攻撃性能も大幅に向上しました。新設計のキッカーデバイスと、3Dプリンターで製作したボール形状に合わせたキャプチャーゾーンにより、キックミスの発生率を低減しました。これにより、シュートの安定性と威力が飛躍的に向上し、安定した前方へのボール供給が可能になりました。その結果、「確実にゴールを決められるロボット」としての性能向上を実現しました。駆動系にはDIGILENT IG22 1/19モーターを採用し、ロボットの制御性とパワを両立させました。さらに、制御機構から発生した二重オムニホイールは、スムーズな動きを実現し、試合中にロボットの修理やデバッグを容易に行えることも重視しました。搭載されたOLEDディスプレイにより、リアルタイムのセンサーデータを確認し、迅速なトラブルシューティングと調整が可能になり、PC接続が不要になり、メンテナンス性が大幅に向上しました。メインマイコンには、高速通信と豊富なI/O拡張性を備えたTensy 4.0を採用。さらに、フラットケーブルやLEDの採用、ノイズ対策を考慮した配線設計などにより、メンテナンス性と安定性を重視しました。また、従来のアタッカーロボットとゴールキーパーロボットを別々に製作していたが、今回から両ロボットでセンサーや基板をすべて共有することで、部品管理の簡素化と高い互換性を実現しました。

Robot's Structure

Battery




Kicker System



XL6009

Unit price:\$0.75


Solenoid




CB10037 x1

Unit price:\$19.6

Debug System




O-LED Display



SSD1306

Unit price:\$0.97

Voltage Conversion Circuit

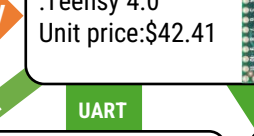


Using IC: OKL-T76

11.1V to 5.0V

Unit price:\$6.03


Main Controller



TEENSY 4.0

Unit price:\$42.41


Gyro Sensor



AE-BNO055-80

Unit price:\$3.29

Motor Driver



DAISIN OSDR3002


IC:VNH2SP3002

0.3V ~ 16.0V

15~30A

Unit price:\$135.69

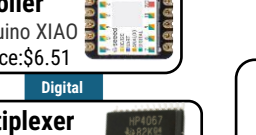
Sub Controller



Seeeduno XIAO

Unit price:\$6.51

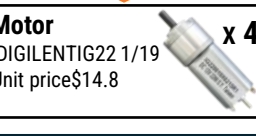
Sub Controller



Seeeduno XIAO

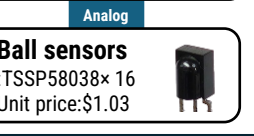
Unit price:\$6.51

Multiplexer




Unit price:\$1.34

Multiplexer



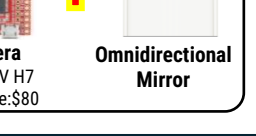
Unit price:\$1.34

Tactile Switch x3



Unit price:\$0.1

Vision System




Camera

OpenMV H7


Unit price:\$80

Omnidirectional Mirror



Unit price:\$80


Motor



DIGILENTIG22 1/19

Unit price:\$14.8


Ball sensors



TSSP30328-16

Unit price:\$1.03

Line Sensors



NJ17502


Unit price:\$0.34

ロボットの努力

道具

私たちのチームは主に以下のソフトウェアを使用しています


Design



KiCad

回路製造に使用


Program



Visual Studio

C++によるプログラミング (カメラを除く) に使用


Management



GitHub

データ保存に使用

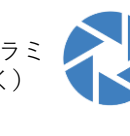
Autodesk Fusion



Autodesk Fusion

ロボットのフレームとコンポーネントの設計に使用されます

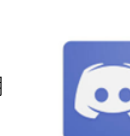
OpenMV IDE



OpenMV IDE

Pythonでカメラをプログラミングするために使用

Discord



Discord

チームメンバー間で情報を共有するために使用されます

01 Line Sensor

今シーズンのロボットは、円形ラインセンサーと外側ラインセンサーを組み合わせた、より高精度なライン検出システムを構築しています。まず、ラインセンサーには従来よりも多くのLEDが搭載されています。これにより、ロボットは外側はより広い範囲の白線を正確に検出できます。センサーの読み取りにはSeeeduno XIAOが使用されていますが、当初はポート数が限られていたため、搭載できるセンサーの数が限られていました。マルチプレクサを導入することでセンサーの数を増やすことができました。このシステムでは、特にゴールキーパーロボットに効果を見られています。このシステムにより、プレー中は常に白線を正確に追跡し、安定した動作を維持できます。円形ラインセンサーは、コースの方向を決定する上で重要な役割を果たします。これを確実に検出することで、ロボットは常に正しい軌道を維持することができます。一方、外側ラインセンサーは、ロボットがコースから外れないように修正するための基準となります。さらに、全国大会に出場したロボットは、フィールド上の白線から外れてしまうことが何回もありました。そこで、ラインセンサーの配置を見直し、ロボット下部に存在する「無反応死角」を削減することに成功しました。これにより、ライン検出精度がさらに向上し、動作の安定性と精度も向上しました。以前は、1個500円のフォトトランジスタを導入しましたが、コスト削減のため、1個50円のフォトトランジスタを導入しました。

02 IRボード

これまで、Seeeduno XIAOはピン数の制限により、最大8個のボールセンサーしか搭載することができませんでした。この構成では角度検出は可能でしたが、センサー間の間隔が広いため、検出精度が低く、誤差が発生していました。この問題を解決するために、マルチプレクサを導入しました。この技術革新により、16個のボールセンサーを搭載することが可能になりました。センサー数が増加したことで、ボールセンサー間の距離が狭まり、より高精度な角度検出が可能になりました。

03 デバッグシステム

ロボットにO-LEDディスプレイを搭載することで、センサーからのデータをリアルタイムに把握できるようになりました。これにより、ロボットや周囲の状態を素早く把握でき、戦術の即座の変更や、問題発生時の対応をより迅速かつ柔軟に行うことができます。従来、センサーデータの確認にはロボットを停止し、PCに接続する必要がありましたが、O-LEDディスプレイの導入により、その時間と労力が大幅に削減されました。これは、特に試合時間が限られており、現場での判断と対応のスピードが求められる場合において、大きなメリットとなります。また、OLEDディスプレイは現場でのデバッグにも活用でき、開発効率の向上にも貢献します。

04 メインボード

メインボードはTensy 4.0を採用し、7つのUART、40のデジタルピン、14のアナログピンと豊富なピン数を備えています。メインボードの設計では、フラットケーブルを使用することでケーブル数を削減しています。また、配線設計では線路を使用することでノイズの影響を低減し、信号の安定性とロボットの全体的な動作信頼性を高めています。ボード上には通電状態を視覚的に確認するためのLEDが備えられており、入ミスを防止します。接続ポイントを明確に示すことで、配線ミスを防ぐだけでなく、プログラミングワークフローの効率化にも貢献します。当社のロボットは、2種類のジャイロセンサーを搭載するように設計されています。これは、1つのセンサーが故障した場合、プログラムを置き換えることで即座に他のセンサーに切り替えることができます。このバックアップ設計により、競技中にセンサーが故障しても安定した動作を維持できます。

05 パワーボードとキッカーボード

従来、電源基板とキッカー基板は別々に製造されていましたが、今シーズンから1枚の基板に統合することで省スペース化と配線の簡素化を実現しました。これにより、ロボットの内部構造がコンパクトになり、組み立てや部品交換の効率が大幅に向上しました。電源基板はDC-DCコンバータを使用し、各モジュールに安定した電力を供給するように設計されています。これにより、動作中の電圧変動が最小限に抑えられ、システム全体の安定性が向上します。キッカー回路には昇圧回路も組み込まれており、安定した高電圧（48V）を出力するように設計されています。

06 モーターとオムニホイール

今シーズンのロボットは、カメラシステムにいくつかの改良が加えられています。まず、ロボットにはDIGILENT IG22 1/19高精度モーターが4つ搭載されており、より高いトルクとより安定した動作性能を提供します。これらのモーターは、4CHモーターコントロールer (DSR1202)によって制御されます。オムニホイールの構造も改良され、ロボットの安定性と制御性が向上しました。従来の1層構造から2層構造に変更することでグリップが向上し、複雑な動きや細かい制御が必要な状況でも、ロボットは安定した走行性能を維持できるようになりました。さらに、オムニホイールの駆動は以前のものよりも大きく、速度を上げることができました。オムニホイールの内側部分は3Dプリンターで製作され、外側部分はジュラルミンで作られています。これにより、大量生産が容易になり、破損しにくくなりました。

07 ビジョンシステム

Open MV H7を単体で使用する場合、視野角が約70度に制限されており、ロボットが同時に周囲全方向を把握するには不十分です。そこで、Open MV H7に全方位ミラーを組み合わせたことで、ロボットが360度の全方位画像を取得できるシステムを構築しました。全方位ミラーは、加熱して柔らかくしたポリ塩化ビニル（PVC）製のもので、コンピュータ上で設計した理想形状に合わせて3Dプリンターで作成した双曲面の上に圧着（ヒートプレス）します。これにより、ロボットの周囲の様子をカメラに映し出すことができ、1台のカメラで全方位の情報を取得することができました。

08 スタンドアロンカメラ

この構成により、ロボットは1台のカメラで常に周囲を監視し、ゴールの位置やコートセンターの方向といった情報をリアルタイムに取得・処理することが可能になりました。従来よりも少ないセンサー数で、より広範囲かつ均等な環境を認識できるようになり、常にゴール方向を向く姿勢制御やライン制御に大きく貢献します。これらのプログラムはPythonを用いています。

09 ボールキャプチャーゾーンとソレノイド

今シーズン、ロボットには新しいキッカー（CB1037）が3つ搭載されています。このキッカーデバイスの性能を最大限に引き出すため、キッカーデバイスの力をボールに与えることなく伝える方法を考案しました。特に重要だったのは、キックした際にボールをしっかりとキャッチすることでした。以前は、ボールをキャッチできても、キッカーデバイスの機構とうまく噛み合わず、シュートすることができませんでした。この問題を解決するため、3Dプリンターを活用し、ボールの形状にぴったりフィットする「ボールキャプチャーゾーン」を作成しました。これにより、ボールを確実にキャプチャーできるようになりました。この設計変更は、キックの成功率を大幅に向上させ、従来のキャプチャーの形状を最適化することで、キックの力がよりしっかりとボールに伝わり、ボールはより正確な角度で飛ぶようになり、全国大会での対戦相手との接戦でロボットが優位に立つようになりました。

09 フレーム

ロボットの土台となるフレームは、ジュラルミン（アルミニウム合金）を用いて製作されています。ジュラルミンは航空機にも使用される、強度と軽量性を兼ね備えた金属です。その特性を活かし、軽量性と軽量化にも徹底的にこだわりました。不要な部分を削り出すことで、必要な強度を保ちつつ軽量化を実現し、強度を保ちながら軽量化する構造を実現しました。フレームの設計にはFusion 360を使用し、実際の切削加工はクラブ内のCNC工機で完了しました。これにより、設計と試作を短いサイクルで繰り返すことで、最適なフレーム構造を導き出すことができました。昨シーズンから1年間の試行錯誤を経て、ついにこのフレームにたどり着きました。

データ / 結果 / 考察

ラインコントロールの向上につながるスピードコントロール

昨シーズンのロボットに使用したモーターは6ボルトでしたが、速度と性能に限界がありました。そこで、より性能の高いモーターを探そうとしました。コストを可能な限り抑えるため、12ボルトモーターを選択しました。その結果、以下のモーターオプションを選択しました。

・DCギヤードモーター 12V 1:19 (DIGILENT IG22 1/19) (約19ドル) 2024年6月

このモーターは私が求めていた性能を備えており、良い選択だと思いました。しかし、このモーターは速度を上げましたが、境界外が増えました。

つまり、ロボットが速くても境界外になってしまえば意味がありません。以前は境界外を防ぐためにモーターの速度を落としましたが、そのままだではモーターが加速し続けるという問題がありました。この問題を解決するため、BNO055を用いてロボットの速度を測定することになりました。これにより、コートの境界線付近ではロボットの速度を落とし、アウトオブバウンズを防ぎながら、コート中央ではロボットが最高速度に達することが可能になりました。

実験環境

検証1および検証2の実験では、ロボットをコートの端から端まで走行させ、記録タイマーを用いて0.1秒ごとに移動距離を計測しました。これらの計測値に基づいて、0.1秒間隔ごとの平均速度を算出し、速度の変化を示すグラフを作成しました。（写真1および2参照）

写真1 実験風景

写真2 実験風景

図1 速度制御の仕組み

速度制御

設定速度を決定します。測定された速度が設定速度を下回っている場合は、モーター出力を上げます。逆に、速度が設定速度を超えている場合は、モーター出力を下げます。この方法を「速度制御」と呼ばれます。（図1参照）この方法が境界外への後進防止に有効かどうかを評価するために、以下の検証を行います。

検証1: 速度制御の有無による違いの調査

速度制御が正しく機能していることを確認するために、速度制御ありとなし

の両方で実験を行います。次に、速度（v: cm/s）と時間（t: 秒）の関係をグラフで比較しました。

結果:

グラフは、速度制御ありとなしを比較すると、速度が 80 cm/s で一定であることを示しています。

移動平均を用いたセンサー読み取り値の平滑化

ボールセンサーからのアナログ値には若干の変動がありました。これを軽減するために、センサーの読み取り値に移動平均フィルタを適用しました。ボールをセンサーから遠ざけ、読み取り値の変化を観察しました。移動平均のサンプル数を変えて変動がどのように変化するかを分析し、最終的に10個のサンプルを選択しました。下のグラフは、移動平均を適用した場合と適用しない場合の結果を比較したものです。

移動平均フィルタの効果

結果: グラフから、移動平均フィルタを適用するとセンサーの読み取り値がより滑らかになることがわかります。この結果に基づき、ボールセンサーだけでなく、ラインセンサーのアナログ値にも移動平均を適用しました。

検証2: 速度制御による減速動作への影響

グラフは、2つのケースにおける速度（v: cm/s）と時間（t: 秒）の関係を比較したものです。最初のケースでは、速度制御を用いてロボットの速度を設定速度まで加速し、その後設定速度を下げた減速させました。2つ目のケースでは、速度制御を行わず、一定時間PWM値を固定してロボットを動作させ、その後PWM値を下げて減速させました。

結果:

図2 減速時の速度制御の効果

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を越えたり壁にぶつかったりすることなく停止できる最大速度を求めることです。ラインを踏んだ際にブレーキをかけるプログラムを作成し、設定速度を変化させ、境界を越えない最大速度を記録しました。

結果:

検証3: 境界を越えない速度の最大値を求める

目標は、ロボットがラインを踏んだ際に境界を