Motor Control Closed-loop System

REFERENCE GUIDE

RD022-RGUIDE-01-E

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Table of Contents

1. INTRODUCTION

This reference guide describes the usage of Motor Control Closed-loop system incorporating Toshiba's BLDC motor driver IC TB6605FTG. The board is designed as a plug-in board (Shield) for Arduino UNO platform. This document provides guidelines to quickly setup the hardware and software for BLDC motor driver IC performance evaluation.

This system can be used as two types:

1. MCD + MCU mode
2. MCD + Op-Amp mode

Motor Control Closed-loop System evaluation kit includes:

– Motor Control Reference Board (with Arduino connector)
  * Arduino is not attached.
– Brushless Motor (attached)

Figure 1-1 Motor Control Closed-loop system

In MCD+MCU mode, the following items are needed for evaluation.

-MCU Arduino (Not attached, Commercial goods)
-LCD module (Commercial goods)

2. FEATURES

- Support three-Hall sensor BLDC motor driver
- Motor control function: Start, Brake, CW/CCW function
- Tune motor rotation speed with potentiometer
- A reference motor is attached for quick start
- Replaceable resistors and capacitors are used to adopt other motors
- Two work modes are available:
  1. MCD+MCU mode
     Designed as plug-in Shield for Arduino UNO platform
     Speed closed-loop control is implemented based on PID control in software.
     Arduino example program (Sketch) is provided.
  2. MCD + Op-Amp mode
     Speed closed-loop control is implemented using Op-Amp on the board.

3. SPECIFICATIONS

Table 3-1 Specifications

Parameter	Description
TB6605FTG	Sine Wave BLDC motor controller
Power supply voltage	10V－28V DC
Attached motor specifications	Middle speed motor 10V－24V/ 0.5A－10A/ Output 26W/ 4000rpm
Motor driver output + MosFET	N ch+ Nch MosFET (Toshiba TK20P04M1)
Main ICs	TB6605FTG, TC75S51F, TK20P04M1 (Arduino UNO is needed in MCD+MCU mode.)
MosFET specifications and package information	TK20P04M1: NMOS, 40V/20A, DPAK, Rdson=19mΩ

4. External View

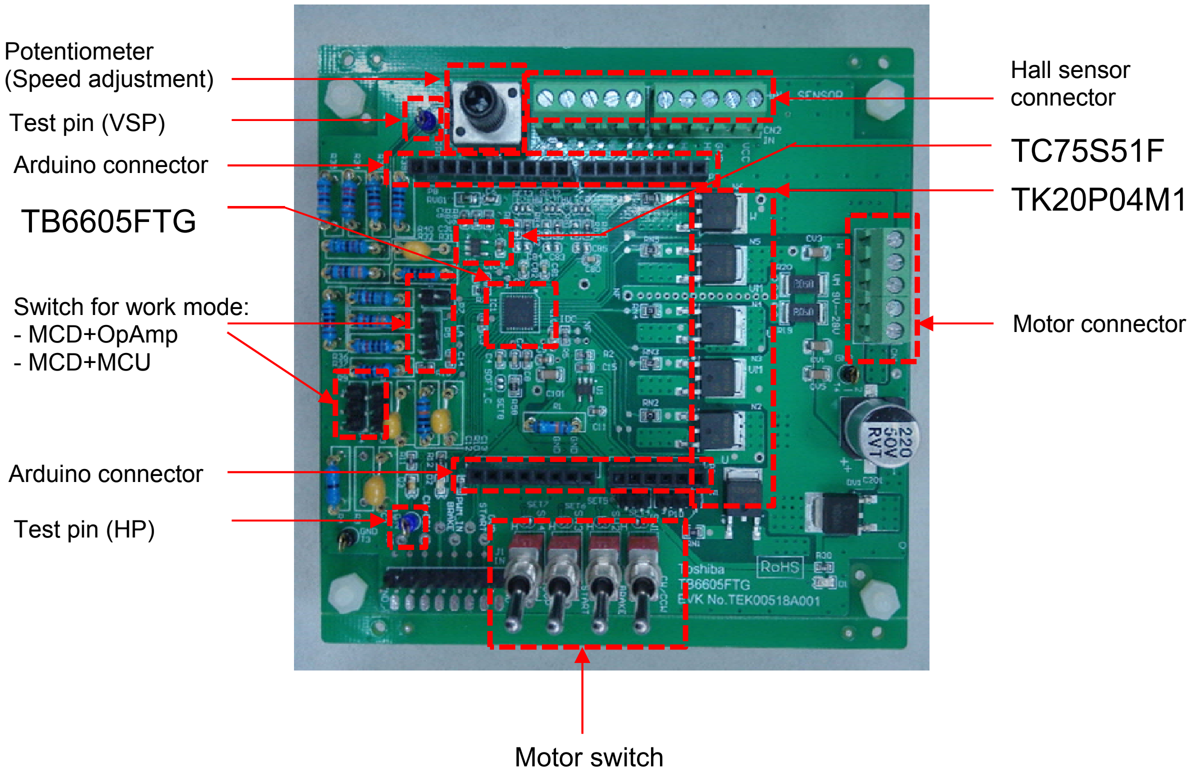
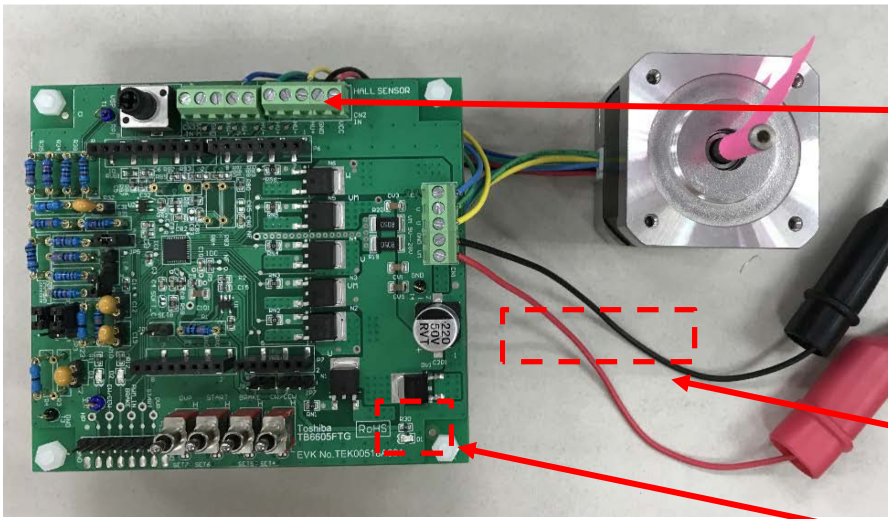


Figure 4-1 External View

5. Wiring Connection



Motor Cable

+5V	RED
HALL U	YELLOW
HALL V	GREEN
HALL W	BLUE
GND	BLACK
Phase U	YELLOW
Phase V	GREEN
Phase W	BLUE

Power supply
10-28V
2A current limited

If an attached motor does not rotate, check if the current is lower than 0.02A.

Figure 5-1 Wiring Connection

Note:
Phase U/V/W line of motor should be connected with U/V/W on reference model.
Do not mix Hall U/V/W line and Phase U/V/W line.

6. Switches

The Closed-loop system includes:

- Start switch for starting or stopping motor rotation
- Brake switch for stopping motor rotation in an emergency
- CW/CCW switch for setting rotation direction
- Speed control by potentiometer (see 6.1)

The settings are shown in below:

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

	H	**L**
CW/CCW	CCW	CW
OVP	Sine wave	Square wave

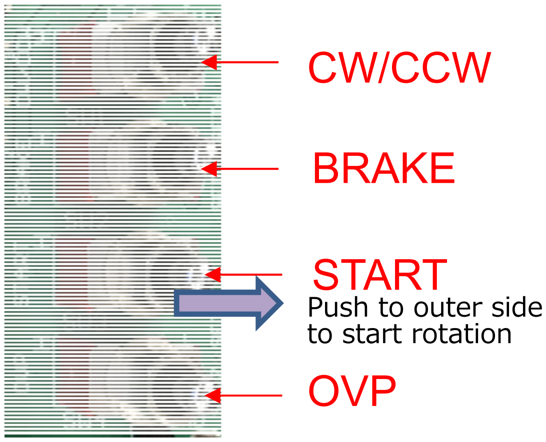
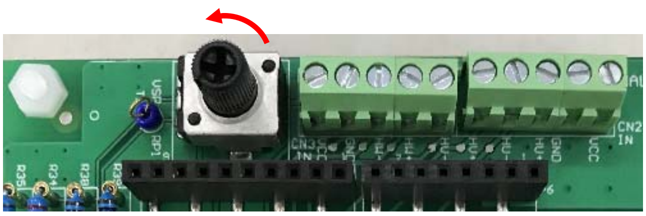


Figure 6-1 Switches

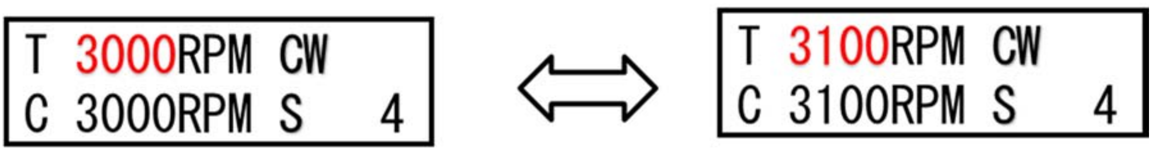
6.1. Potentiometer

Motor speed can be controlled using potentiometer.



In MCD + Op-Amp mode:
Target speed increases in a counter-clockwise direction

In MCD+MCU mode:
Target speed increases by 100RPM in a counter-clockwise direction.
LCD displays target speed is shown as below:



7. MCD+MCU Mode

7.1. Block Diagram in Case of MCD+MCU Mode

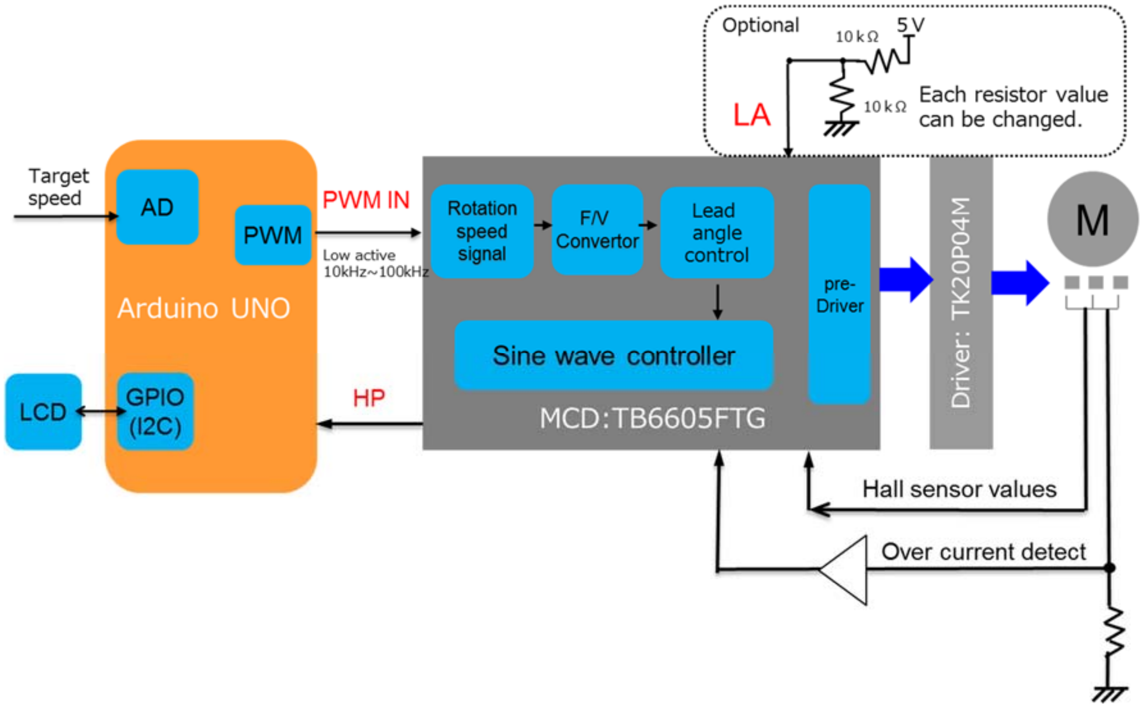


Figure 7-1 Block Diagram in MCD+MCU Mode

7.2. Hardware Connection

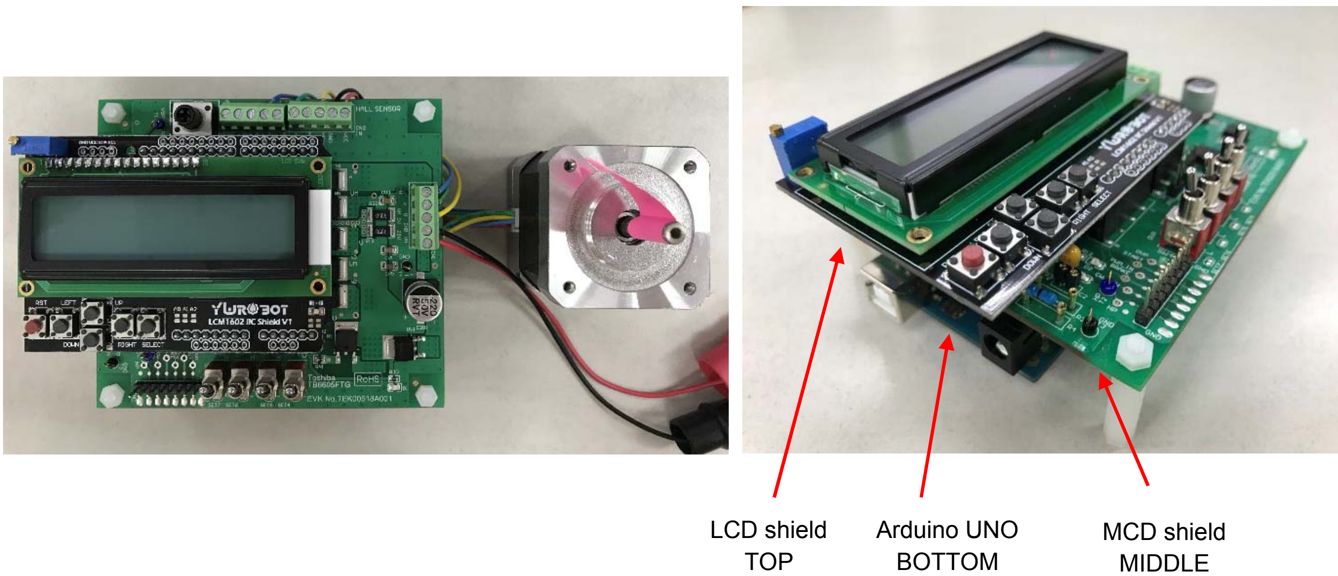


Figure 7-2 Hardware Connection

7.3. Jumper Setting

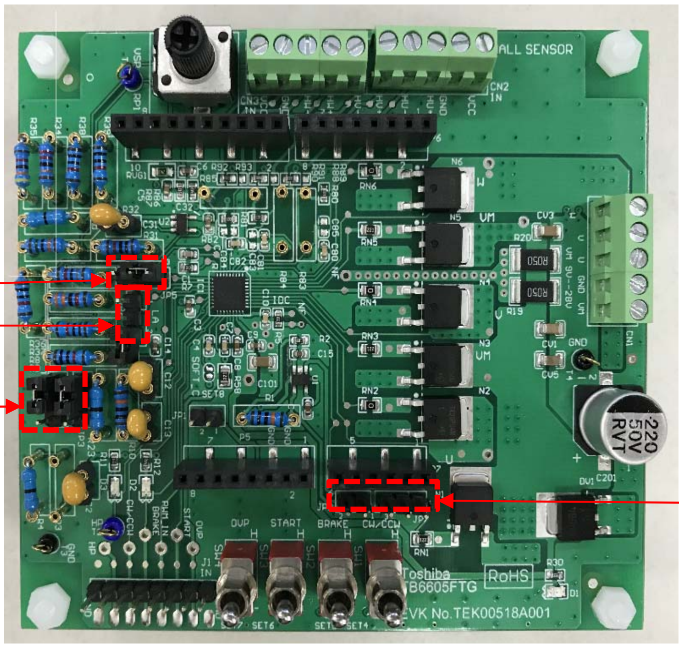
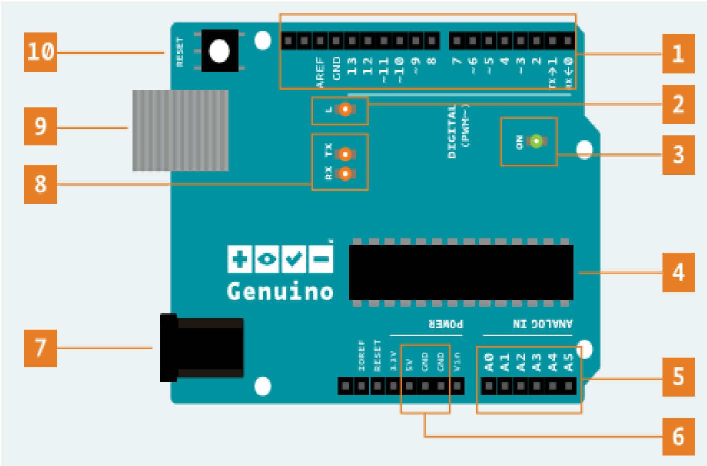


Figure 7-3 Jumper Setting

Table 7-1 Jumper Setting

Jumper	Setting	Function
JP1	Open	VREG(reserved)
JP2	Open	IDC: no connection with MCU (for future use)
JP3	Short pin2 and pin3	PWM-IN: input from MCU for speed control
JP4	Short pin2 and pin3	HP: hall sensor pulse output to MCU for detecting rotation speed
JP5	Short pin2 and pin3	DIF-IN: GND
JP6	Open	LA: not used (for future use)
JP7	Open	IDC: no connection with MCU (for future use)

7.4. Arduino Connection



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: for connection with LCD shield
D0, D1: for serial Tx/Rx, debug log out and set target speed

xxx: used for motor control
IDC, LA are not used in the current solution.

Figure 7-4 Arduino Connection

7.5. LCD Shield

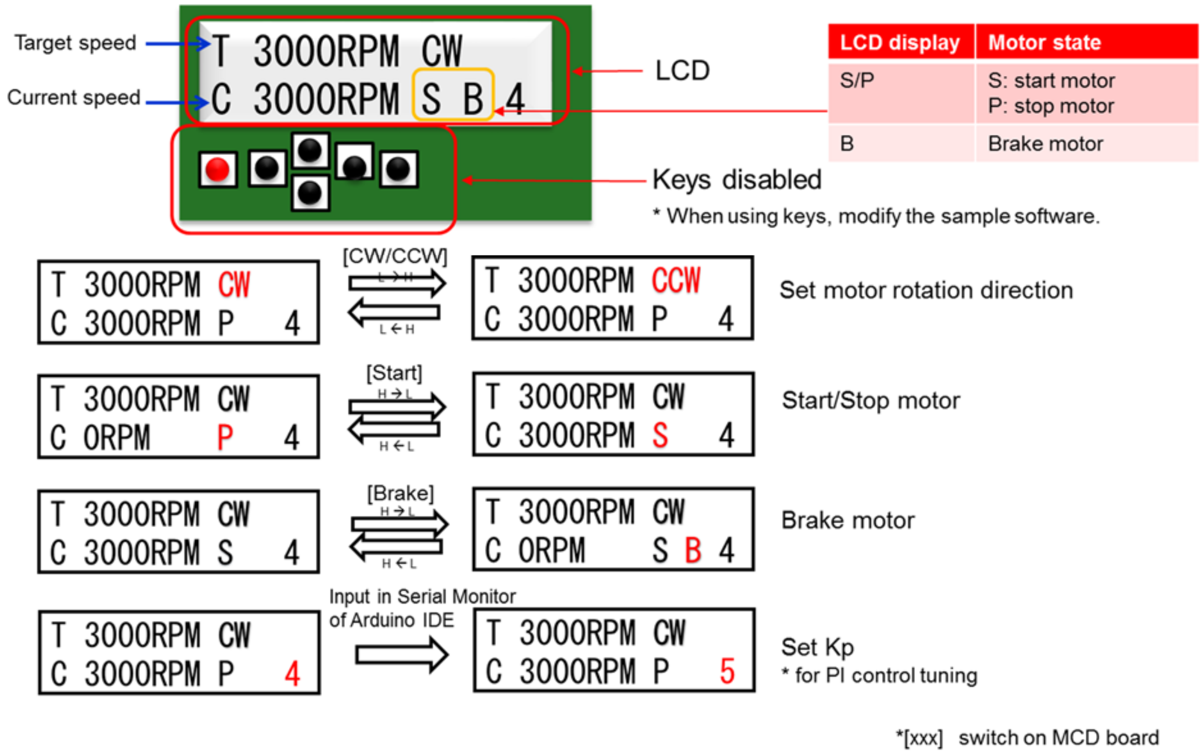


Figure 7-5 LCD Shield

7.6. Usage

Perform steps (1) to (11) in order.
This system includes a 42BLF BLDC motor manufactured by ACT motor.
Motors must be a Hall sensor BLDC type for normal working.

Table 7-2 Usage for MCD+MCU mode

Step	Item
(1)	Check the R/B exterior, set the switch correctly as described in 6.
(2)	Connect the motor to the board. (see 5)
(3)	Insert the R/B board to Arduino and insert the LCD board to the R/B board.
(4)	Connect the stabilized power supply to the R/B board.
(5)	Check the motor control switch. Set the start switch to “H”. Set the brake switch to “H”. Set the OVP switch as desired (“L”: square wave, “H”: sine wave). Set the CW/CCW switch as desired rotation direction. (“L”: CW, “H”: CCW)
(6)	Supply power to the R/B board.
(7)	Connect PC with Arduino board via USB cable. Upload the attached sketch to Arduino. (See 7.4)
(8)	Tune potentiometer on R/B board to set target speed which is displayed on LCD.
(9)	Start motor rotation by changing the start switch to “L”. Check if motor rotation speed displayed on LCD can achieve target speed.
(10)	Tune potentiometer on R/B board to change target speed which is displayed on LCD. Check if motor rotation speed displayed on LCD can achieve to the target speed.
(11)	Stop motor rotation by changing Start switch to “H”.

To rotate the motor, shift the start switch of the shield to “L”.
Current rotation speed is displayed as “C” parameter of the LCD module.

After rotation is started, rotation speed is controlled to maintain the current speed even power supply voltage is changed.

7.7. Software on Arduino UNO

The latest Arduino Software (IDE) can be downloaded here:

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

Read the appropriate procedure for your system.
Please download and install it.
Run the Arduino IDE and open the provided example sketches.

Figure 7-6 Arduino IDE

7.8. Arduino Sketch

A sketch is the name that Arduino uses for a program. This system needs a sketch “closed_loop_mcd.ino” to implement closed-loop system.

closed_loop_mcd.ino includes 3rd party libraries:
− #include <LiquidCrystal_I2C.h>
− #include <TimerOne.h>
* Refer to next page for how to install 3rd party libraries.

LiquidCrystal_I2C: Display characters on LCD shield
TimerOne: Use hardware Timer 1 for running an periodic interrupt function

7.8.1. How to Install

1. Unzip LiquidCrystal_I2C.zip and TimerOne.zip
2. Copy folder LiquidCrystal_I2C and TimerOne to
   "…\Documents\Arduino\libraries"
3. Check if library is installed:
   Arduino IDE->Sketch->Include Library-> LiquidCrystalI2C
   Arduino IDE->Sketch->Include Library-> TimerOne

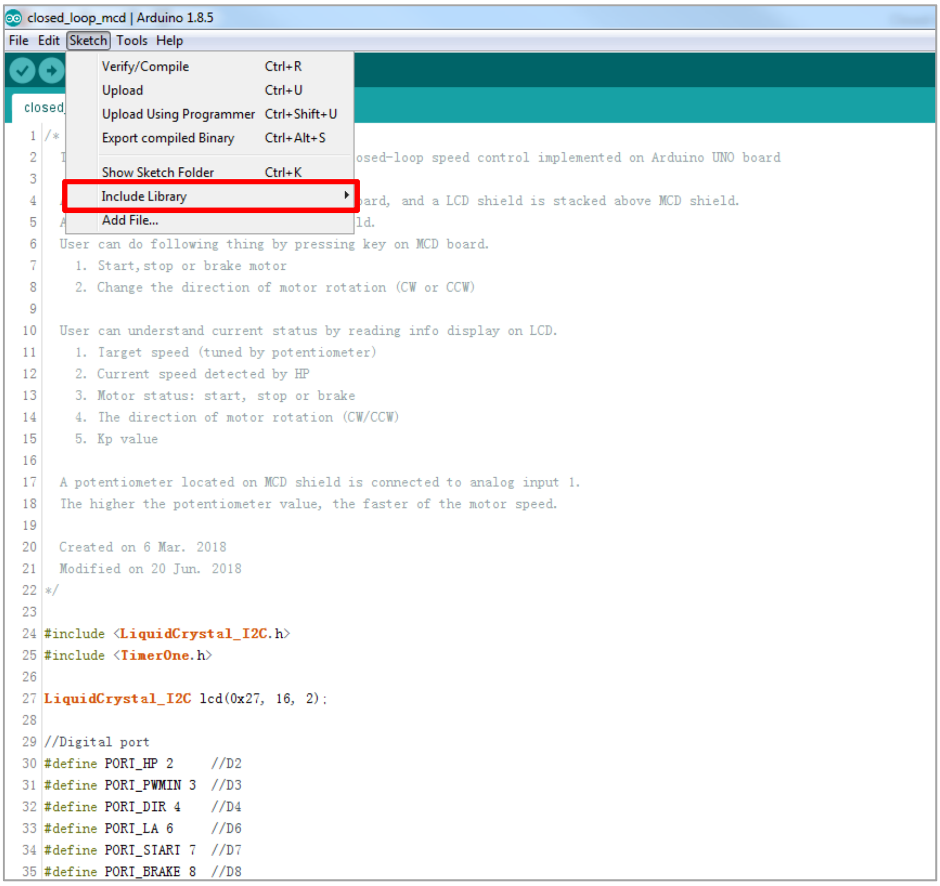


Figure 7-7 Install

7.9. Connection with Arduino UNO

1. Open closed_loop_mcd.ino using Arduino IDE
2. Connect the PC with Arduino board with USB cable
3. Select Board and Port

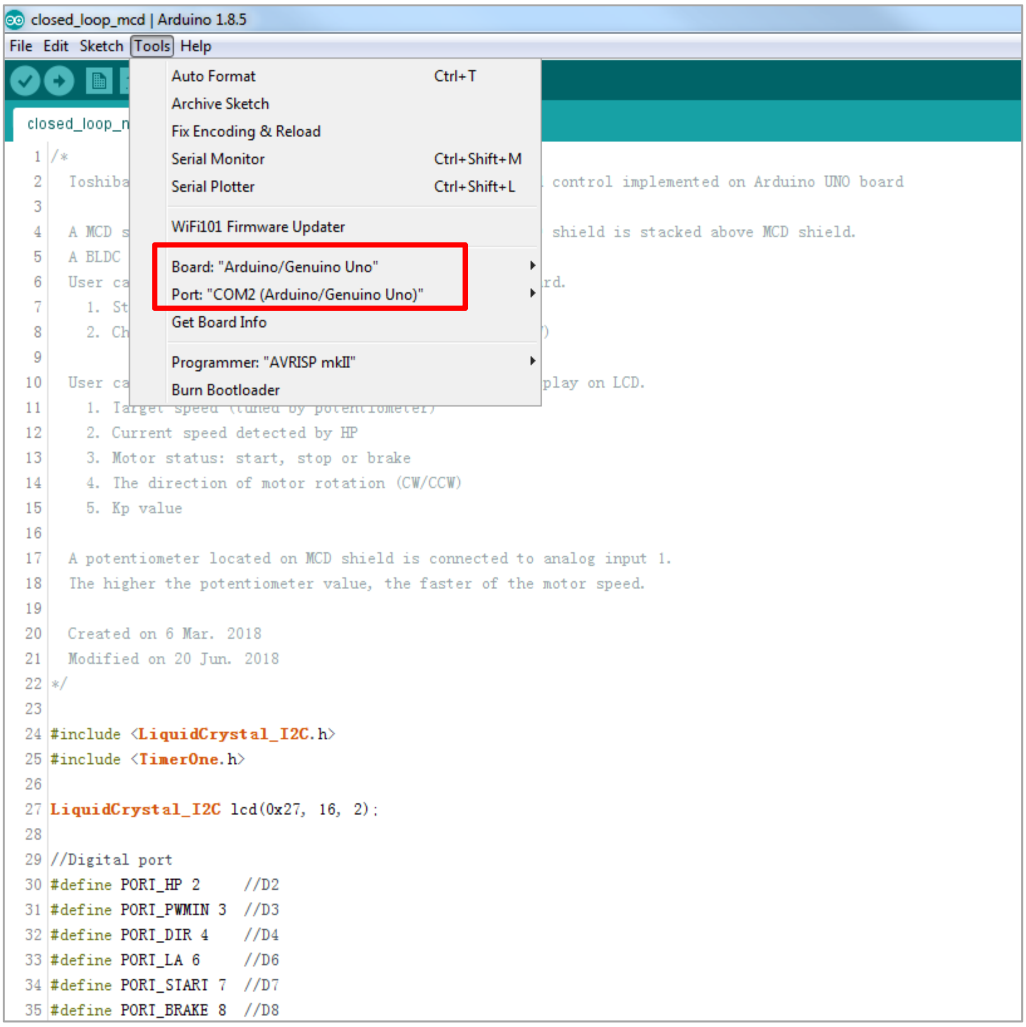


Figure 7-8 Connection with Arduino UNO

7.10. Code Uploading to Arduino UNO

1. Verify/compile code till no errors reported
2. Upload program to Arduino board by USB cable

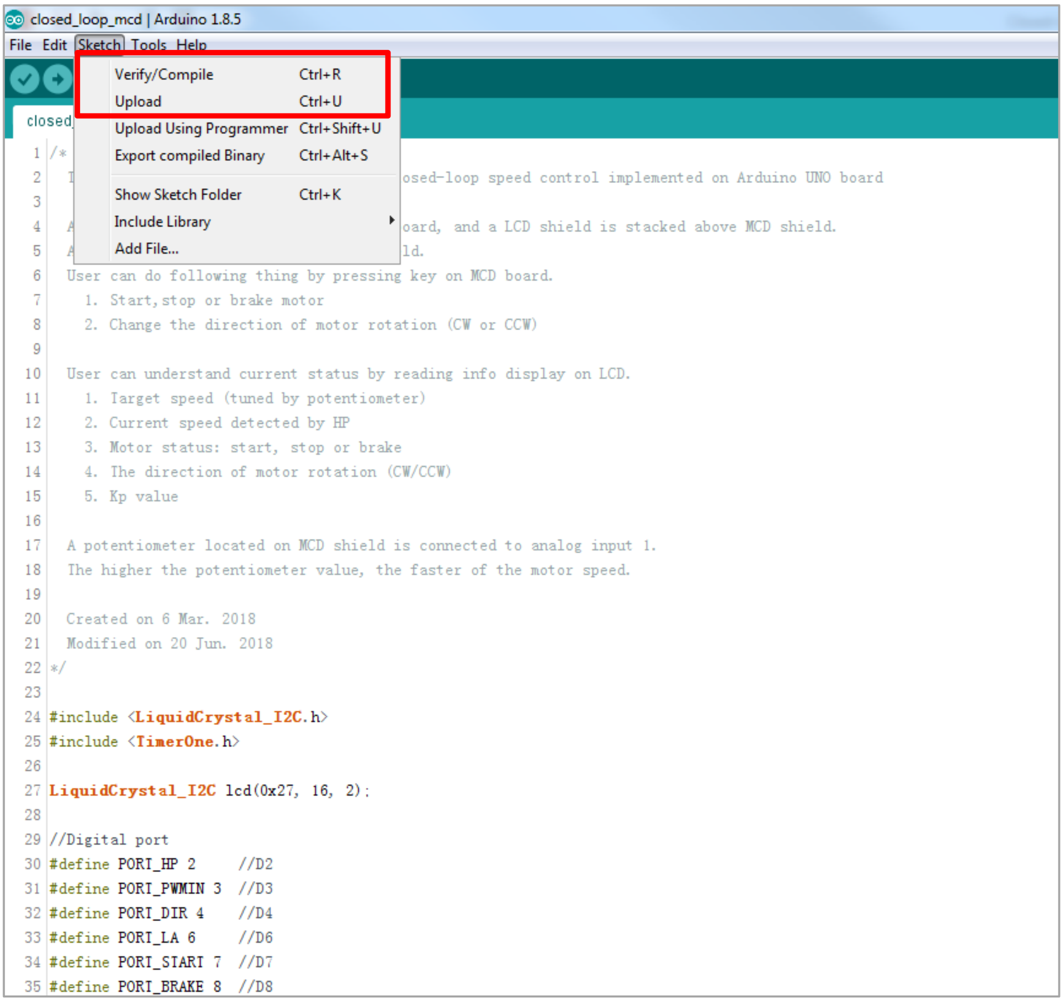


Figure 7-9 Code Uploading to Arduino UNO

7.11. Serial Monitor of Arduino IDE

Monitor Serial output for debugging

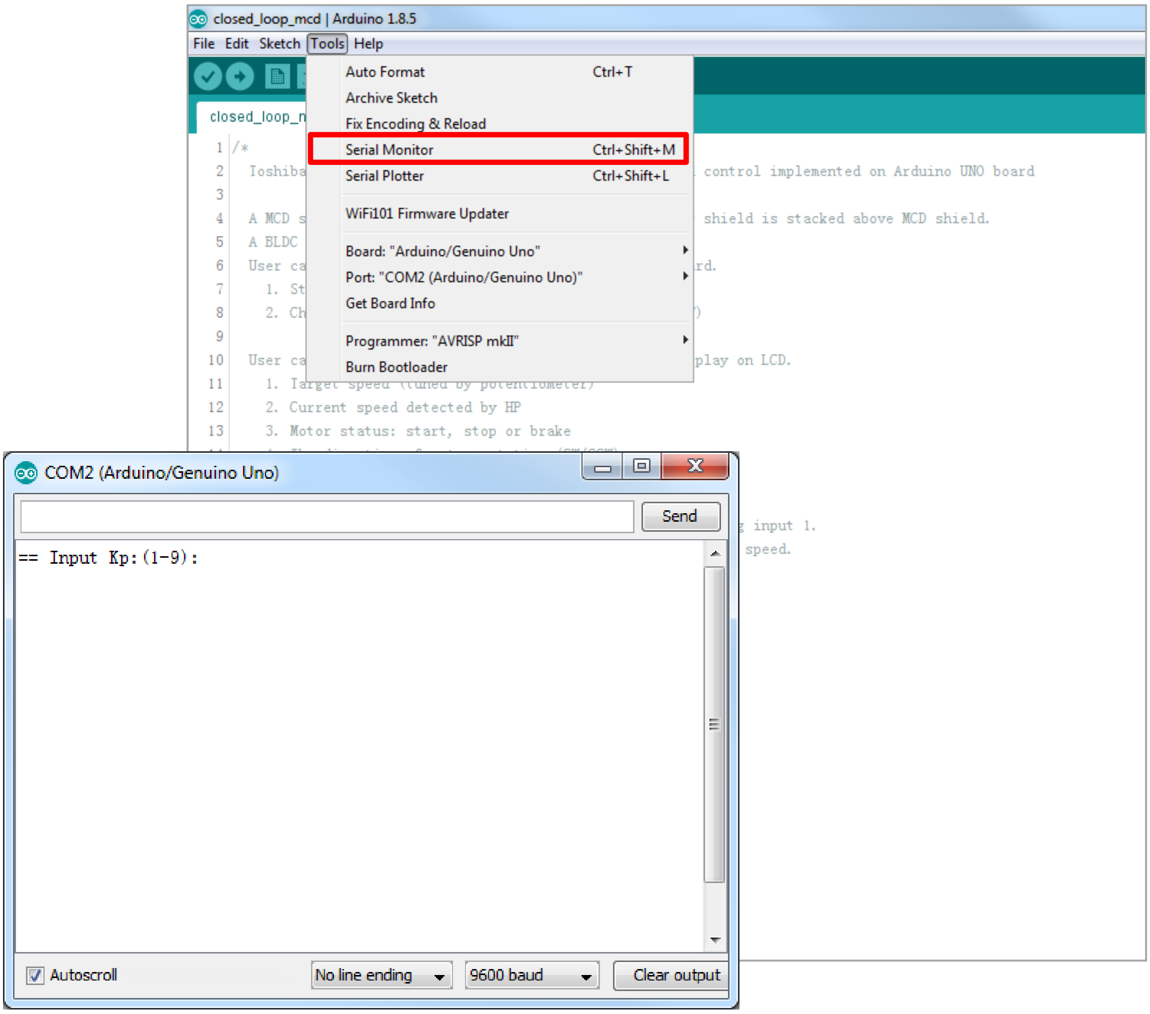
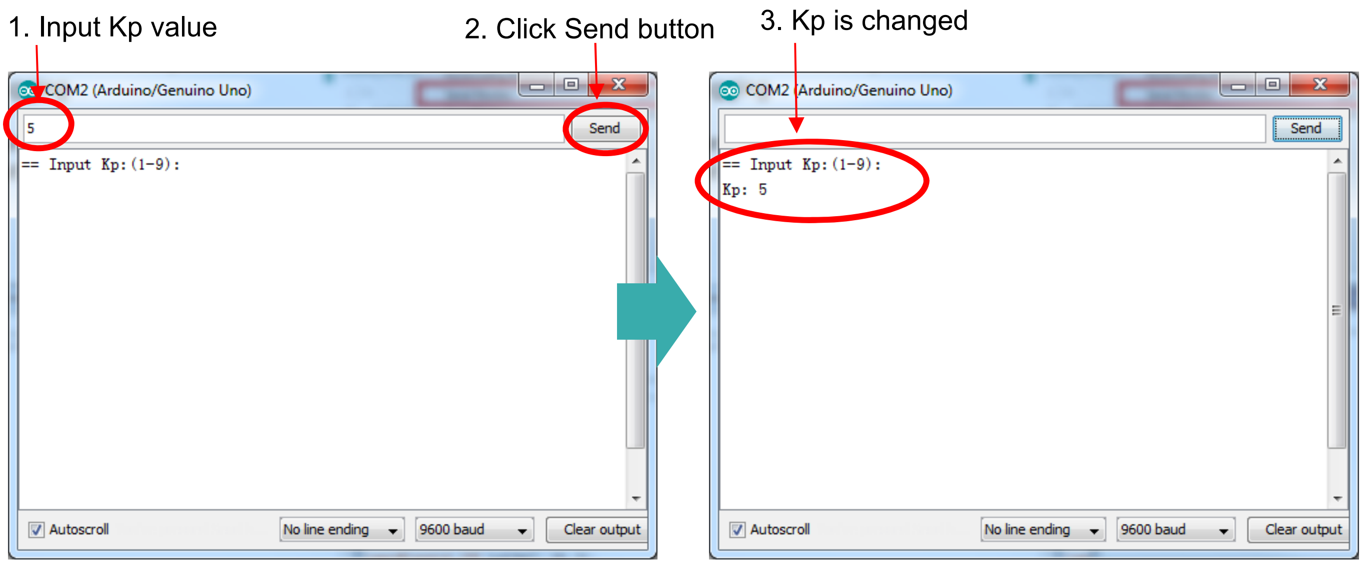


Figure 7-10 Serial Monitor

7.12. How to Change Kp in Serial Monitor

A Kp value can be changed using Serial Monitor for PI control.



When a Kp value is changed, upload the revised sketch to Arduino again.

Figure 7-11 Changing Kp value

7.13. Selected LCD Shield

- HW
  Use three signal pins of Arduino connectors.

- SW
  Library “LiquidCrystalI2C” is dedicated for the LCD shield in this R/B.

7.13.1. Other Type Shield

- HW
  May use more signal pins of Arduino connectors.
  Please refer to pin assignment in 7.9. Do not use pins for motor and serial.

- SW
  Use a library provided by vender of your LCD shield and modify the software accordingly.

7.14. Motor Speed Control for MCD + MCU

PID control is used to implement closed-loop control through motor speed feedback to achieve target speed in case MCD controlled by Arduino UNO.

7.14.1. PID Algorithm

Apply incremental PI control, and D is not used for standard PID calculation.

CV(t) = Kp*E(t)+Ki* ∫ E(t)dt+Kd*(dE(t)/dt
simplified incremental PI calculation

CV(t) = CV(t-1) + Kp*(E(t)-E(t-1)) + Ki*E(t)
E(t) = Set point – Input

- Input – Process variable
  - Motor rotation speed detected by hall sensor (HP signal)
- Output – Control variable
  - PWM duty (0-255, low active) (PWM IN signal)
- Set point
  - Target speed set by tuning potentiometer.

7.14.2. PID Control

P (Proportional term): CV(t) = Kp*E(t)

If Proportion Gain Kp is fixed, controller output (CV) is proportional to error E (t). If error is bigger, CV will become bigger automatically. If Kp is too big, the system will become unstable, on the other, if Kp is too small, the system will become less responsive.

Because the control output (CV) of Proportional term depends on error between SP and PV, a steady-state error (droop) is exist if use Proportional control only. To mitigate the droop, a compensating bias term (called offset) need to be set in advance. The best measures is add an Integral term into system.

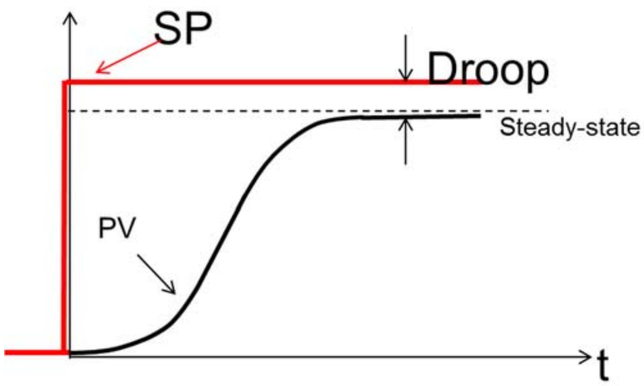


Figure 7-12 PID Control

If Kp is increasing:

- Rise time: decrease
- Overshoot: increase
- Settling time: small change
- Droop: decrease
- Stability: degrade

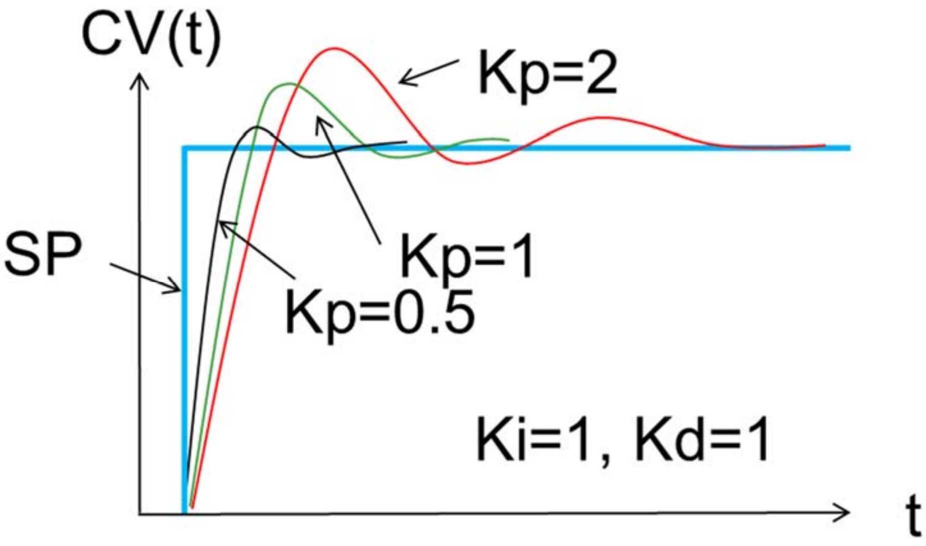


Figure 7-13 Increase of Kp Value

If Ki is increasing:

- Rise time: decrease
- Overshoot: increase
- Settling time: increase
- Droop: Eliminate
- Stability: degrade

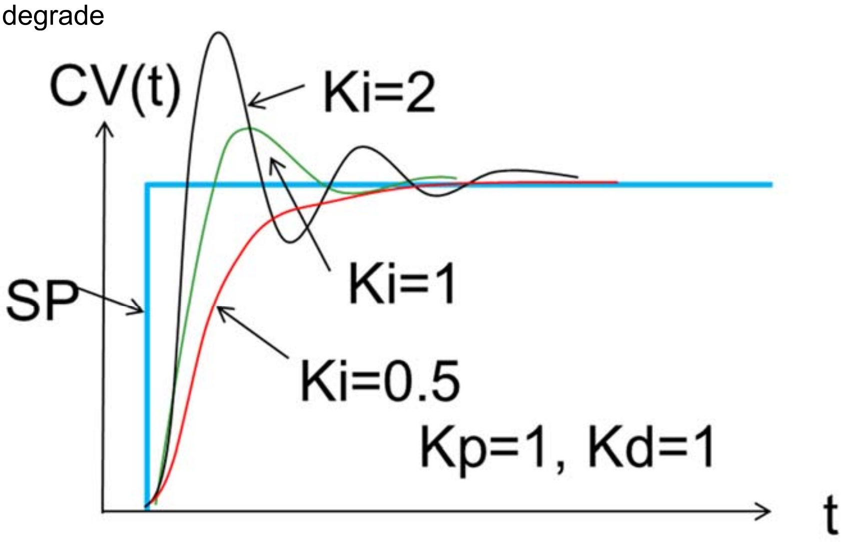


Figure 7-14 Increase of Ki Value

7.14.3. PI Tuning Procedure

1. Keep Ki and Kd values to zero at first, increase the Kp until the output of the loop oscillates. Then the Kp should be set to about half of that value for a "quarter amplitude decay" type response.

2. Then increase Ki until any offset is corrected in sufficient time for the process. However, too much Ki will cause instability.

3. The tuning is a repeat process to approach the balance between the response speed and system stability.

7.14.4. Source code

```
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 Mode

In MCD + Op-Amp mode, the speed control part realizes the closed loop by using PI control.

・ The VSP is used to determine the target speed.
・ The LA voltage is used as a feedback signal to represent current speed.
・ The Dif-in is used to adjust output duty to control motor speed.
・ PWM-in should be connected to GND.

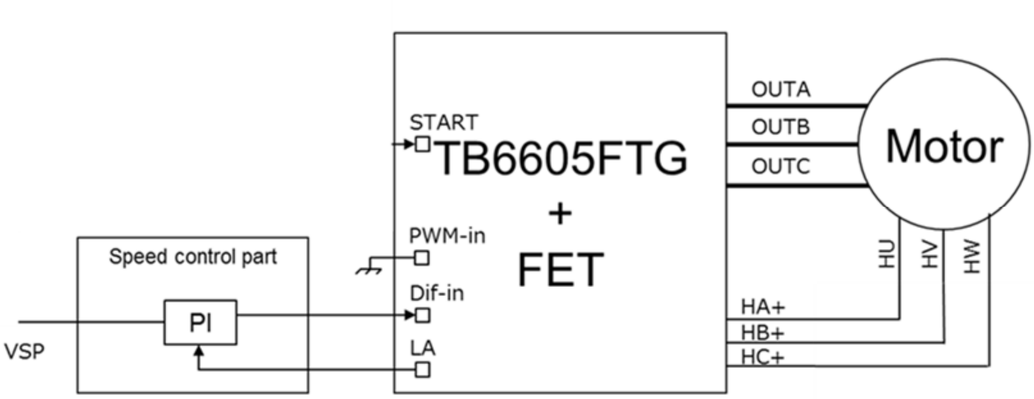


Figure 8-1 MCD + Op-Amp Mode

8.1. Block Diagram In Case of MCD＋Op-AMP

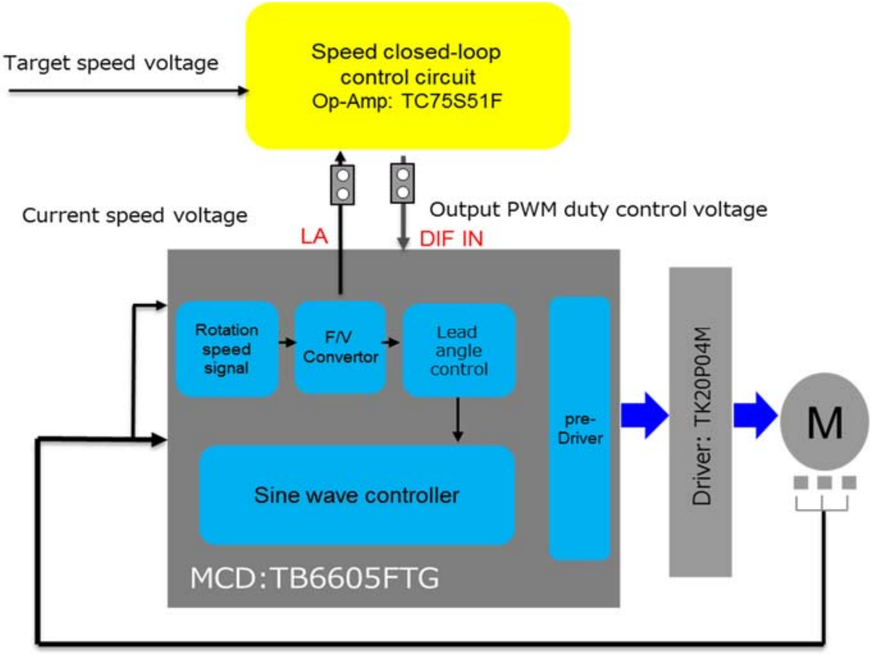


Figure 8-2 Block Diagram In Case of MCD＋Op-Amp

8.2. Usage

Perform steps (1) to (8) in order.
The product includes a 42BLF BLDC motor manufactured by ACT motor.
A motor must be a Hall sensor BLDC type for normal working.

Table 8-1 Usage for MCD + Op-Amp mode

Step	Item
(1)	Check the R/B exterior, set switch correctly as described in 6.
(2)	Connect the motor to the board. (see 5)
(3)	Connect the stabilized power supply to MCD board.
(4)	Check the motor control switch. Set the start switch to “H”. Set the brake switch to “H”. Set the OVP switch as desired. (“L”: square wave, “H”: sine wave). Set the CW/CCW switch as desired rotation direction. (“L”: CW, “H”: CCW)
(5)	Supply power and check if the power LED is ON.
(6)	Set the start switch to “L” to start rotating the motor.
(7)	Tune potentiometer to change a motor rotation speed. Check how motor rotation speed changes by monitoring Test point “HP”.
(8)	Set the start switch to “H” to stop rotating the motor.

If an Error Occurs

- Power supply LED no “lighting”: Check power supply cable and voltage.
- Motor no rotation: Check motor connection and Motor Control switch set.

8.3. OPERATING PROCEDURE

8.3.1. HW setting for MCD + Op-Amp

8.3.2. Jumper Setting

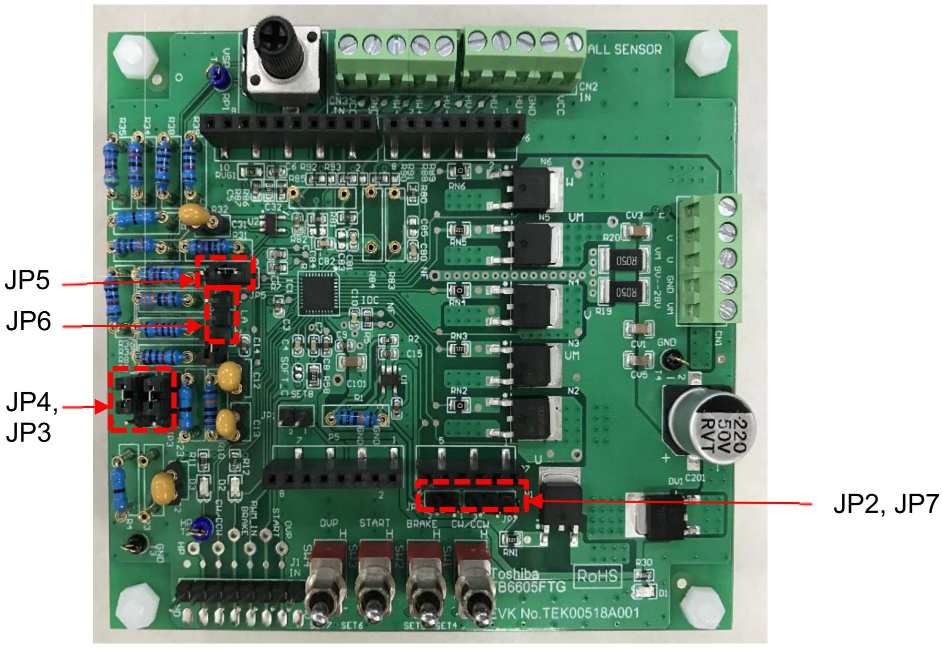


Figure 8-3 Jumper Setting

Table 8-2 Jumper Setting

Jumper	setting	Function
JP1	Open	VREG(reserved)
JP2	Open	IDC: no connection with MCU
JP3	Short pin1 and pin2	PWM-IN: GND
JP4	Open or short pin1 and pin2	HP: not connection with MCU
JP5	Short pin1 and pin2	DIF-IN: OpAmp output for speed control
JP6	Short pin1 and pin2	LA: OpAmp input
JP7	Open	IDC: not connection with MCU

8.4. Speed Control Part

Use a rail-to-rail Op-Amp.

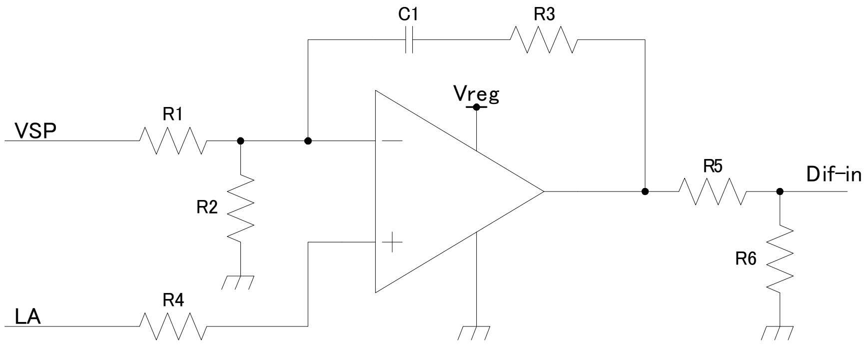


Figure 8-4 Speed Control Part

8.4.1. Parameter adjustment

R1 and R2:
if VSP=VMAX represents max speed, and LA=VLAMAX at max speed.

Adjust R1 and R2 so that $LAMAX=VMAX\times R2/(R1+R2)$

R3: The factor of proportional control Kp is related to R3. Increase R3 to increase Kp.

C1: The factor of integral control Ki is related to C1. Increase C1 to increase Ki.

R5 and R6 :
Adjust R5 and R6 to ensure Dif-in will not exceed 3V.

If Dif-in=VMIN makes motor run at min speed, it is recommended to set R5 and R6 so that $MIN<5\times R6/(R5+R6)<3$.

Total resistant of R5 and R6 should be around 50k to 100k.

8.5. Auto Lead Angle Setting

For high efficiency, TB6605FTG needs to be set a lead angle parameter.
When you use a different motor, change a lead angle parameter if necessary.
And also change the TCR parameter for good lead angle value.

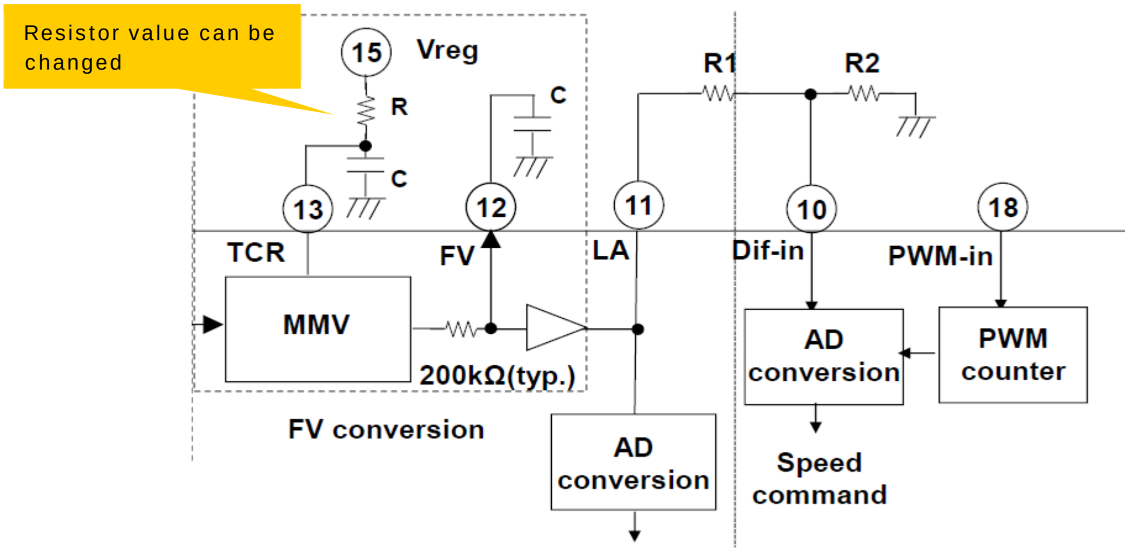


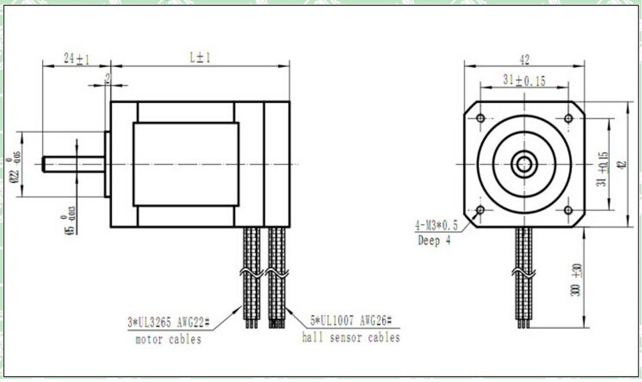
Figure 8-5 Auto Lead Angle Setting

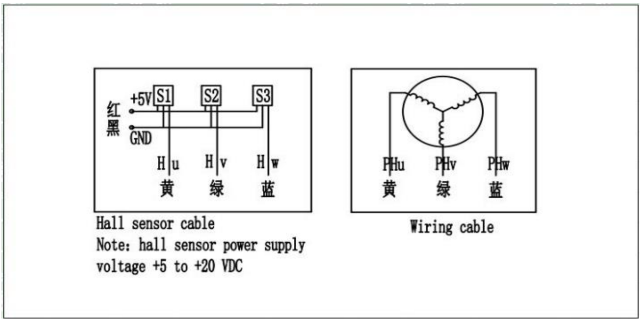
Usage Notes

- To ensure that power supply is at the range of 10V to 28V, and current is under 2A (limited by Motor spec).
- At high current working states, motor and control board may have a high temperature.
- Some of the unused pins of the product have not been processed in any way.
  Make sure to process these pins as appropriate for your system.
- Resistors and capacitors in pin sockets can be changed if necessary when other than the attached motor is used.

9. ATTACHED BRUSHLESS MOTOR PARAMETERS

外型图 (Mechanical Dimensions)：





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————————120º 电角度 Electrical angle

绝缘等级 Insulation Class——————————B

环境温度 Ambient Temperature Range——————————-20℃~+50℃

绝缘电阻 Insulation Resistanc——————————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
额定力矩 Rated Torque N-m	0.063
额定电流 Rated Current Amps	1.9
输出功率 Output Power Watts	26
峰值力矩 Peak Torque N-m	0.18
峰值电流 Peak Current Amps	5.7
力矩常数 Torqueconstant N-m/Amps	0.035
反电势常数 Back EMF V/Krpm	3.7
转动惯量 Rotor Inertia g.cm2	24
机身长度 Body Length mm	47
重 量 Mass kg	0.29

Terms of use

This terms of use is made between Toshiba Electronic Devices and Storage Corporation ("We") and customers who use documents and data that are consulted to design electronics applications on which our semiconductor devices are mounted ("this Reference Design"). Customers shall comply with this terms of use. Please note that it is assumed that customers agree to any and all this terms of use if customers download this Reference Design. We may, at its sole and exclusive discretion, change, alter, modify, add, and/or remove any part of this terms of use at any time without any prior notice. We may terminate this terms of use at any time and for any reason. Upon termination of this terms of use, customers shall destroy this Reference Design. In the event of any breach thereof by customers, customers shall destroy this Reference Design, and furnish us a written confirmation to prove such destruction.

1. Restrictions on usage
    1. This Reference Design is provided solely as reference data for designing electronics applications. Customers shall not use this Reference Design for any other purpose, including without limitation, verification of reliability.
    2. This Reference Design is for customer's own use and not for sale, lease or other transfer.
    3. Customers shall not use this Reference Design for evaluation in high or low temperature, high humidity, or high electromagnetic environments.
    4. This Reference Design shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
    5.
2. Limitations
    1. We reserve the right to make changes to this Reference Design without notice.
    2. This Reference Design should be treated as a reference only. We are not responsible for any incorrect or incomplete data and information.
    3. Semiconductor devices can malfunction or fail. When designing electronics applications by referring to this Reference Design, customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of semiconductor devices could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Customers must also refer to and comply with the latest versions of all relevant our information, including without limitation, specifications, data sheets and application notes for semiconductor devices, as well as the precautions and conditions set forth in the "Semiconductor Reliability Handbook".
    4. When designing electronics applications by referring to this Reference Design, customers must evaluate the whole system adequately. Customers are solely responsible for all aspects of their own product design or applications. WE ASSUME NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.
    5. No responsibility is assumed by us for any infringement of patents or any other intellectual property rights of third parties that may result from the use of this Reference Design. No license to any intellectual property right is granted by this terms of use, whether express or implied, by estoppel or otherwise.
    6. THIS REFERENCE DESIGN IS PROVIDED "AS IS". WE (a) ASSUME NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (b) DISCLAIM ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO THIS REFERENCE DESIGN, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.