

モータ制御 Closed-loop システム リファレンスガイド

RD022-RGUIDE-01-J

東芝デバイス&ストレージ株式会社

これは参考資料です。本資料での最終機器設計はしないでください。

目次

1. はじめに.....	4
2. 特長.....	5
3. 仕様.....	5
4. 外観.....	6
5. モータ接続.....	7
6. スイッチ.....	8
6.1. ポテンショメータ.....	8
7. MCD+MCU モード.....	9
7.1. MCD+MCU モード時ブロック図.....	9
7.2. ハードウェア接続.....	9
7.3. ジャンパ設定.....	10
7.4. Arduino 接続.....	11
7.5. LCD シールド.....	12
7.6. 使用方法.....	13
7.7. Arduino UNO 上で使用するソフトウェア.....	14
7.8. Arduino Sketch.....	14
7.8.1. インストール方法.....	15
7.9. Arduino UNO との接続.....	16
7.10. コードのアップロード.....	17
7.11. Arduino IDE のシリアルモニタ.....	18
7.12. シリアルモニタ上での Kp 値の変更方法.....	19
7.13. LCD シールド.....	20
7.13.1. その他シールド.....	20
7.14. MCD + MCU の場合のモータ速度制御.....	21
7.14.1. PID アルゴリズム.....	21
7.14.2. PID 制御.....	21
7.14.3. PI 調整手順.....	23
7.14.4. ソースコード.....	23

8. MCD+OP-AMP モード	24
8.1. MCD+Op-AMP モード時のブロック図.....	25
8.2. 使用方法	26
8.3. 操作手順	27
8.3.1. MCD + Op-Amp のハードウェア設定.....	27
8.3.2. ジャンパ設定.....	27
8.4. 速度制御部.....	28
8.4.1. パラメータ調整.....	28
8.5. 自動進角設定	29
9. 添付 BLDC モータ仕様	30

1. はじめに

本リファレンスガイドは、東芝製 BLDC モータドライバ IC TB6605FTG を搭載したモータ制御 Closed-loop システムの使用方法について記載しています。本ボードは、Arduino UNO 用のプラグインボード(シールド)として設計されています。本リファレンスガイドでは、TB6605FTG を早期に評価するためのハードウェアとソフトウェアのセットアップのためのガイドラインを提供しています。

本システムは下記の 2 つのモードでの使用が可能です。

1. MCD + MCU モード
2. MCD + Op-Amp モード

また、モータ制御 Closed-loop システムには下記が必要です。

- モータ制御リファレンスボード (Arduino 互換コネクタ装備)
* Arduino 本体は含まれておりません。
- リファレンス用ブラシレスモータ (付属)



図 1-1 モータ制御 Closed-loop システム

MCD+MCU モードで使用する際には、評価用に下記が必要です。

- Arduino MCU ボード (付属していません。市販品をご用意ください。)
- LCD モジュール (市販品)

2. 特長

- 3-Hall センサ BLDC モータドライバをサポート
- モータ制御機能: Start, Brake, CW/CCW
- ポテンシオメータで回転スピードを制御
- リファレンス用モータを添付
- 別モータ調整のため差し替え可能な抵抗とコンデンサを使用
- 2つの動作モードを用意
 1. MCD+MCU モード
 Arduino UNO のプラグインシールドとして使用可能
 PID 制御をソフトウェアで実現し、Closed-loop 制御を実装
 サンプルプログラム(Sketch) を提供
 2. MCD + Op-Amp モード
 Closed-loop 制御を Op-amp を使用して実装

3. 仕様

表 3-1 仕様

項目	説明
TB6605FTG	正弦波 BLDC モータ制御
電源電圧	10V – 28V DC
添付モータ仕様	中速モータ 10V – 24V/ 0.5A – 10A/ 出力 26W/ 4000rpm
モータドライバ出力 + MosFET	N ch+ Nch MosFET (Toshiba TK20P04M1)
メイン IC	TB6605FTG, TC75S5 1 F, TK20P04M1 (MCD+MCU モードでは Arduino UNO を使用)
MosFET 仕様およびパッケージ情報	TK20P04M1: NMOS, 40V/20A, DPAK, R _{dson} =19mΩ

4. 外観

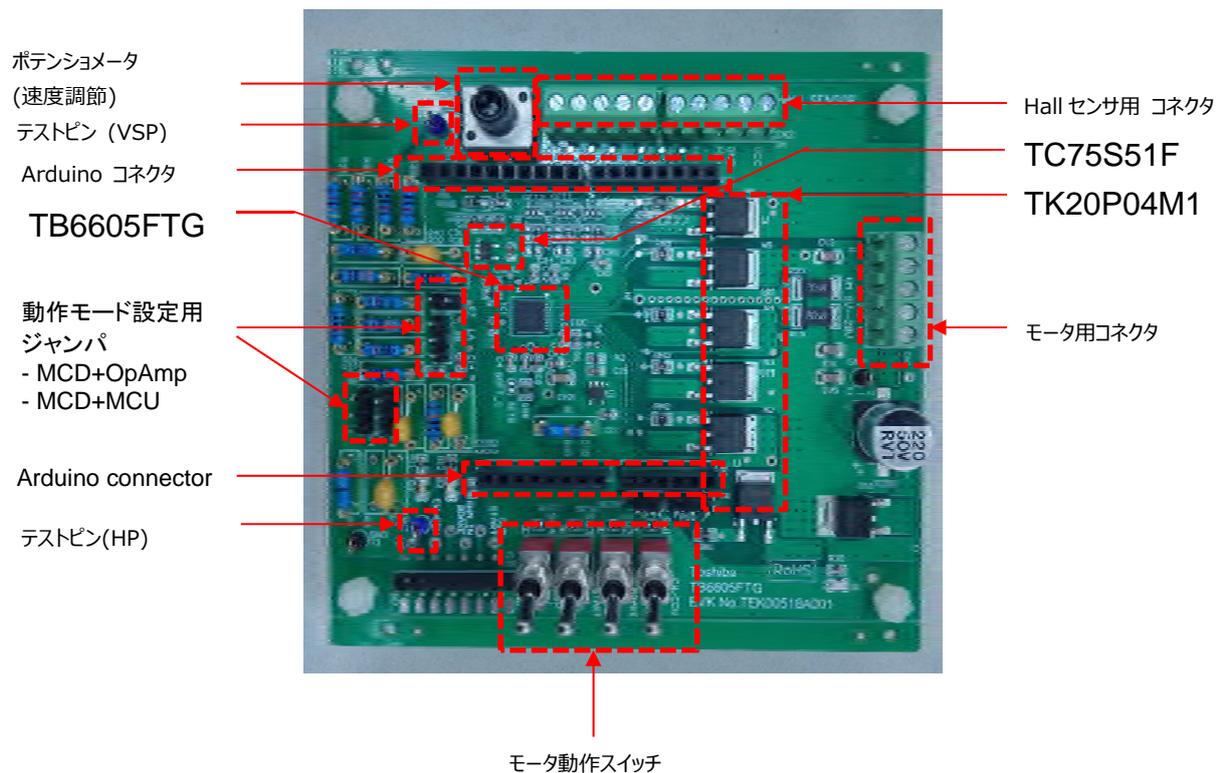
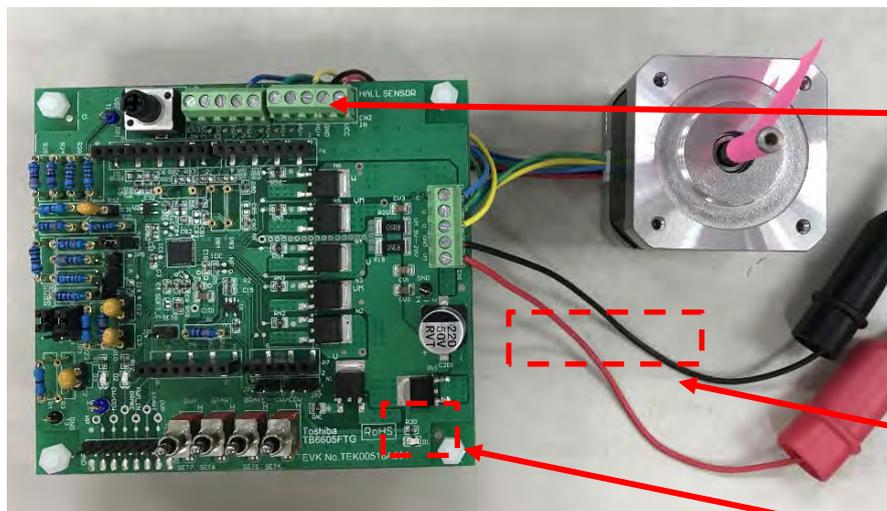


図 4-1 外観

5. モータ接続



モータケーブル

+5V	赤
HALL U	黄色
HALL V	緑
HALL W	青
GND	黒
Phase U	黄色
Phase V	緑
Phase W	青

電源電圧
10-28V
電流 2A 以下

添付のモータが動作しないときは、電流が 0.02A 以下になっていないか確認してください。

図 5-1 モータ接続

注)

モータの U/V/W 相のケーブルは、リファレンスボードの U/V/W 相に接続してください。
ホールセンサの U/V/W 相ケーブルとリファレンスボードの U/V/W 相のケーブルを接続しないようにしてください。

6. スイッチ

本 Closed-loop システムは、下記機能を有します。

- Start: モータ回転のオン/オフ
- Brake: モータ緊急停止用
- CW/CCW: モータ回転方向設定。右回り(CW) / 左周り(CCW)
- OVP: 正弦波 / 矩形波選択
- ポテンシオメータ: モータ速度調整 (6.1 参照)

下記に設定を記します。

START	BRAKE	Mode
L	H	Active/Normal
L	L	Active/Brake
H	H/L	Standby

	H	L
CW/CCW	CCW	CW
OVP	正弦波	矩形波

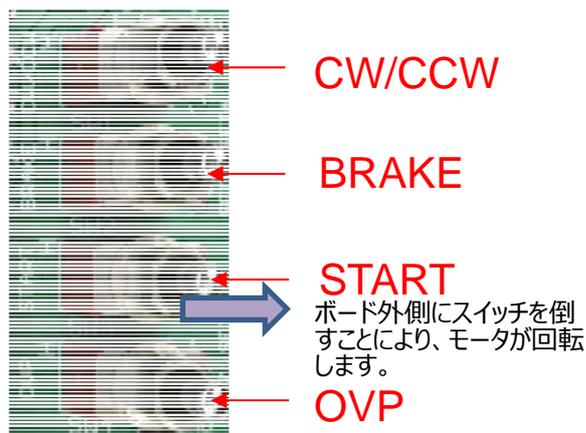
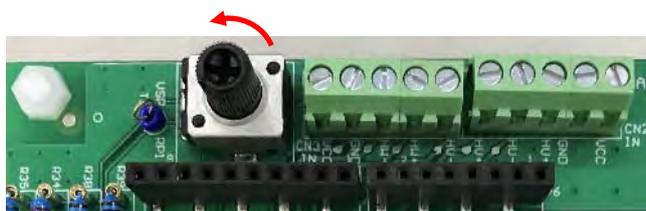


図 6-1 スイッチ

6.1. ポテンシオメータ

モータの回転速度は、ボード上のポテンシオメータで調整可能です。



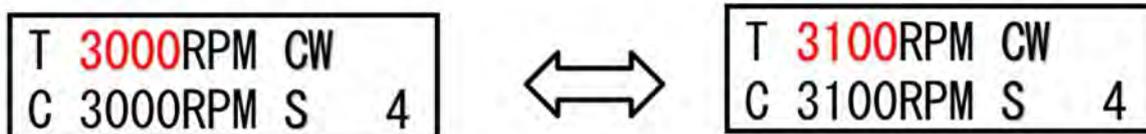
MCD + Op-Amp モード

ターゲットスピードはメータを左回りに回して上げます。

MCD+MCU モード

ターゲットスピードはメータを左回りに回すと 100 RPM 単位で上がります。

ターゲットスピードは、下記のように LCD に表示されます。



7. MCD+MCU モード

7.1. MCD+MCU モード時ブロック図

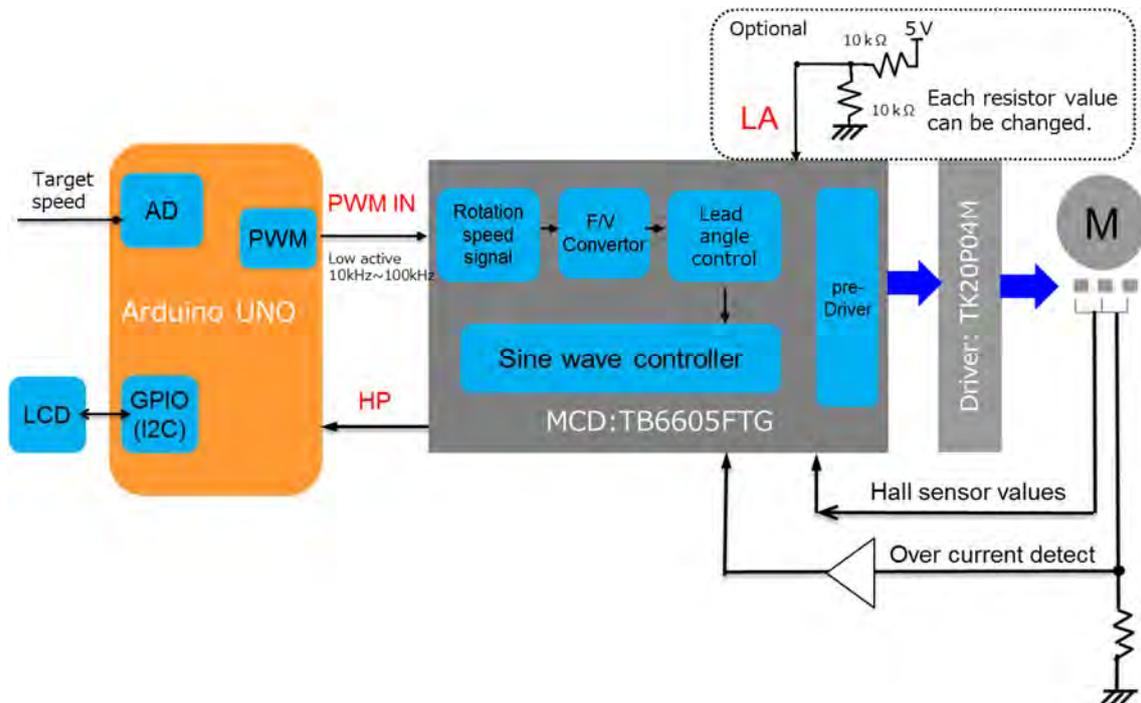


図 7-1 MCD+MCU モード時ブロック図

7.2. ハードウェア接続

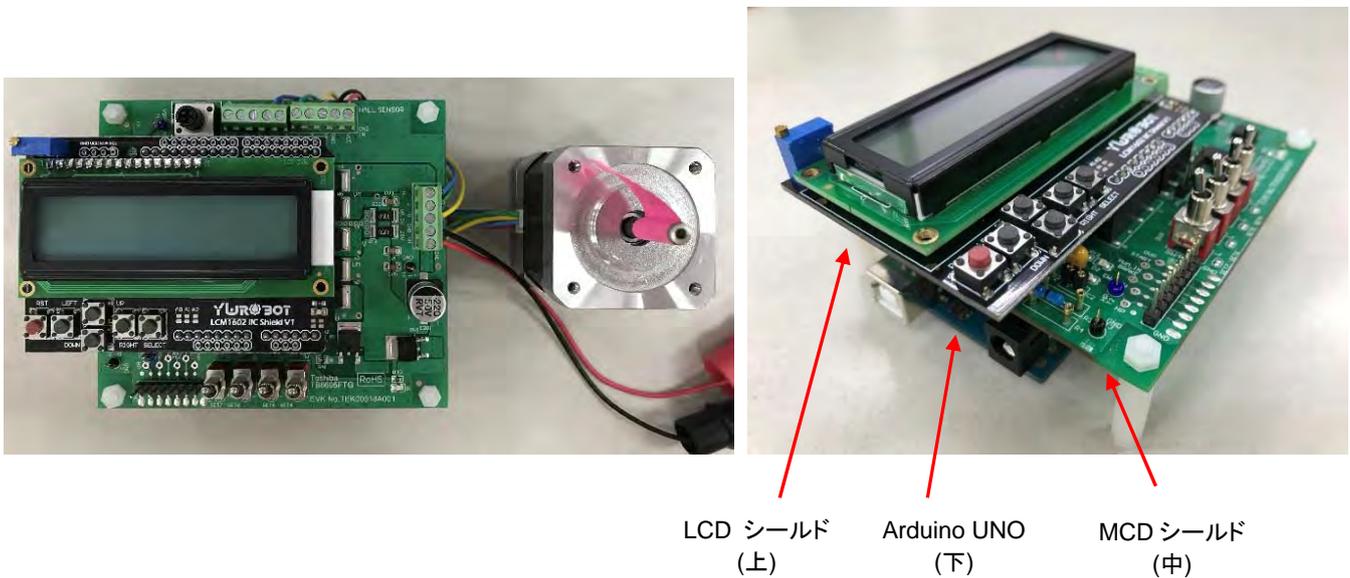


図 7-2 ハードウェア接続

7.3. ジャンパ設定

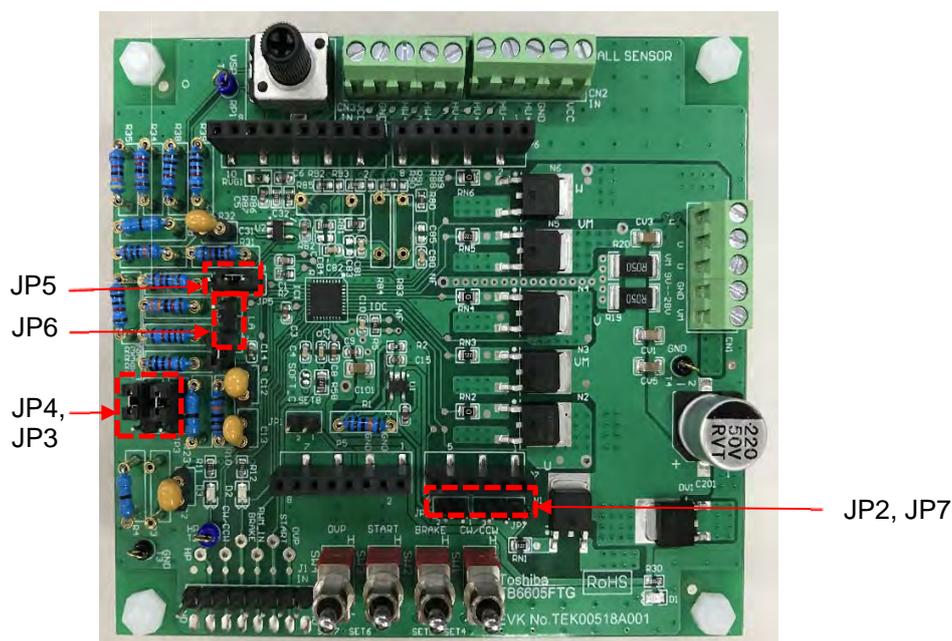
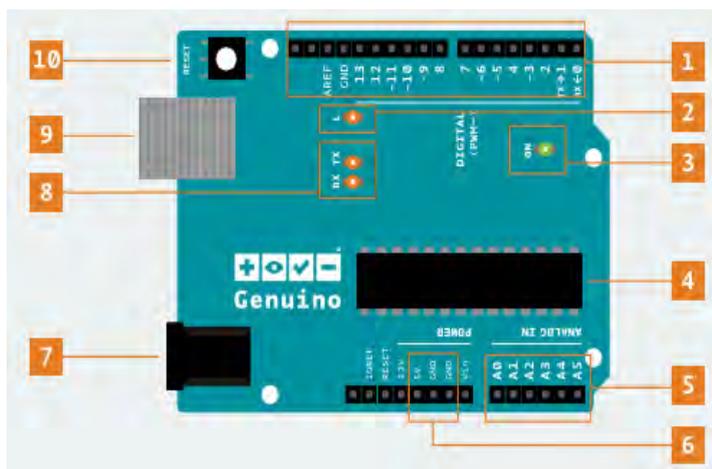


図 7-3 ジャンパ設定

表 7-1 ジャンパ設定

ジャンパ	設定	機能
JP1	Open	VREG (Reserved)
JP2	Open	IDC: MCU との接続なし。(機能追加用)
JP3	Pin 2 と Pin 3 を接続	PWM-IN: MCU からの入力。速度調整用。
JP4	Pin 2 と Pin 3 を接続	HP: MCU への Hall センサパルス出力。回転数検知用。
JP5	Pin 2 と Pin 3 を接続	DIF-IN: GND
JP6	Open	LA: 未使用。(機能追加用)
JP7	Open	IDC: MCU との接続なし。(機能追加用)

7.4. Arduino 接続



Analog pins/Digital pins		
Pin	Function	I/O
A0	Key(LCD)	I
A1	VSP*	I
A2	IDC	I
A3		
A4	SDA(LCD)	I/O
A5	SCL(LCD)	O

Digital pins		
Pin	Function	I/O
D0	Rx(Serial)	I
D1	Tx(Serial)	O
D2	HP	I
D3	PWM IN	O
D4	CW/CCW	O
D5		
D6		O
D7	START	O
D8	BRAKE	O
D9		
D10		
D11	LA(PWM)	O
D12		
D13		

A0, A4, A5: LCD シールド接続用
 D0, D1: シリアル信号 Tx/Rx 用。デバッグログおよびターゲットスピード出力用

xxx: モーター制御用
 IDC および LA は、現ソリューションでは未使用。

図 7-4 Arduino 接続

7.5. LCD シールド

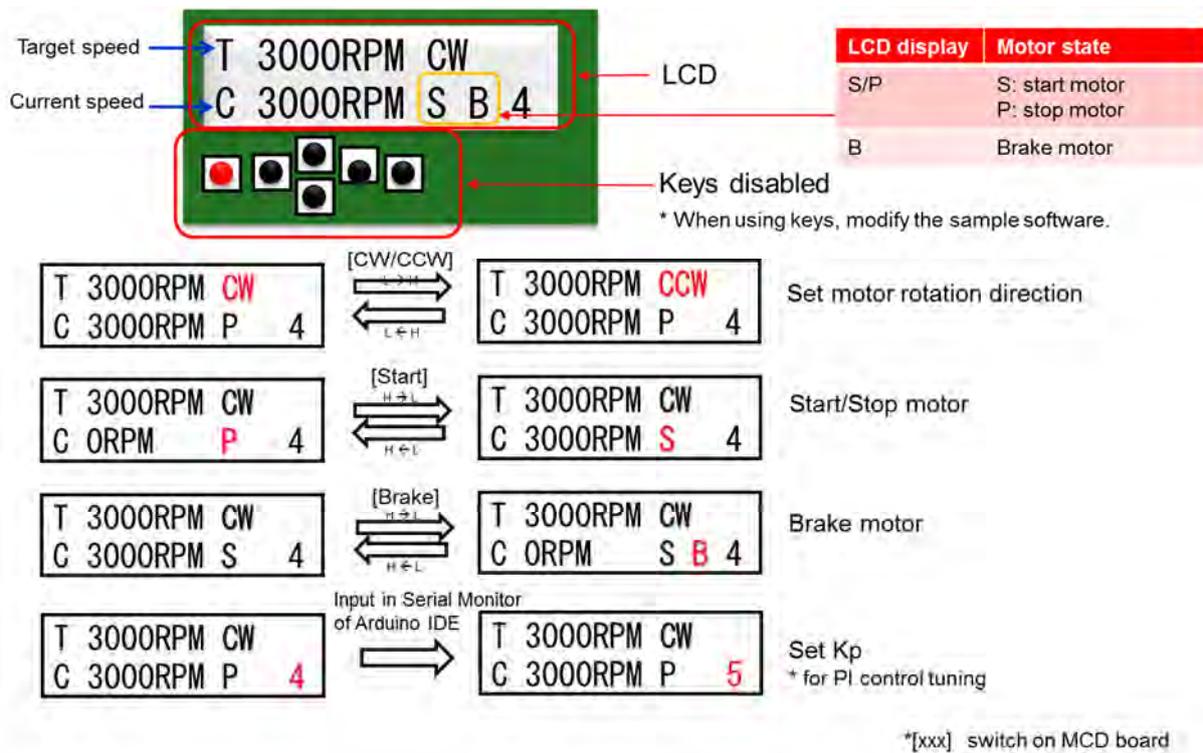


図 7-5 LCD シールド

7.6. 使用方法

下記手順を(1)から(11)まで順番に行ってください。

本システムには、ACT Motor 社製 42BLF BLDC モータが添付されています。

別のモータを使用する場合、ボードを正常動作させるためには Hall センサタイプの BLDC タイプモータを使用してください。

表 7-2 MCD+MCU モードの使用方法

ステップ	説明
(1)	リファレンスボードの外観を確認し、6 章に記述したようにスイッチを設定。
(2)	モータをリファレンスボードに接続。(5 章参照)
(3)	リファレンスボードを Arduino に差し込み、LCD ボードをリファレンスボードに差し込む。
(4)	安定化電源をリファレンスボードに接続。
(5)	モータ制御スイッチを確認し、Start スイッチを“H”に設定する。Brake スイッチは“H”に設定、OVP スイッチは、希望する波形に設定する。(“L”: 矩形波, “H”: 正弦波)。CW/CCW スイッチは希望する回転方向に設定する。(“L”: CW, “H”: CCW)
(6)	リファレンスボードに電源を供給
(7)	Arduino と PC を USB 接続し、添付の sketch を Arduino にアップロードします。(7.4 参照)
(8)	ポテンシオメータを使用して、LCD 上に表示されるターゲットスピードを設定。
(9)	Start スイッチを“L”に入れるとモータが回転。その際 LCD に表示されるモータ回転スピードがターゲットスピードに届くかどうかを確認。
(10)	ポテンシオメータを使用して、LCD 上に表示されるターゲットスピードを変更。その際 LCD に表示されるモータ回転スピードがターゲットスピードに届くかどうかを確認。
(11)	Start スイッチを“H”に入れ、モータ回転を停止。

モータを回転させるには、リファレンスボード上の Start スイッチを“L”に入れます。

現在の回転スピードは LCD 上に“C” の項目で表示されます。

回転開始後は、供給電圧が変動しても回転スピードを一定に維持するよう制御します。

7.7. Arduino UNO 上で使用するソフトウェア

Arduino Software (IDE) は下記サイトからダウンロードしてください。

<https://www.arduino.cc/en/Main/Software>

該当するシステムに沿った手順を確認してください。

ダウンロードを行い、インストールしてください。

Arduino IDE を実行し、添付のサンプル sketch をオープンしてください。

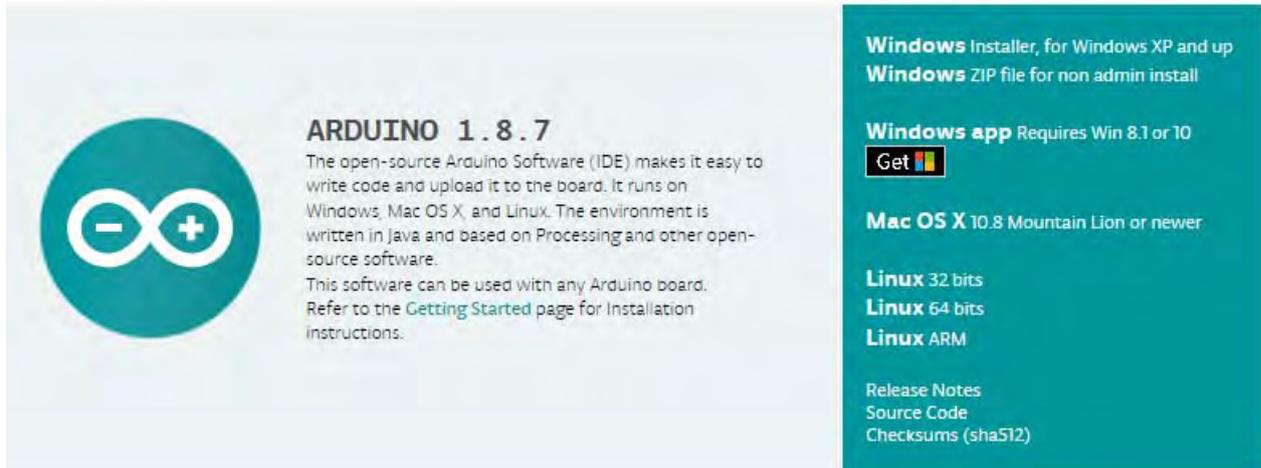


図 7-6 Arduino IDE

7.8. Arduino Sketch

Sketch とは、Arduino を使う際のプログラムを Sketch と呼びます。本 Closed-loop システムは、添付の “closed_loop_mcd.ino” という Sketch を Closed-loop システムを実現する際に必要とします。

closed_loop_mcd.ino には 3rd パーティ製の下記のライブラリを使用します。

– #include <LiquidCrystal_I2C.h>

– #include <TimerOne.h>

* 3rd パーティ製ライブラリのインストールについては、次ページを参照してください。

LiquidCrystal_I2C: 文字を LCD シールドに表示するためのソフト

TimerOne: 周期割り込み用のハードウェアタイマ

7.8.1. インストール方法

1. 添付の LiquidCrystal_I2C.zip と TimerOne.zip を解凍。
2. LiquidCrystal_I2C と TimerOne を下記にコピー。
“...\Documents\Arduino\libraries”
3. 下記のようにライブラリがインストールされているか確認。
Arduino IDE->Sketch->Include Library-> LiquidCrystallI2C
Arduino IDE->Sketch->Include Library-> TimerOne

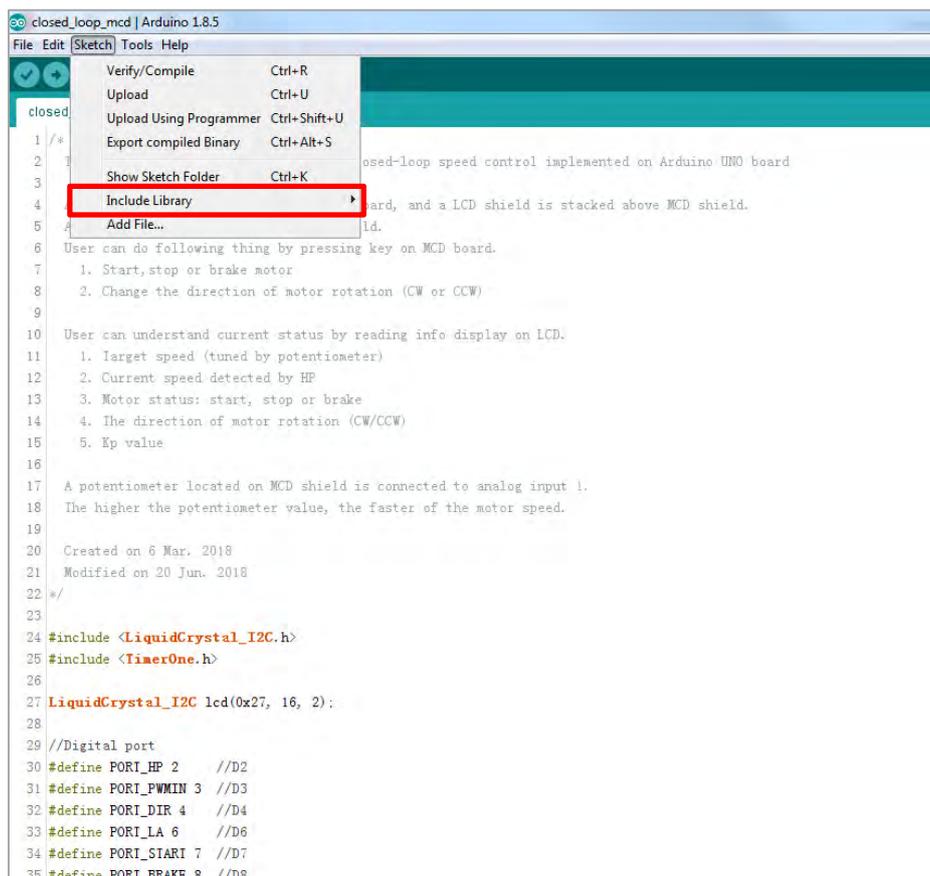


図 7-7 インストール

7.9. Arduino UNO との接続

1. closed_loop_mcd.ino を Arduino IDE 上でオープン
2. PC と Arduino ボードを USB ケーブルで接続
3. メニューからボード Board と Port を選択

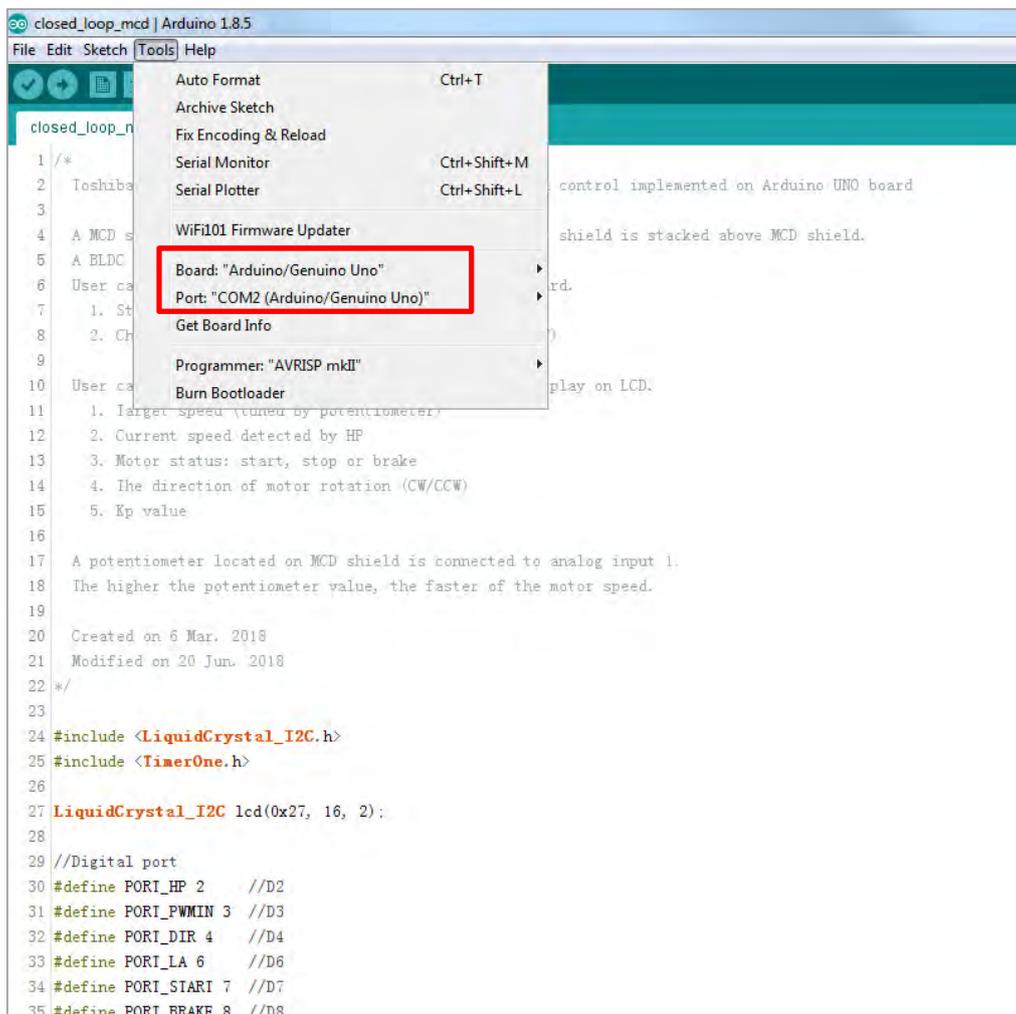


図 7-8 Arduino UNO との接続

7.10. コードのアップロード

1. エラーがなくなるまでコードを修正し、Verify/compile する
2. USB ケーブルを介してプログラムを Arduino にアップロードする

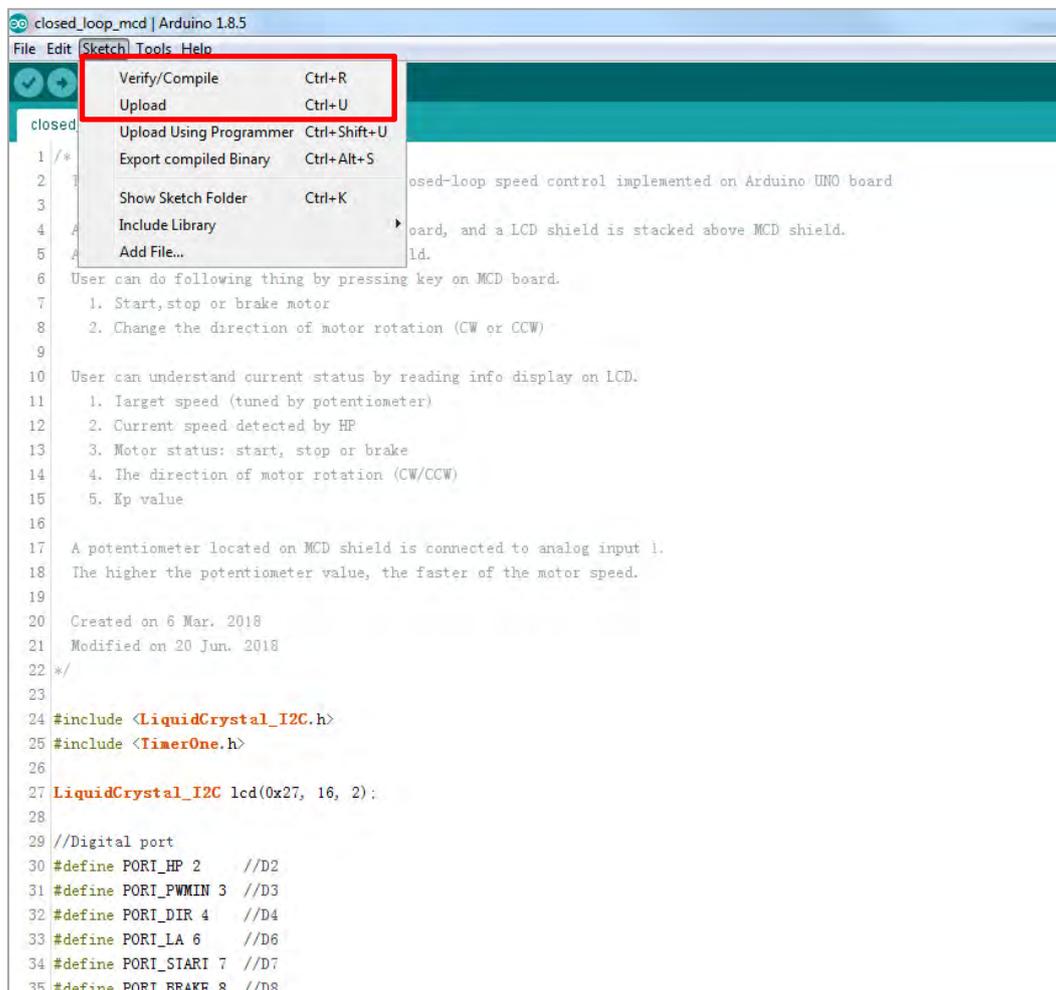


図 7-9 コードのアップロード

7.11. Arduino IDE のシリアルモニタ

デバッグ用のシリアルモニタ出力

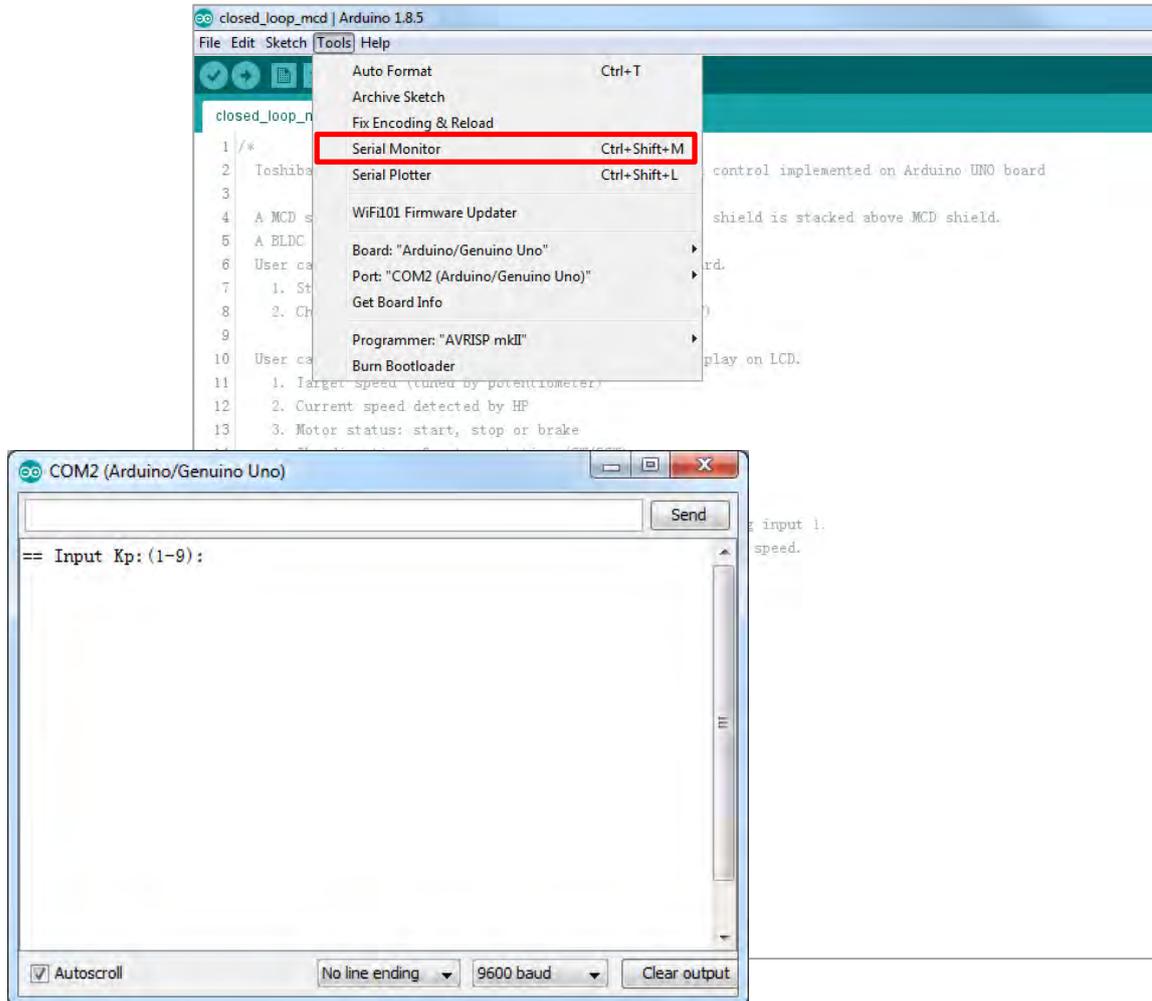


図 7-10 シリアルモニタ

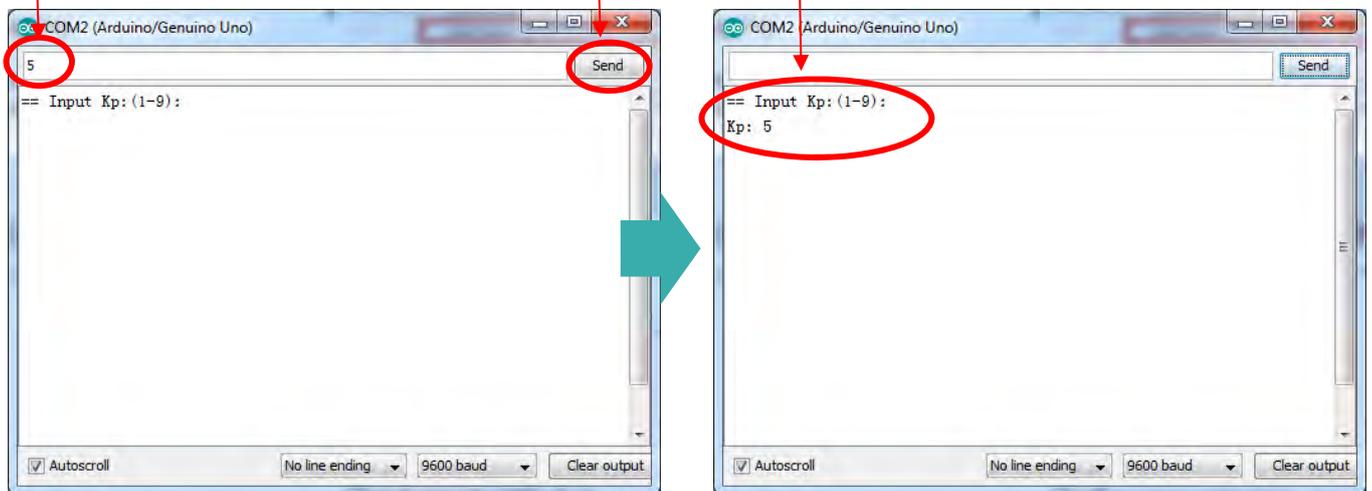
7.12. シリアルモニタ上での Kp 値の変更方法

PI 制御用の Kp 値はシリアルモニタを使用して変更が可能です。

1. Kp 値を入力

2. 送信ボタンをクリック

3. Kp 値が変わる



Kp 値を変更した場合、変更後の Sketch を Arduino アップロードしてください。

図 7-11 Kp 値の変更

7.13. LCD シールド

- ハードウェア
Arduino コネクタのうち 3 本の信号線を使用
- ソフトウェア
LCD シールド用に“LiquidCrystal2C”ライブラリを使用

7.13.1. その他シールド

- ハードウェア
上述より使用する信号数の多い LCD シールドを使用する場合があります。
ピンアサインについては、7.9 章を参照してください。モータおよびシリアル用のピンを使用しないようにしてください。
- ソフトウェア
使用している LCD シールドのベンダから供給されたライブラリを使用し、ソフトウェアを適切に変更してください。

7.14. MCD + MCU の場合のモータ速度制御

PID 制御は Closed-loop 制御を実装する際に使用されます。Arduino UNO により MCD を制御しターゲットスピードへ到達するようモータ速度制御にフィードバックをかけます。

7.14.1. PID アルゴリズム

PI 制御が実装されています。標準の PID 演算では、微分“D”は使用しません。

$$CV(t) = K_p \cdot E(t) + K_i \cdot \int E(t) dt + K_d \cdot (dE(t)/dt)$$

簡略化した PI 演算

$$CV(t) = CV(t-1) + K_p \cdot (E(t) - E(t-1)) + K_i \cdot E(t)$$

$$E(t) = \text{Set point} - \text{Input}$$

- 入力 – 処理変数
 - Hall センサ(HP 信号)を検出によるモータ回転速度の検出
- 出力 – 制御変数
 - PWM デューティ (0-255, ローアクティブ) (PWM IN 信号)
- 設定ポイント
 - ポテンシオメータによるターゲット速度の設定

7.14.2. PID 制御

P (Proportional term): $CV(t) = K_p \cdot E(t)$

比例ゲイン K_p が固定の場合、制御出力(CV)はエラー $E(t)$ に比例します。エラーが大きくなる場合、CV 値も自動的に大きくなります。 K_p 値が大き過ぎる場合、システムは不安定になりますが、逆に K_p 値が小さ過ぎる場合は、システムの応答が悪くなります。

なぜなら比例項の制御出力(CV)は、SP と PV 間のエラーに依存するからです。スタンバイ状態エラー(ドループ)は、比例制御のみ使用する場合において存在します。このスタンバイ状態エラーを抑えるためには、補償バイアス項(いわゆるオフセット)を事前に設定することが必要です。最も良い対策としては、積分値を自身のシステムに追加することです。

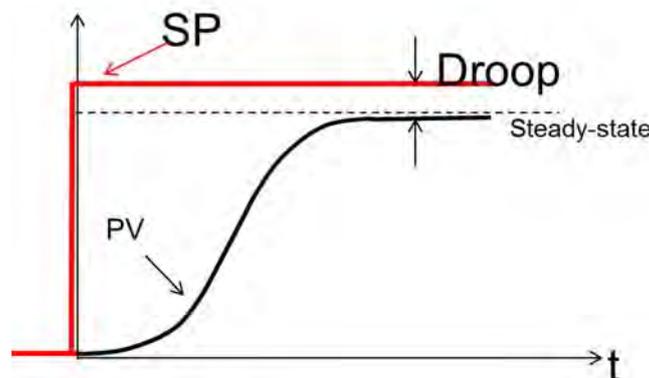


図 7-12 PID 制御

K_p 値が増加する場合

- 立ち上がり時間: 減少
- オーバershoot: 増加
- 設定時間: 微小変化
- ドループ: 減少
- 安定性: 劣化

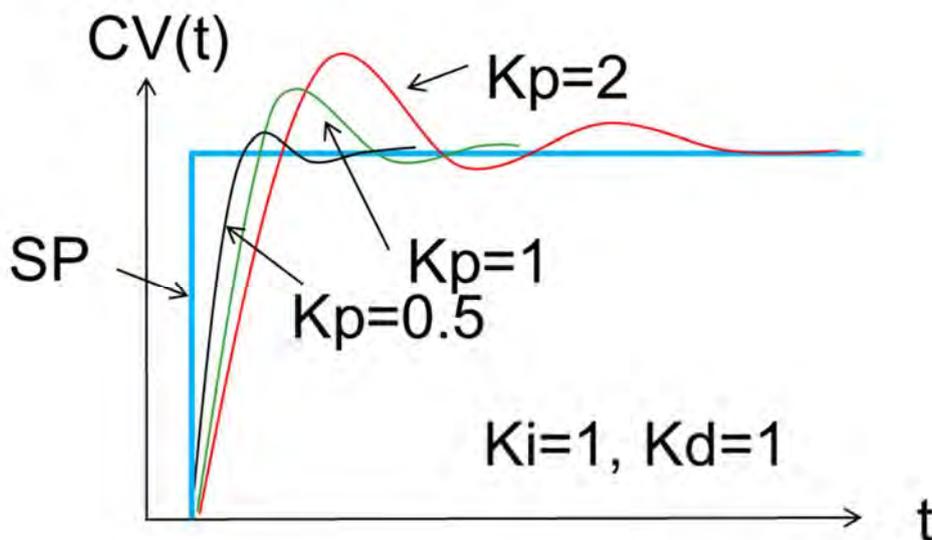


図 7-13 K_p 値の増加

K_i 値が増加する場合

- 立ち上がり時間: 減少
- オーバershoot: 増加
- 設定時間: 増加
- ドループ: 排除
- Stability: 劣化

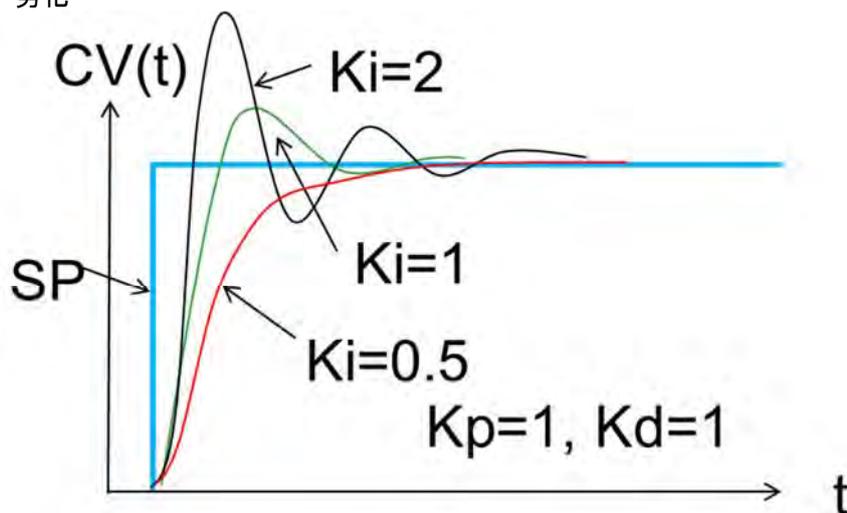


図 7-14 K_i 値の増加

7.14.3. PI 調整手順

1. まず Ki 値と Kd 値 を 0 に設定してください。Kp 値をループ出力が振動するまで増加させます。そして Kp を 1/4 振幅減衰応答型用に、その値の半分の値に設定します。
2. 次にオフセットが補正処理されるまでの十分な時間まで Ki 値を増加させますが、Ki 値が過剰になるとシステムは不安定になります。
3. この調整は、応答速度とシステムの安定性のバランスが取れるまで繰り返し行います。

7.14.4. ソースコード

```
err = rotateSpeedHp - detectedHp;    //error = set value - actual value
if (detectedHp == 0) {
    bias = Kp_s * (err - err1); // do once only at startup when speed is zero
}
else
{
    bias = Kp * (err - err1) + Ki * err;
    if (err < 2 && err > -2)
    {
        pidOver_U = pidOver2_U;
        pidOver_D = pidOver2_D;
    }
    else
    {
        pidOver_U = pidOver1_U;
        pidOver_D = pidOver1_D;
    }
    if (bias > pidOver_U)
        bias = pidOver_U;
    else if (bias < pidOver_D)
        bias = pidOver_D;
}
```

8. MCD + Op-Amp モード

MCD + Op-Amp モードでは、スピード制御は PI 制御を利用したスピード制御部分で行われます。

- ・ VSP はターゲットスピードを決定するために使用。
- ・ LA 電圧は、現状スピードを表すフィードバック信号として使用。
- ・ Dif-in は、モータスピードの出力デューティを調整するために使用。
- ・ PWM-in は、GND に接続。

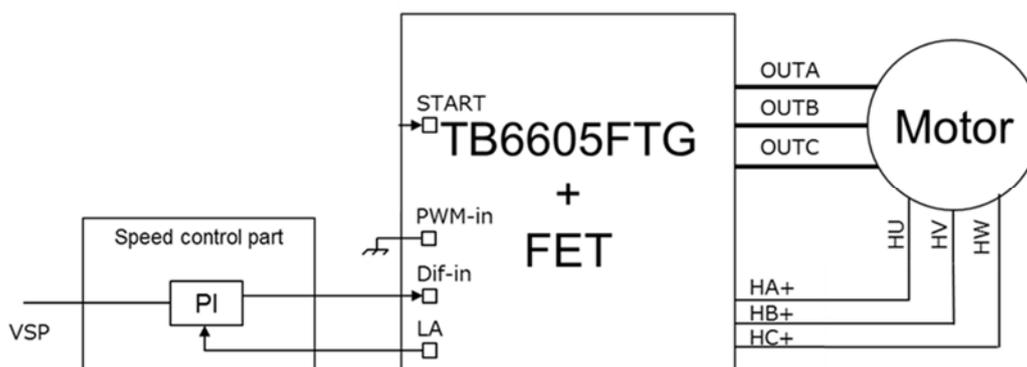


図 8-1 MCD + Op-Amp モード

8.1. MCD+Op-AMP モード時のブロック図

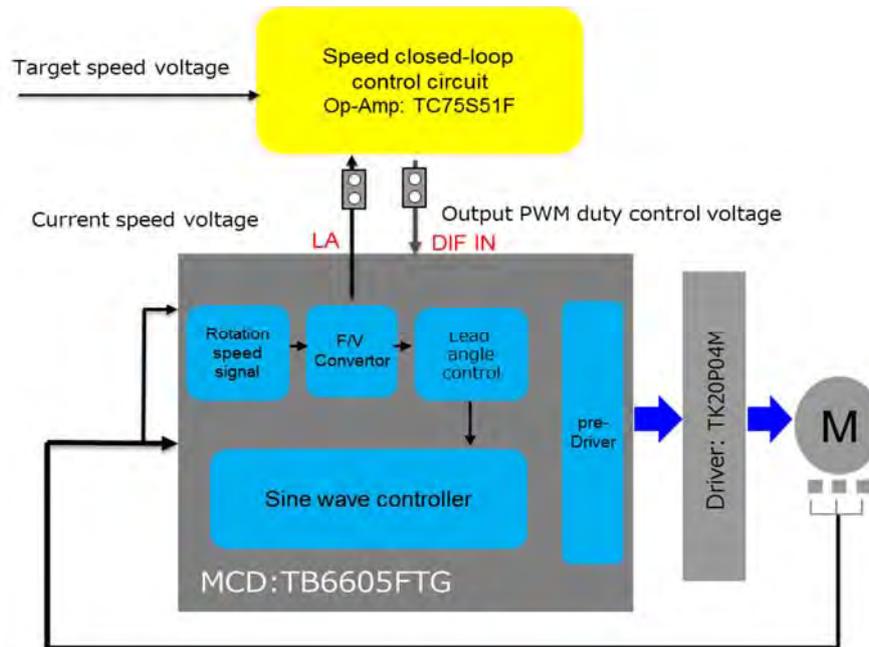


図 8-2 MCD + Op-Amp モード時のブロック図

8.2. 使用方法

下記ステップ(1)から(8)を順に実行します。

本 Closed-loop システムは、ACT Motor 社製 42BLF BLDC モータを含みます。

別のモータを使用する場合は、ホールセンサ BLDC タイプのモータを使用してください。

表 8-1 MCD + Op-Amp モード時の使用方法

ステップ	説明
(1)	外観を確認の上、スイッチを正しく設定(6章参照)。
(2)	リファレンスボードをモータに接続 (5 参照)
(3)	安定化電源をリファレンスボードに接続。
(4)	モータ制御スイッチを確認し、Start スイッチを“H”に、Brake スイッチは “H”に設定、OVP スイッチは希望する波形に設定する。(“L”: 矩形波, “H”: 正弦波)。CW/CCW スイッチは希望する回転方向に設定する。(“L”: CW, “H”: CCW)
(5)	リファレンスボードに電源を供給し、LED が ON になったことを確認。
(6)	Start スイッチを“L”に入れるとモータが回転。
(7)	ポテンシオメータを使用して、モータ速度を変更。 “HP”をモニタし、どのようにモータ回転が変化するかを確認。
(8)	Start スイッチを“H”に入れ、モータ回転を停止。

エラーが発生したら

- パイロットの LED が光らないとき: 電源供給ケーブルと電圧が適正かどうか確認してください。
- モータが回転しないとき: モータとの接続を確認の上、モータ制御スイッチを確認してください。

8.3. 操作手順

8.3.1. MCD + Op-Amp のハードウェア設定

8.3.2. ジャンパ設定

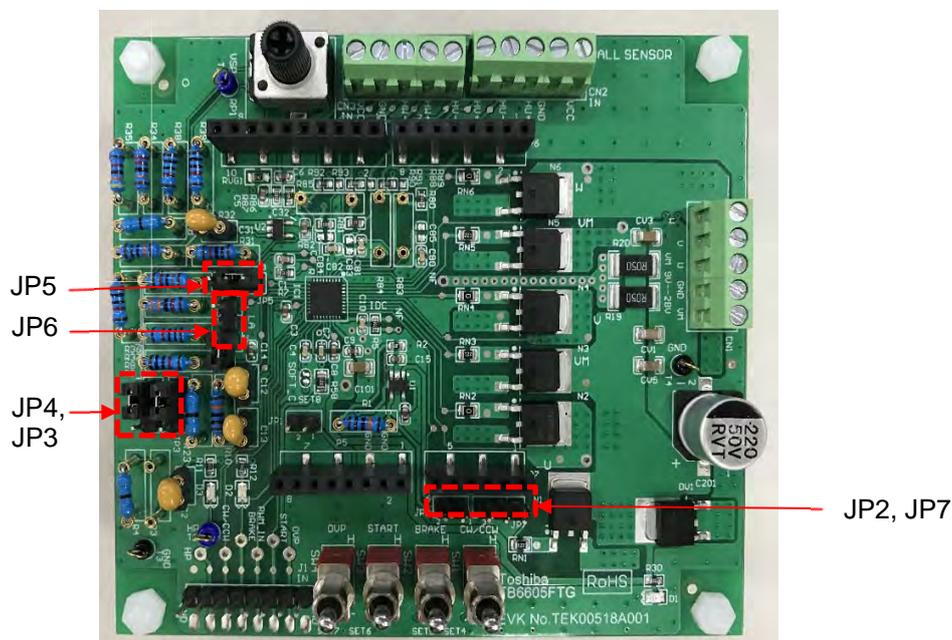


図 8-3 ジャンパ設定

表 8-2 ジャンパ設定

ジャンパ	設定	機能
JP1	Open	VREG (Reserved)
JP2	Open	IDC: MCU との接続なし。
JP3	Pin 1 と Pin 2 を接続	PWM-IN: GND
JP4	Open	HP: MCU との接続なし。
JP5	Pin 1 と Pin 2 を接続	DIF-IN: Op-Amp 出力。速度制御用。
JP6	Pin 1 と Pin 2 を接続	LA: Op-Amp 出力
JP7	Open	IDC: MCU との接続なし。

8.4. 速度制御部

Rail-to-rail タイプ Op-Amp を使用してください。

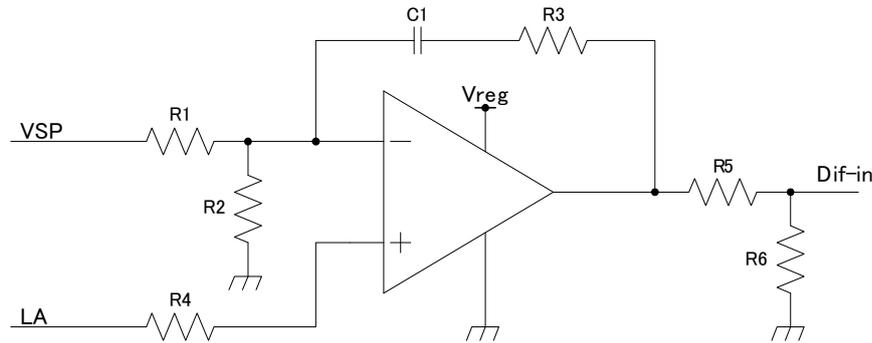


図 8-4 速度制御部

8.4.1. パラメータ調整

R1 および R2:

$VSP=VMAX$ が最大速度を表す場合、 $LA=VLAMAX$ は最大速度となります。

R1 および R2 は、 $LAMAX=VMAX \times R2 / (R1 + R2)$ の関係となりますので、値を調整してください。

R3: 比例制御 Kp の定数は、R3 と関連します。 Kp を増加させるには、R3 を増加してください。

C1: 積分制御 Ki の定数は、C1 と関連します。 Ki を増加させるには、C1 を増加させてください。

R5 および R6 :

R5 および R6 を調整し、Dif-in が 3V を超えないよう確実に調整してください。

Dif-in=VMIN でモータが最小速度で動作する場合、R5 および R6 が $MIN < 5 \times R6 / (R5 + R6) < 3$ になるよう設定することを推奨します。

R5 および R6 の合計抵抗値は、約 50k ~ 100k になるように設定してください。

8.5. 自動進角設定

高効率を実現するため、TB6605FTG に進角パラメータを必要とします。添付とは異なるモータを使用する場合、進角パラメータを必要に応じて変更してください。
また、TCR パラメータも適切な進角のために変更が必要です。

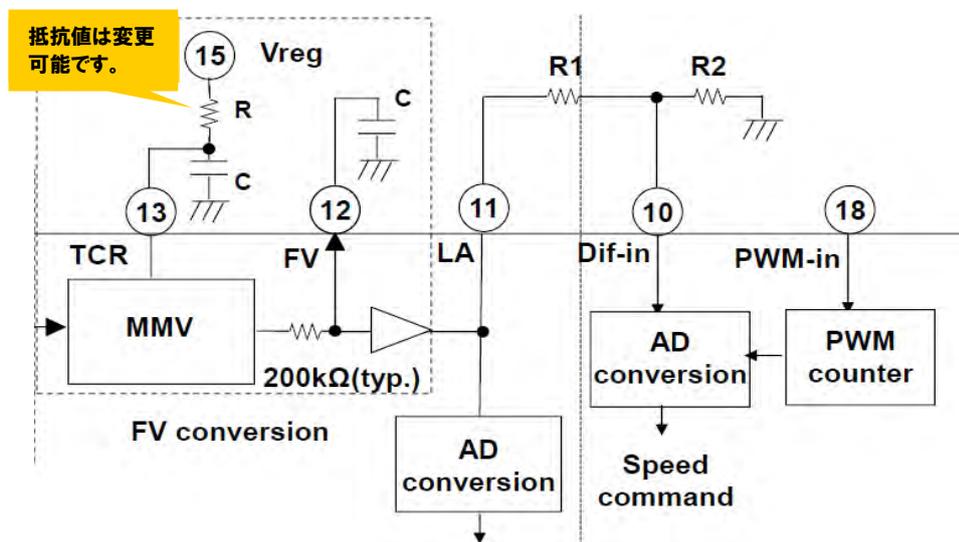
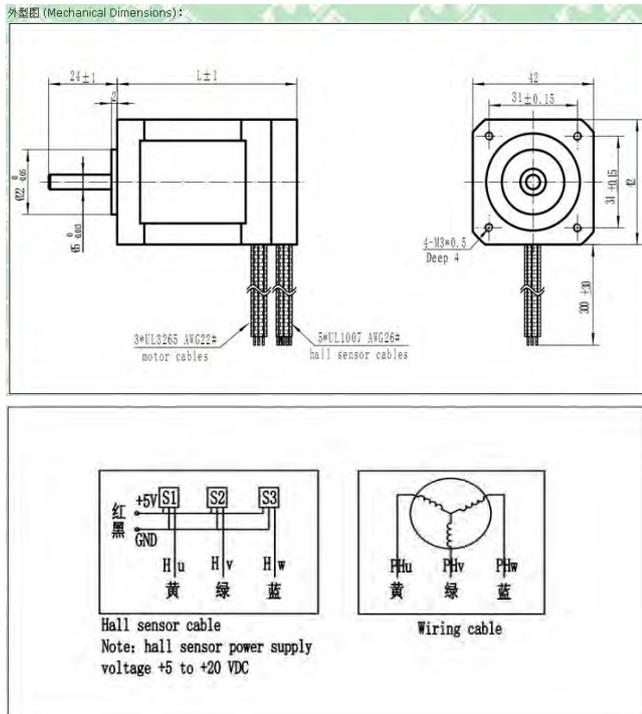


図 8-5 自動進角設定

注意

- 電源供給は 10V ~ 28V、電流は 2A 以下であることを確かめてください。(添付モータ仕様による制限)
- 高電流での動作時、モータおよびリファレンスボードは高温になる可能性があります。
- 本システムの未使用ピンは、製作されるシステムに合うよう適切に処理してください。
- 添付以外のモータを使用する場合、抵抗およびピンソケット用のコンデンサは、必要に応じて変更してください。

9. 添付 BLDC モータ仕様



42BLF

产品型号: 42BLF (42BLF English)

绕组连接方式 Winding Type-----星形 Star

霍尔反映角度 Hall Effect Angle-----120° 电角度 Electrical angle

绝缘等级 Insulation Class-----B

环境温度 Ambient Temperature Range----- -20°C~+50°C

绝缘电阻 Insulation Resistant----- MIN100MΩ . 500V

介电强度 Dielectric Strength-----500V AC 1Minute

电机参数 (Electrical Specifications):

电机型号 Model	42BLF01
极数 Number of Poles	8
相数 Number of Phases	3
额定电压 Rated Voltage VDC	24
额定转速 Rated Speed RPM	4000
额定转矩 Rated Torque N·m	0.063
额定电流 Rated Current Amps	1.9
输出功率 Output Power Watts	26
峰值扭矩 Peak Torque N·m	0.18
峰值电流 Peak Current Amps	5.7
转矩常数 Torqueconstant N·m/Amps	0.035
反电势常数 Back EMF V/Krpm	3.7
转动惯量 Rotor Inertia g·cm ²	24
机身长度 Body Length mm	47
重量 Mass kg	0.29

ご利用規約

本規約は、お客様と東芝デバイス&ストレージ株式会社(以下「当社」といいます)との間で、当社半導体製品を搭載した機器を設計する際に参考となるドキュメント及びデータ(以下「本リファレンスデザイン」といいます)の使用に関する条件を定めるものです。お客様は本規約を遵守しなければなりません。本リファレンスデザインをダウンロードすることをもって、お客様は本規約に同意したものとみなされます。なお、本規約は変更される場合があります。当社は、理由の如何を問わずいつでも本規約を解除することができます。本規約が解除された場合は、お客様は、本リファレンスデザインを破棄しなければなりません。またお客様が本規約に違反した場合は、お客様は、本リファレンスデザインを破棄し、その破棄したことを証する書面を当社に提出しなければなりません。

1. 禁止事項

1. 本リファレンスデザインは、機器設計の参考データとして使用されることを意図しています。信頼性検証など、それ以外の目的には使用しないでください。
2. 本リファレンスデザインを販売、譲渡、貸与等しないでください。
3. 本リファレンスデザインは、高温・多湿・強電磁界などの対環境評価には使用できません。
4. 本リファレンスデザインを、国内外の法令、規則及び命令により、製造、使用、販売を禁止されている製品に使用しないでください。

2. 保証制限等

1. 本リファレンスデザインは、技術の進歩などにより予告なしに変更されることがあります。
2. 本リファレンスデザインは参考用のデータです。当社は、データおよび情報の正確性、完全性に関して一切の保証をいたしません。
3. 半導体素子は誤作動したり故障したりすることがあります。本リファレンスデザインを参考に機器設計を行う場合は、誤作動や故障により生命・身体・財産が侵害されることのないように、お客様の責任において、お客様のハードウェア・ソフトウェア・システムに必要な安全設計を行うことをお願いします。また、使用されている半導体素子に関する最新の情報(半導体信頼性ハンドブック、仕様書、データシート、アプリケーションノートなど)をご確認の上、これに従ってください。
4. 本リファレンスデザインを参考に機器設計を行う場合は、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。当社は、適用可否に対する責任を負いません。
5. 本リファレンスデザインは、その使用に際して当社及び第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。
6. 当社は、本リファレンスデザインに関して、明示的にも黙示的にも一切の保証(機能動作の保証、商品性の保証、特定目的への合致の保証、情報の正確性の保証、第三者の権利の非侵害保証を含むがこれに限らない。)をせず、また当社は、本リファレンスデザインに関する一切の損害(間接損害、結果的損害、特別損害、付随的損害、逸失利益、機会損失、休業損、データ喪失等を含むがこれに限らない。)につき一切の責任を負いません。

3. 輸出管理

お客様は本リファレンスデザインを、大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用してはなりません。また、お客様は「外国為替及び外国貿易法」、「米国輸出管理規則」等、適用ある輸出関連法令を遵守しなければなりません。

4. 準拠法

本規約の準拠法は日本法とします。