

M5GO を用いた倒立振子の試作と自動起立への挑戦

1. はじめに

倒立振子は、車体の傾きをセンサで測定し、モータを制御することで倒立状態を保つ装置です。倒立振子を製作することで、センサによる計測やモータ制御、フィードバック制御などを試すことができます。

近年では、マイコンやセンサを比較的安価に入手できるようになりました。また、生成 AI を利用することで、マイコンのプログラムも作成しやすくなっています。

そこで今回は、M5GO というマイコンとモータを使って倒立振子を試作しました。また、プログラムの作成や制御方法の検討に生成 AI を利用しました。

2. 試作した倒立振子

試作した倒立振子を図1に示します。

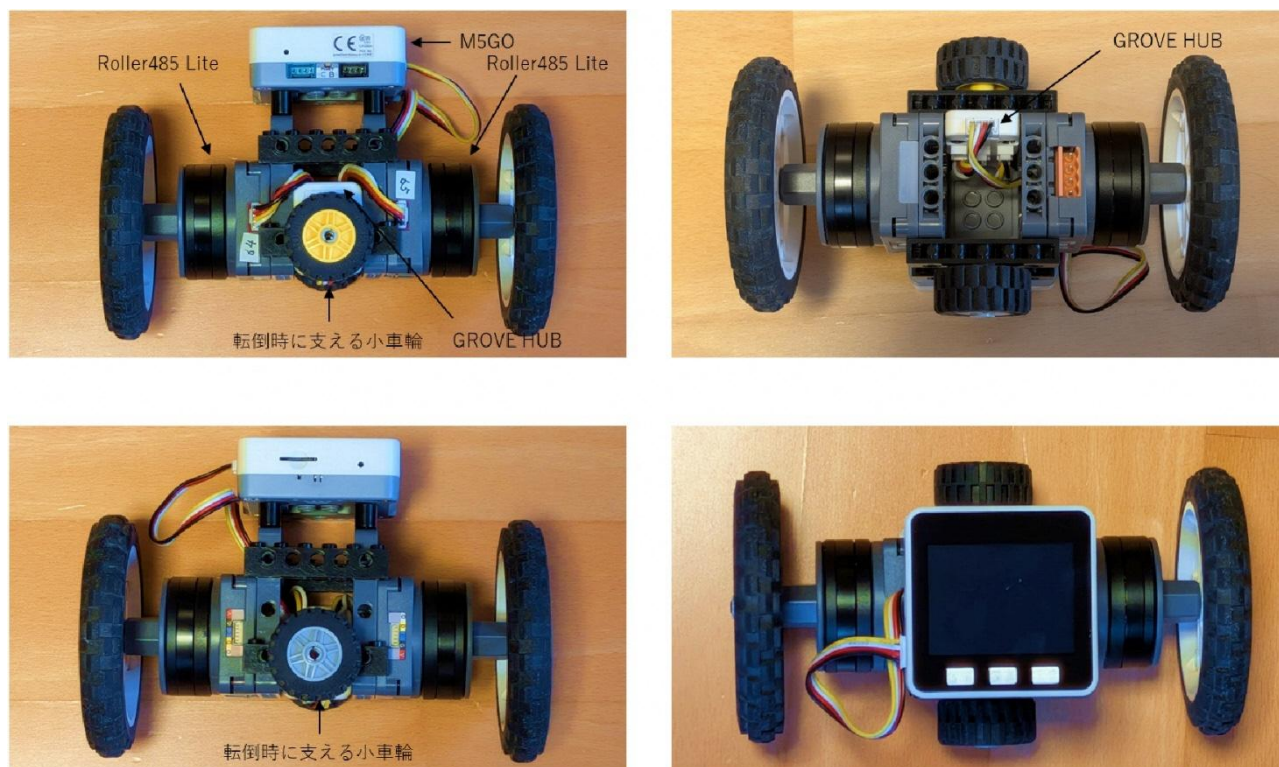


図1 試作した倒立振子

マイコンには M5GO、モータには Roller485 Lite を 2 台使用しました。左右の Roller485 Lite は GROVE HUB を介して M5GO に接続しました。車体にはブロックを使用し、転倒時に車体を支えるための小車輪を取り付けました。車輪の直径は約 80 mm、車体の質量は約 330 g です。

車体の傾きは M5GO に内蔵されている IMU（慣性計測装置）を使って測定しました。IMU には加速度センサとジャイロセンサが搭載されています。加速度センサから車体の傾斜角を求め、ジャイロセンサから角速度を取得しました。今回は両者を相補フィルタによって組み合わせて車体の傾斜角を求めました。制御周期は 5 ms（200 Hz）としました。

3. 倒立制御

倒立制御には PD 制御を使用しました。

PD 制御では車体の傾斜角と角速度からモータの出力を求めます。車体が傾いた場合には元の位置に戻す方向にモータを動かし、角速度が大きい場合にはその動きを抑える方向に制御します。今回の機体では、実際に倒立させながら制御パラメータを調整しました。本試作機では、比例ゲイン $K_p=4000$ 、微分ゲイン $K_d=100$ としました。

図 2 に倒立している様子を示します。



図 2 倒立制御中の倒立振り子

最初に Roller485 Lite の速度制御を使ってモータを動かしました。この場合でも倒立しましたが、倒立中に細かな振動が見られました。

そこで、速度制御から電流制御に変更しました。電流制御では、PD 制御で求めた値をモータの電流指令値として使用しました。その結果、速度制御の場合に比べて振動が小さくなり、静かに倒立するようになりました。

倒立中の車体を指で軽く押してみました。車体は一旦傾きますが、その後、車輪が動いて倒立状態に戻ることを確認しました。

今回の機体では、モータの制御方法によって倒立時の挙動が異なることが分かりました。

4. 倒れた状態から起き上がらせる

倒立できることを確認した後、倒れた状態から自動的に起き上がらせることも試しました。

今回の機体は、モータを停止すると約 30° 傾いた状態になります。そこで、モータに一定の電流を与えて車体を起こし、直立付近まで来たところで PD 制御に切り替えました。

しかし、この方法ではうまく倒立できませんでした。車体を起こすためにはある程度大きな電流が必要です。一方、電流を大きくすると直立付近でも大きな角速度が残り、そのまま反対側へ倒れる場合があります。また、起立電流を小さくすると、車体が動き出さない場合があります。このため、単に車体の角度だけを見て倒立制御へ切り替えることは難しいことが分かりました。

5. 起立時にブレーキをかける

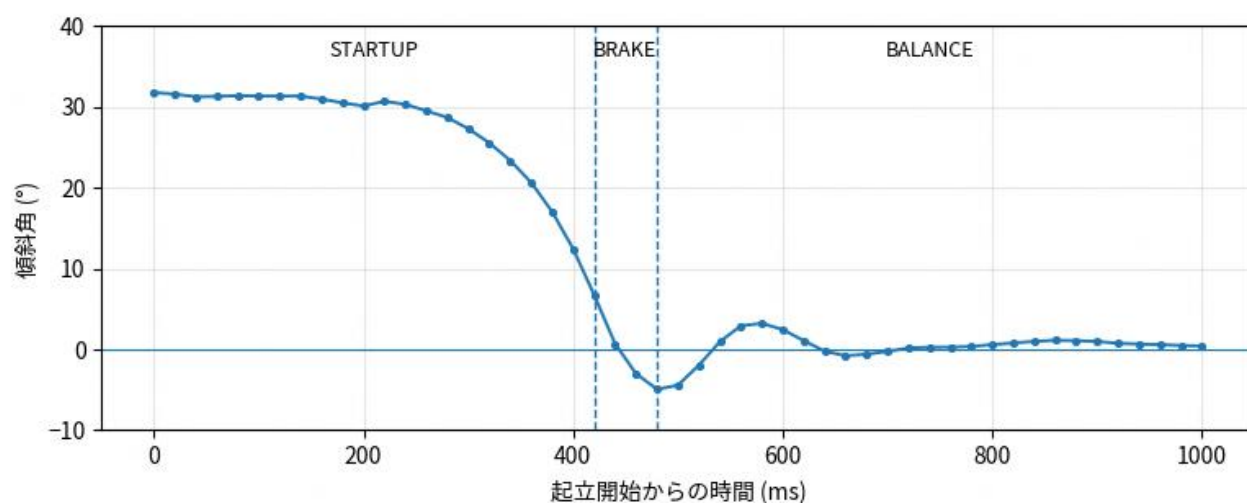
そこで、起立制御と倒立制御の間に、車体の回転を減速させる処理を追加しました。

制御を次の三つの状態に分けました。

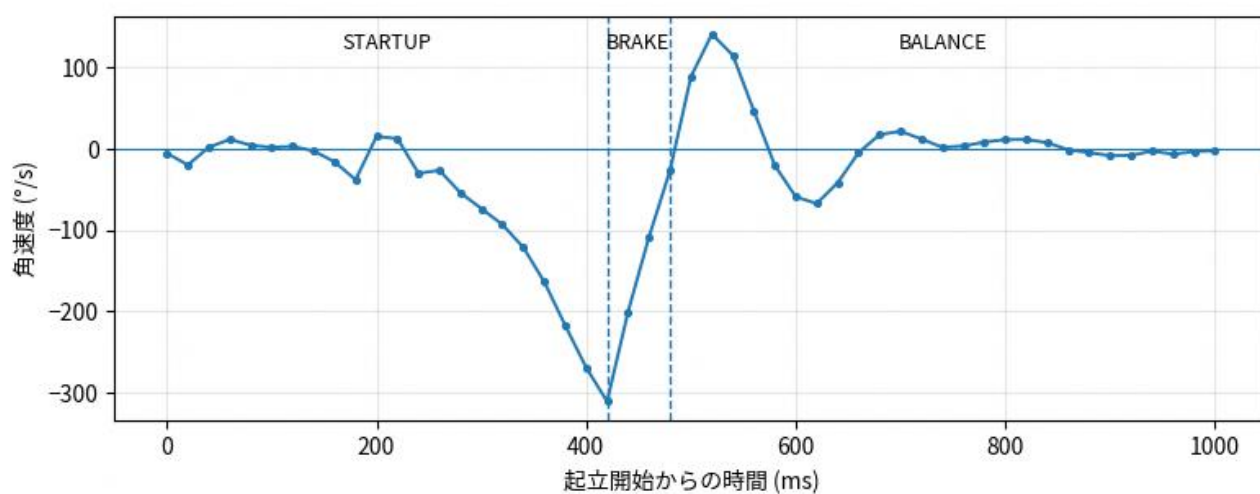
STARTUP (起立) → BRAKE (減速) → BALANCE (倒立制御)

STARTUP では、モータに一定の電流を与えて車体を起こします。傾斜角が 12° になったら BRAKE に切り替え、ジャイロセンサで測定した角速度と反対方向にモータのトルクを発生させました。角度と角速度が小さくなったところで BALANCE に切り替え、通常の PD 制御を行います。

図 3 に起立時の傾斜角と角速度の変化を示します。STARTUP では、車体が直立位置に近づくにつれて角速度が大きくなりました。そこで BRAKE に切り替えてモータに逆方向のトルクを与えました。その結果、角速度が低下し、BALANCE へ移行して倒立させることができました。



(a) 傾斜角



(b) 角速度

図3 起立時の傾斜角と角速度

6. 自動起立を試してみて

起立時の電流などを変更しながら実験した結果、倒れ角がそれほど大きくない場合には、倒れた状態から起き上がり、そのまま倒立できることを確認しました。

しかし、毎回同じように起立できるわけではありませんでした。起立時の電流条件が小さい場合には、車体が動き出さないことがありました。反対に電流が大きい場合には、勢いがつきすぎて反対側へ倒れることがありました。最終的には、モータドライバの電流指令値（内部設定値）21000 として実験しました。

起立動作には、初期の倒れ角、車輪と床の状態、車体の重心位置などが影響していると考えられます。また、今回のプログラムでは、制御を開始する前に車体を手で直立させてキャリブレーションを行っています。キャリブレーション時の姿勢が完全な直立位置からずれていることも、

起立動作のばらつきに影響している可能性があります。

今回は、倒れ角がそれほど大きくなければ、倒れた状態から起き上がって倒立できることを確認したところまでとしました。

7. 生成 AI を使ったプログラム作成

今回のプログラムは、生成 AI を利用しながら作成しました。

生成 AI に使用するマイコン、モータ、IMU、制御方法などを入力し、プログラムを作成しました。その後、実機で動作を確認しながら修正しました。実際に試してみると、速度制御では細かな振動が発生したり、自動起立では直立位置を通り過ぎたりするなど、さまざまな問題が発生しました。そこで、シリアルモニタに傾斜角、角速度、モータの電流指令値などを出力し、その結果を生成 AI に入力して原因や修正方法を検討しました。この作業を繰り返しながら、制御条件を調整しました。

今回の試作では、生成 AI を利用することで比較的短時間でプログラムを作成することができました。その一方で、生成 AI が作成したプログラムをそのまま使用するだけでは、実機を安定して動作させることはできませんでした。実際の動作や測定値を確認しながら、プログラムや制御条件を修正する必要がありました。

8. まとめ

今回は、M5GO と Roller485 Lite を使って倒立振子を試作しました。IMU で車体の傾斜角と角速度を測定し、PD 制御を行うことで倒立させることができました。また、速度制御から電流制御に変更することで、倒立時の振動が小さくなることを確認しました。

さらに、起立時にブレーキ処理を追加することで、倒れた状態から起き上がって倒立させることもできました。ただし、初期の倒れ角や起立時の条件によっては、うまく起立できない場合もありました。

今回の試作では、生成 AI をプログラム作成や制御条件の検討に利用しました。生成 AI を利用する場合でも、実際の動作や測定値を確認しながら調整することが重要です。(検証実施年月：令和 8 年 9 月)。

問い合わせ：新潟県工業技術総合研究所
技術統括センター 斎藤 雄治
TEL：025-247-1320