BRUSHLESS DC MOTOR DRIVER

BLR800A
User manual

Giden Electronics

Всё для машиностроения и автоматизации

www.giden.ru

sale@giden.ru

+7 (495) 225-54-52

1 Introduction

BLR800A brushless DC motor driver is a high-performance, low-cost brushless driver for low-voltage 800W brushless DC motors. This brushless DC driver supports Modbus communication protocol, providing users with more flexible choices in practical applications.

1. 1 Advantges

- Support Modbus communication protocol, suitable for touch screen or PC
- Built-in potentiometer RV speed control, external analog speed control, external potentiometer speed control, PWM speed control
- Input voltage 12~60VDC
- Working temperature -25℃~+50℃
- Optional speed control in open and closed loop modes
- Works for 800W BLDC motor or below
- 5 seconds waiting time for locked-rotor

2 Electrical performance and environmental indicators

2.1 Electrical Specifications

Parameter	Min.	Typical	Max.
Input VoltageDC(V)	12	48	60
Peak Current (A)	-	21.5	23
Rotate Speed(rpm)	100①	-	-

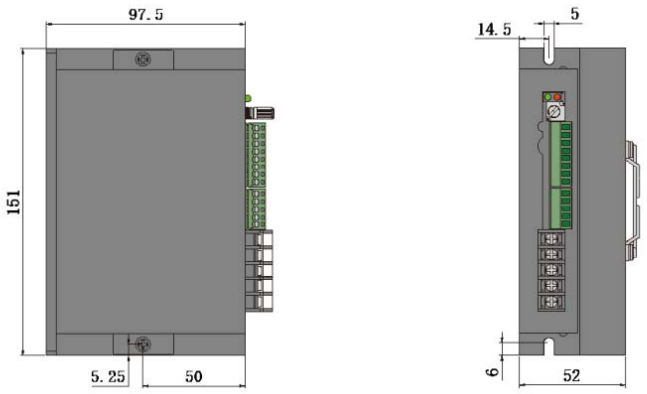
① The minimum value requires that the pole pair number set by the driver is consistent with the pole pair number of the motor. This value is also affected by the characteristics of the motor itself and it is normal for it to be different.

2. 2 Environmental indicators

Factors	Indicators
Cooling method	Natural cooling or forced cooling
Occasions	Avoid dust, oil and corrosive gases
Operating Temperature	-25℃~+50℃
Storage Temperature	-30℃~+70℃

3 Demmension and interface

3. 1 Dimension (Unit: mm)

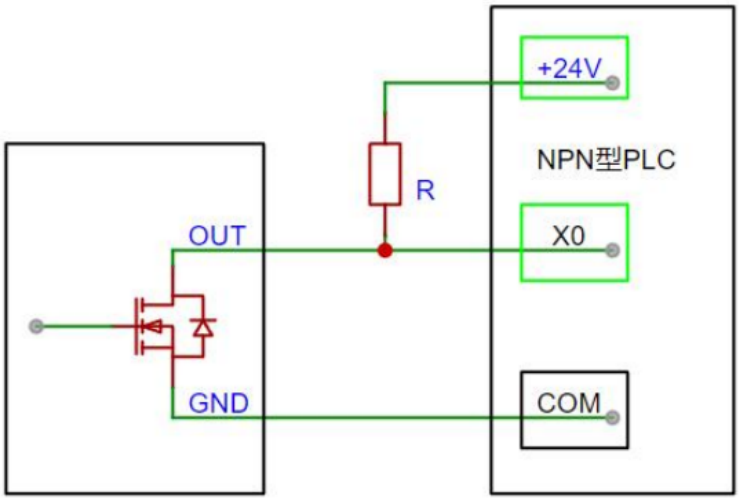


3. 2 Port signal description

Signal	Terminal	Function
Power Input	VCC	DC+
	GND	DC-
Motor Connection	W	Motor line W phase
	V	Motor line V phase
	U	Motor line U phase
Hall Signal	GND	Hall sensor signal ground electrode
	HW	Hall sensor signal HW
	HV	Hall sensor signal HV
	HU	Hall sensor signal HU
	5V	Hall sensor signal power supply
Control Signal	5V	Control signal power supply
	SV	①External potentiometer ② External analog signal input
	COM	BRK/FR/PWM interface common terminal (PNP type input interface)
	GND	Control signal ground wire
	F/R	F/R disconnected GND, motor rotates CW,otherwise motor rotates CCW（(PNP type input interface)
	BRK	BRK disconnected GND, motor runs, otherwise motor stops(NPN input)
	PWM/F	Pulse speed regulation
	OUT	**Alarm output interface, open drain output, requires an external pull-up resistor R (recommended resistance 2K-10K)**

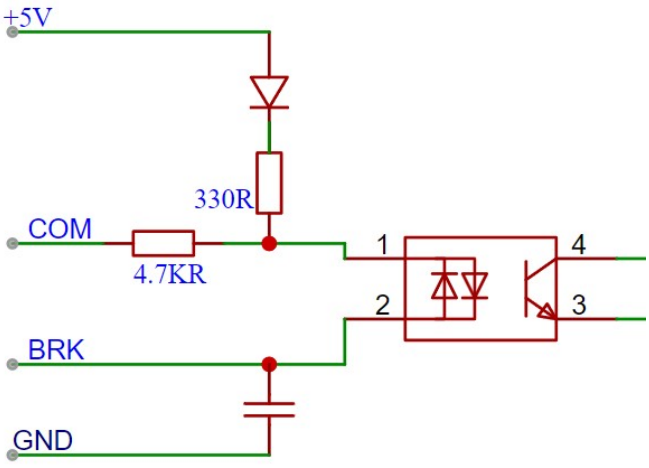
3. 4 Wiring method between OUT terminal and PLC

The internal structure of the OUT terminal of this driver is as follows, which is related to the wiring method of the PL C and the type of the PLC input terminal. For example, Mitsubishi NPN type PLC wiring.



Connect the wires according to the wiring method shown in the figure above. When an abnormality occurs, the OUT port outputs a low level. When there is no abnormality in the driver, the OUT port outputs a high impedance.

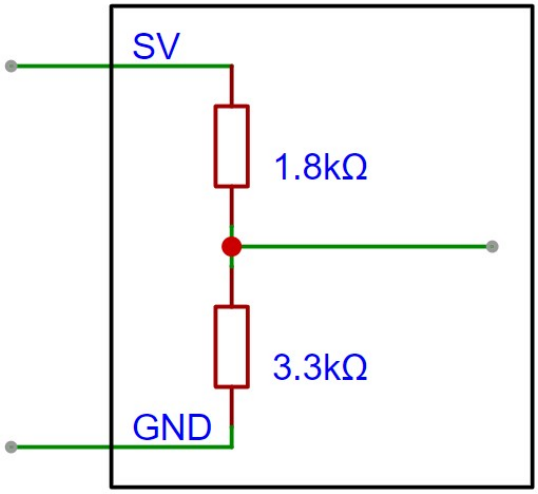
3.5 BRK,F/R,PWM Interface internal circuit



The interface circuits of BRK, F/R, and PWM are consistent with the above schematic diagram.

3. 6 SV analog input interface internal circuit

SV signal tolerates 0-5V speed regulation.



4 Analog speed regulation

4. 1 Built-in analog regulation

After changing the speed adjustment mode to analog adjustment through the host computer. Rotate the built-in speed potentiometer RV clockwise, and the motor starts to run after a "click". Continue to rotate clockwise, and the motor speed increases . Rotate the built-in speed potentiometer RV counterclockwise, and the motor speed decreases; continue to rotate counterclockwise to the limit position after a "click". At this time, the built-in speed potentiometer RV is closed and the motor stops running.

4.2 External analog quantity adjustment

When it is necessary to switch to the external analog speed control mode, the built-in potentiometer RV must be in the c losed state. That is, the built-in potentiometer RV is rotated counterclockwise to the limit position after a "click" sound.

Speed regulation mode: SV port inputs 0-5V voltage signal

5 Current limit adjustment

The function of current limit value adjustment is to limit the peak value of the motor phase line current value, thereby protecting the motor. The principle of protection is that before the motor winding is installed in the motor stator core slot, a layer of insulating paint will be applied on its surface. If the temperature of the winding is too high, the insulating paint will age or fall off, causing the copper wire to contact the core, resulting in a large current and burning the motor (this principle is also used when disassembling the motor, passing a constant current through the winding to heat the winding, so that the glue between the winding and the core melts). Adjusting the current limit value can limit the heat generated by the winding, thereby protecting the motor.

The limit value of the driver bus current can be adjusted by the potentiometer P_SV. When P_SV is adjusted clockwise to the maximum value, the bus current value is 21.5A. Since there is a certain error between the set output peak current and the actual output peak current, for safety reasons, please appropriately reduce the set output peak current.The limit value of the driver bus current can be adjusted by the potentiometer P_SV. When P_SV is adjusted clockwise to the maximum value, the bus current value is 21.5A. Since there is a certain error between the set output peak current and the actual output peak current, for safety reasons, please appropriately reduce the set output peak current.

6 Acceleration and deceleration time value adjustment

The acceleration time and deceleration time of the motor are set by the potentiometer ACC/DEC. The acceleration and deceleration time can be increased or decreased by rotating the potentiometer ACC/DEC left or right. Setting range: 0.3~10S. The acceleration time is the time required for the motor to reach the rated speed from a stationary state, and the deceleration time is the time required for the motor to stop from the rated speed.

(Note: This is not the actual speed of the motor, but the time it takes for the speed value to change. The actual speed of t he motor will take different times to reach the maximum speed due to different loads or motors. This time is only for referen ce of speed regulation)

7 Status indication and exception handling

7. 1 Status Indicator

When the motor experiences overvoltage, Hall signal error, stall, overtemperature, etc., the driver will send out an alarm signal and stop working.

Alarm indicator	Status description
Red light flashes twice, motor stops	Driver components over temperature (over 85)
Red light flashes 3 times, motor stops	The power supply voltage exceeds 60V
Red light flashes 4 times, motor stops	The power supply voltage is lower than 9.8V
Red light flashes 5 times, motor stops	The driver cannot receive the Hall signal or receives an incorrect Hall signal.
Red light flashes 8times, motor stops	The motor is stalled and reaches the stall alarm time

Red light flashes 10-25 times	Other hardware circuit abnormalities

7. 2 Exception handling

When the above abnormality occurs, you should first check the relevant reasons, then send a reset command to the driver to eliminate the alarm signal. If the alarm signal cannot be eliminated, proceed as shown in the table below.

The reset command refers to one of the following instructions

- ◆Short-circuit the BRK port on the driver board with GND again;
- ◆485 Send a stop command or reset the alarm once;
- ◆Restart the power supply.

LED display	Issue handling
Red light flashes twice	Please check whether the operating temperature is too high
Red light flashes 3 times	Please check the power supply voltage
Red light flashes 4 times	Please check the power supply voltage
Red light flashes 5 times	Please check whether the motor wiring is secure,ensure Hall signal line is not damaged.
Red light flashes 8 times	Check otor wire is not loose, ensure motor is not damaged and the current limit setting is too small
Red light flashes 10-25 times	Please power on again. If the alarm persists, replace the driver.

8 Modbus Communication

8. 1 DIP switch function (dip switch means 1, default is 0)

SW1	SW2	SW3	Server Address
0	0	0	Broadcast communication
1	0	0	0x01
0	1	0	0x02
1	1	0	0x03
0	0	1	0x04
1	0	1	0x05
0	1	1	0x06
1	1	1	0x07

This driver uses RJ12 interface. The wiring diagram of RJ12 interface is as follows:

8. 2 All holding register addresses of the server

The client can read all the server's holding register addresses:

Server address (1byte)	Function code (1byte)	Holding register access initial address (2byte)	Access data size (2byte)	CRC check (2byte)	Function description
0xnn	0x03	0x0040	0x0001	CRC Check	Read data return mode. (0 =return, 1 = no return)
0xnn	0x03	0x0056	0x0001	CRC Check	Read the drive setting speed (unit: rpm)
0xnn	0x03	0x0058	0x0001	CRC Check	Read the open-loop and closed-loop status of the driver. (0 = Closed loop, 1 = open loop.

0xnn	0x03	0x005F	0x0001	CRC Check	Read motor feedback speed (unit: rpm)
0xnn	0x03	0x0066	0x0001	CRC Check	Read driver whether enable; 0=enable state1=disable state
0xnn	0x03	0x006A	0x0001	CRC Check	Read driver whether brake, 0=brake ; 1=not brake
0xnn	0x03	0x006D	0x0001	CRC Check	Read driver rotate direction, 1=forward rotation ; 0=reverse rotation
0xnn	0x03	0x0076	0x0001	CRC Check	Read the alarm code，2=NTC1 detects that the PCB temperature over 85 ℃
0xnn	0x03	0x0086	0x0001	CRC Check	Read the set number of poles
0xnn	0x03	0x008A	0x0001	CRC Check	Read acceleration time setting value(unit:1ms)
0xnn	0x03	0x008C	0x0001	CRC Check	Read the deceleration time setting value (unit: 1ms)
0xnn	0x03	0x0092	0x0001	CRC Check	**Read the maximum speed of analog speed regulation (unit: rpm/min)**
0xnn	0x03	0x00B6	0x0001	CRC Check	Read RS-485 connection status
0xnn	0x03	0x00BB	0x0001	CRC Check	Driver program version, not used by users
0xnn	0x03	0x00C8	0x0002	CRC Check	Read the supply voltage value (unit: 0.1V)
0xnn	0x03	0x00D2	0x0002	CRC Check	Read temperature value (unit: 0.1)

The client can write to all the server's holding register addresses:

Server address (1byte)	Function code (1byte)	Holding register access initial address (2byte)	Register value (2 bytes)	CRC Check (2byte)	Function description	Default value	After writing 1 to 0X00BC, should the value be saved in the memory chip
0xnn	0x06	0x00B6	–	CRC Check	Write RS-485 connected state (the function of setting this parameter is deciding to follow which one when the driver receives both speed setting signals : built-in signal and signal from client); Write 0=485 control disable, client can only read holding register, but cannot write other holding register except 0x00B6; Write 1=485 control enable, client can read and write register.	0	No
0xnn	0x06	0x0040	–	CRC Check	Post-back function setting, which would affect communication rate, but be convenient to adjust machine. If write 0, then data would be post back.	0	Yes

					For an example: 0x0040 wrote 0, every time the PLC sending 06 function code command to driver, the driver would execute and post back data at once. 0x0040 write 1, driver would only execute but not post back.		
0xnn	0x06	0x0056	–	CRC Check	Set the motor speed (unit: rpm), the value range of this register is 0~60000(Note: If the acceleration time value has been set, please first write 0 to address 0x0066 to set the motor to disabled state.)	0	No
0xnn	0x06	0x0058	–	CRC Check	Set open-loop and closed-loop state. 0=closed loop, 1= open loop. (Note: Before modifying the open-loop and closed-loop state, stop the motor first)	0	Yes
0xnn	0x06	0x0066	–	CRC Check	Write 0: enable state; write 1: disable state	0	No
0xnn	0x06	0x006A	–	CRC Check	Write 1: brake state; write 0: not brake state	0	No
0xnn	0x06	0x006D	–	CRC Check	Write 1: CW rotation; write 0: CCW rotation	0	No
0xnn	0x06	0x0076	–	CRC Check	Write non-zero, register keep the original value Write 0: reset alarm (Note: Before resetting the alarm, stop the motor first)	0	No
0xnn	0x06	0x0086	–	CRC Check	Motor pole number setting, the value range of this register is 0~254, writing other values is invalid （(Note: Before resetting,stop the motor first)	4	No
0xnn	0x06	0x008A	–	CRC Check	Acceleration time setting (unit: 1ms), the parameter range is 0-65535, and other values are invalid. The definition of this parameter refers to the completion time of the given speed accelerating from 0 to 3000. For example, if the acceleration time is set to 12000, it means that the time for the driver's given speed to rise from 0 to 3000 is 12s. (Note: 1. In addition, the given speed is not the actual speed, and the actual speed will not be synchronized with the given speed. 2. Before setting the acceleration time value in the serial port, set the value of the potentiometer ACC/DEC to 0.)	0	Yes

0xnn	0x06	0x008C	–	CRC Check	Deceleration time setting (unit: 1ms), the parameter range is 0-65535, other values are invalid. The definition of this parameter refers to the completion time of the given speed from 3000 to 0. For example, if the acceleration time is set to 12000, it means that the time for the driver's given speed to drop from 3000 to 0 is 12s. (Note: 1. In addition, the given speed is not the actual speed, and the actual speed will not be synchronized with the given speed. 2. Before setting the deceleration time value in the serial port, set the value of the potentiometer ACC/DEC to 0.)	0	Yes
0xnn	0x06	0x0092	–	CRC Check	The maximum speed of analog speed regulation (unit: rpm), the value range of this register is 0-60000, and other values are invalid. The definition of this parameter is the actual speed of the motor when the SV pin inputs 5V. If the SV pin inputs 1V, the speed of the motor is the value of this parameter/5. In addition, this parameter has no effect on the speed regulation range of the client (touch screen and other devices)	3000	Yes
0xnn	0x06	0x00BC	–	CRC Check	Write 1 to save all the set parameters.	-	-
0xnn	0x06	0x00CC	–	CRC Check	Restore factory settings Write 1: Restore factory settings. The specific operation is to restore all parameters involved in writing in this table (excluding factory settings) to the default values, and then restore all saved values to the default values. For example, if the current saved value of the motor pole pair number (0x0086) is 2, it is restored to the default value of 4. After the restoration work is completed, the 0x06 holding register is reset to 0 Write 0: Do not restore factory settings or restore factory settings successfully (Note: Before restoring factory settings, first write 0 to address 0x0056 to set the speed to 0, and write 0 to address 0x0066 to disable the motor)	-	–

8. 3 Communication steps when the client is a PC or other device

This section is applicable to the case where the client is a PC or other device. If the user uses a touch screen to communicate with this driver, you can skip the reading of this section. Before communication, you need to have a certain understanding of the following two standards:

① GB/T 19582.1-2008：《Industrial Automation Network Specification Based on Modbus Protocol Part 1: Modbus Application Protocol》

② GB/T 19582.2-2008：《Industrial automation network specification based on Modbus protocol Part 2: Guidelines for the implementation of Modbus protocol on serial links》

When communicating, user needs to write the program according to following steps:

① Determine the address of the drive

② Write a program to send a single byte with 1 start bit and 2 stop bits according to the baud rate

③ Write CRC verification program

④ The structure of ADU is determined according to the function, and the ADU is sent

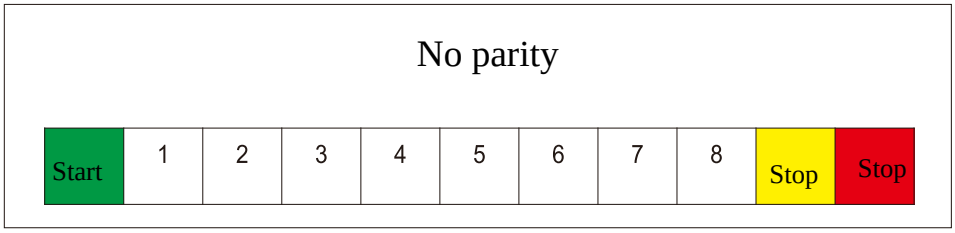
⑤ Analyze the data according to the ADU returned by the drive

The address of driver could refer to section 9.3 of this manual, the detailed description for 2 ~5 is followed as below:

Write a program to send a single byte with start bit, check bit, and stop bit at the same bit rate:

About the details of this portion, users can refer to part 6.5.1 of national standard GB/T 19582.2-2008.

*Notice: each character or byte is sent from left to right: LSB MSB



In the figure above, the start signal of a single data packet of serial communication is represented by a logic 0 data bit, and the two stop signals of the data packet are represented by logic 1 data bits.

● Write CRC verification program

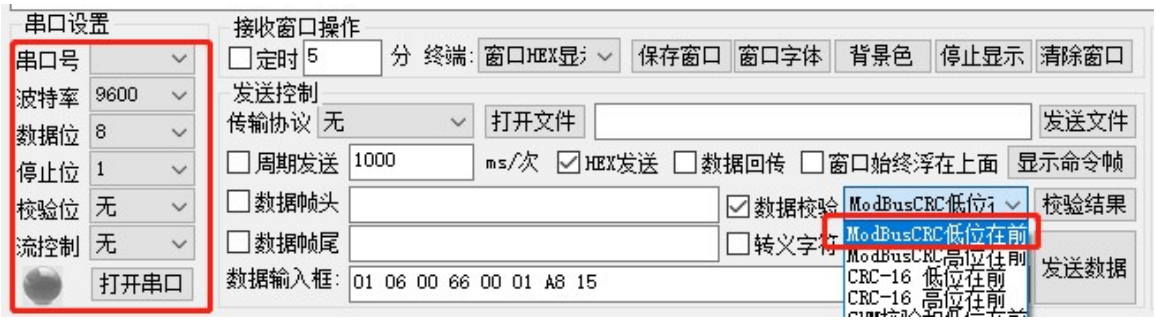
The driver adopts CRC-16/Modbus algorithm model. Users can refer to Appendix B.2 of GB/T 19582.2-2008 for more details about this calibration method. Below is C source program for generating verified code which is for user's reference.

```
typedef unsigned char u8;
typedef unsigned int u16;
/**
  *@brief    generating check code
  *@param *ptr    array for storing information code, the first address of array is placed the first byte of ADU
  *@param length    bytes of check code removed by ADU
  *@retval u16   check code
  */
u16 getCRC16(u8 *ptr, u8 length)
{
   u8 i;
   u16 crc = 0xFFFF;

   if(length == 0)
      length = 1;
```

```
    while(length--) {
        crc ^= *ptr;
        for(i = 0; i < 8; i++)  {
            if(crc & 1)  {
                crc >>= 1;
                crc ^= 0xA001;
            }
            else
                crc >>= 1;
        }
        ptr++;
    }
  return(crc);
}
```

Example: The client PC serial port debugging assistant can be set according to the following conditions



Baud rate 9600 8 data bits, 1 stop bit, Modbus CRC low bit first, no check bit

●Determine the structure of ADU according to function and send the ADU

Since this driver only uses two function codes: 03 and 06, you only need to be familiar with the formats of two ADUs. For the request ADU and response ADU of the 03 function code, refer to Section 7.3 of GB/T19582.1-2008. For the request ADU and response ADU of the 06 function code, refer to Section 7.6 of GB/T19582.1-2008. The following table lists the commonly used ADUs for user reference. In the following example, the slave address is 1 (dial 1).

Read register	
Read the acceleration time setting value	Send：**01 03 00 8A 00 01 74 0A** Receive：**01 03 02 00 00 B8 44** (The acceleration time is 0)
Read the set speed	Send：**01 03 00 56 00 01 05 C8** Receive：**01 03 02 04 27 FA 9E** (The drive set speed is 1063)
Unicast mode write register	
Write acceleration time setting	Send：**01 06 00 8A 01 2C A8 6D** Receive**01 06 00 8A 01 2C A8 6D** (The driver set acceleration time is 300))
Write speed	Send：**01 06 00 56 04 E2 EB 53** Receive**01 06 00 56 04 E2 EB 53** (The drive write speed is 1250)

Write limit maximum speed	Send：01 06 00 92 0B B8 2F 65 Receive01 06 00 92 0B B8 2F 65 (The maximum write speed limit of the drive is 3000)
Broadcast mode write register	
Write pole pair number 5	Send：00 06 00 02 00 05 E9 D8 Receive：No response