BIGTREETECH TMC5160-V1.0 Manual

1.Product Introduction

TMC5160 is a high-power stepper motor driving control chip, which has externally expanded power MOS tube, the maximum current can reach 20A, low heat when working. Supporting TRINAMICS unique stealthChop2 mode eliminates motor noise by reducing resonance and achieves a silent effect.StallGuard2 ™ blocking rotation detection is a safe stop to detect the motor and replaces a mechanical stop switch.It can achieve stepping motor torque control or sensorless back to the origin.

dcStep ™ can let the motor run near its limit load and speed limit, can be achieved in the absence of any step 10 times or more dynamic range.

spreadCycle ™ chopper algorithm has high precision, used for high dynamic motor sports and current wave absolutely clean. Low noise, low resonance and low vibration chopper.

coolStep ™ current control function, optimize the drive performance, balance the speed and motor torque and optimize energy efficiency, drive smoothly and no noise, energy consumption can be reduced by 75%.

TMC5160 is an expansion of the TMC2100, TMC2130 and TMC5130 series to higher voltages and higher motor currents.

2.Product parameters

Driver chip: TMC5160-WA;

Product size: 15.3mm*20.4mm;

Power supply voltage (VM) : 8V-- -40v;

Maximum current: 4.4A (the sense resistor determines the maximum current);

Maximum subdivision: 256;

Working mode: SPI mode

3.Product Advantages

(1)External power MOS tube can support larger current, the maximum current can reach 20A (because the module is limited by area, the current cannot exceed)

(2)Super-mute mode

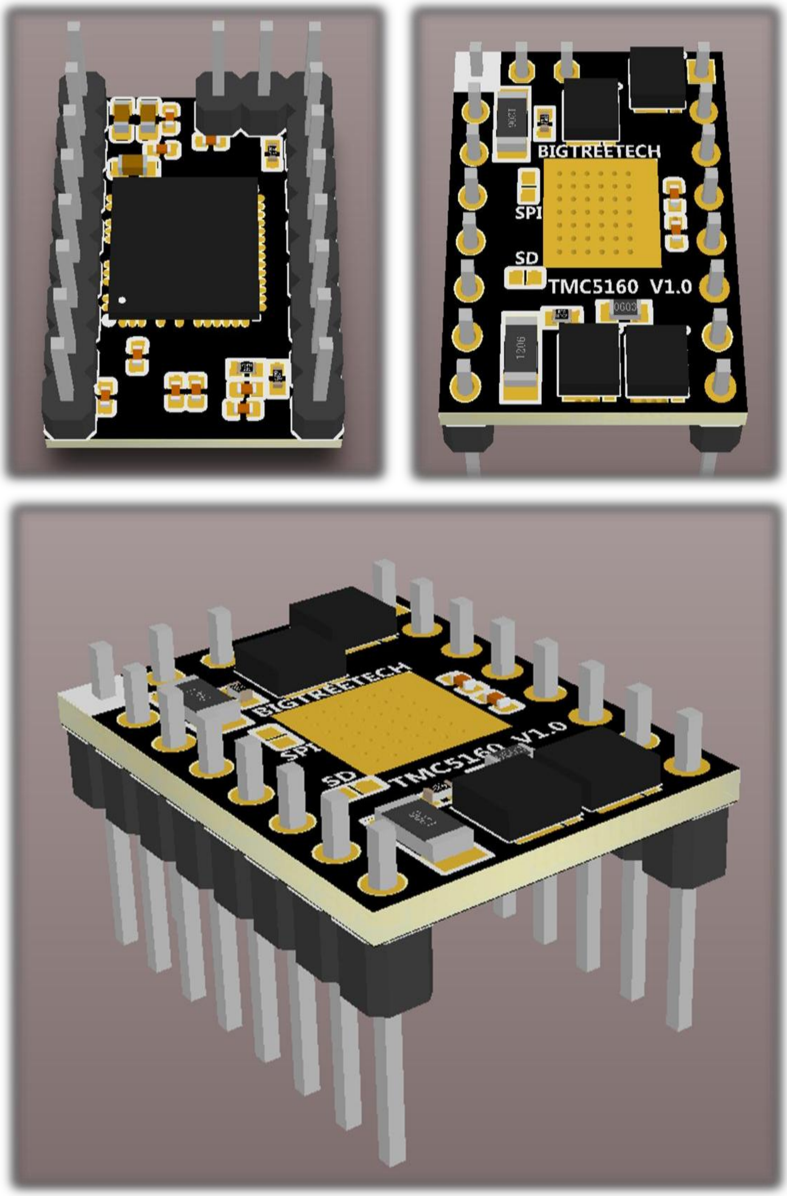
(3)Low calorific value

(4)can prevent motor shake

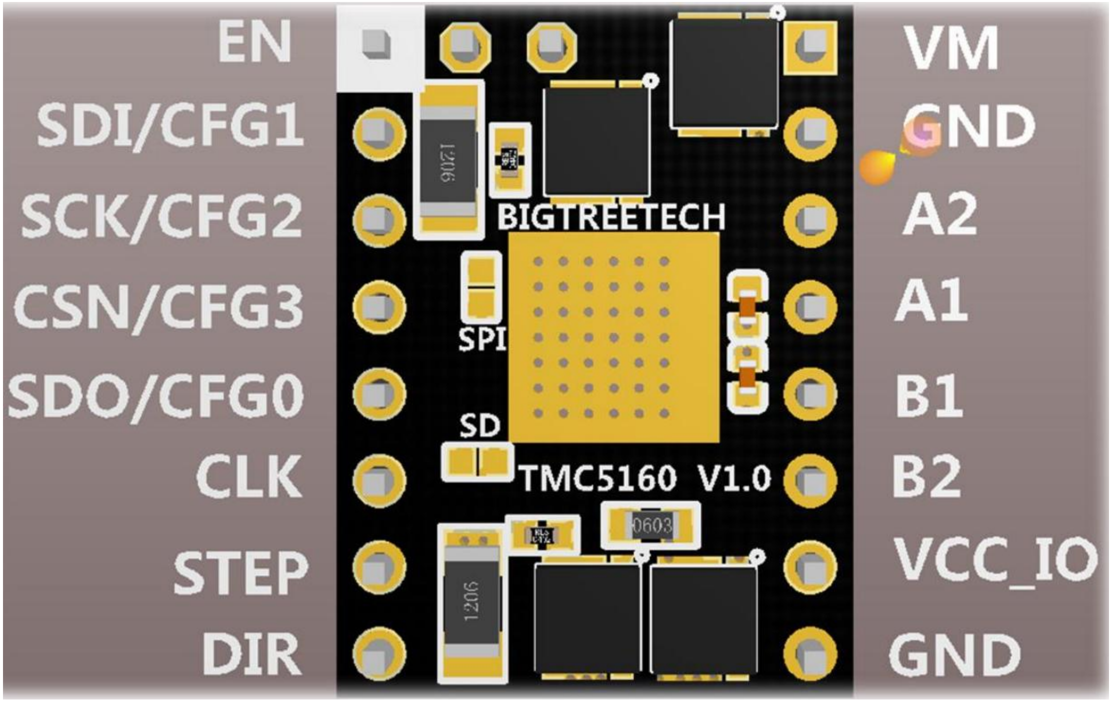
(5)not easy to lose step

(6)can drive 57 stepper motor

4.3D Renderings



4.(1)Pin description

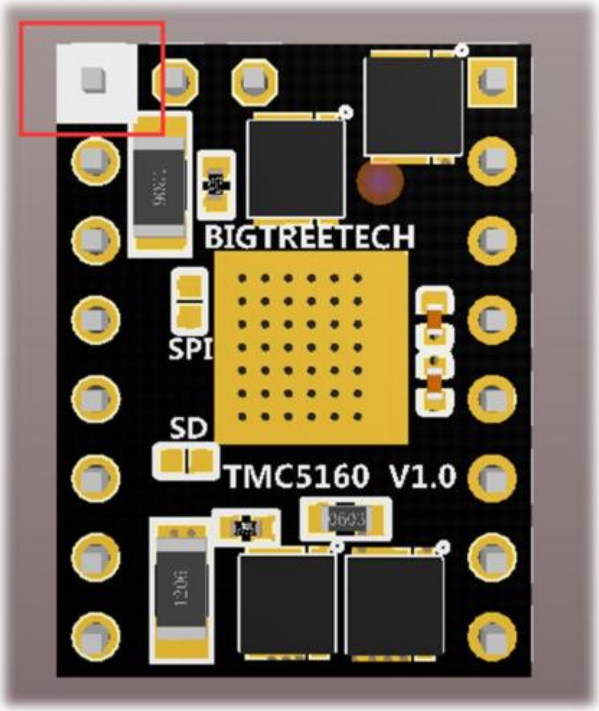


Pin function

J1	Function	J2	Function
1	(EN) Enable	1	(VM) Motor Supply Voltage
2	(SDI/CFG1) data	2	(GND) Grounding
3	(SCK/CFG2) Clock	3	（A2）
4	(CSN/CFG3) (Chip Selection)	4	（A1）
5	(SDO/CFG0) data	5	（B1）
6	（CLK）	6	（B2）
7	(STEP) Pulse Input	7	(VCC_IO) Logical Voltage
8	DIR Directional Input	8	(GND) Grounding

6.Driver installation instructions

The pins with white block diagram on the driver are enabled (EN) pins as shown in the red box below.

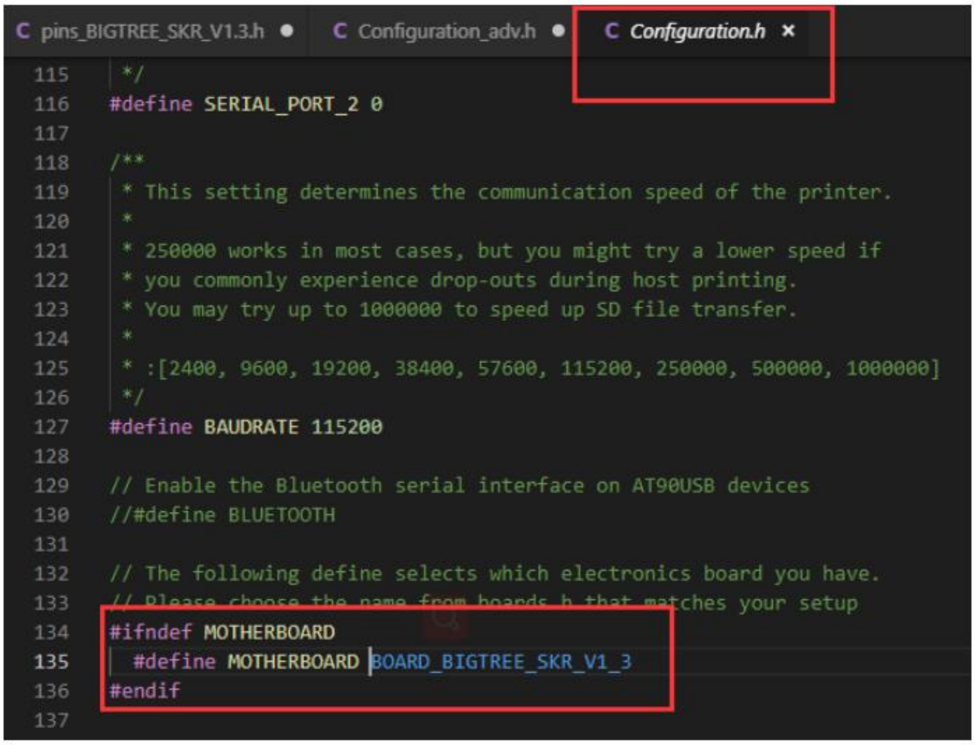


7. Firmware modification instructions

Special note *: Currently only firmware of Marlin 2.0 and above supports the SPI mode of TMC5160.

Step 1:

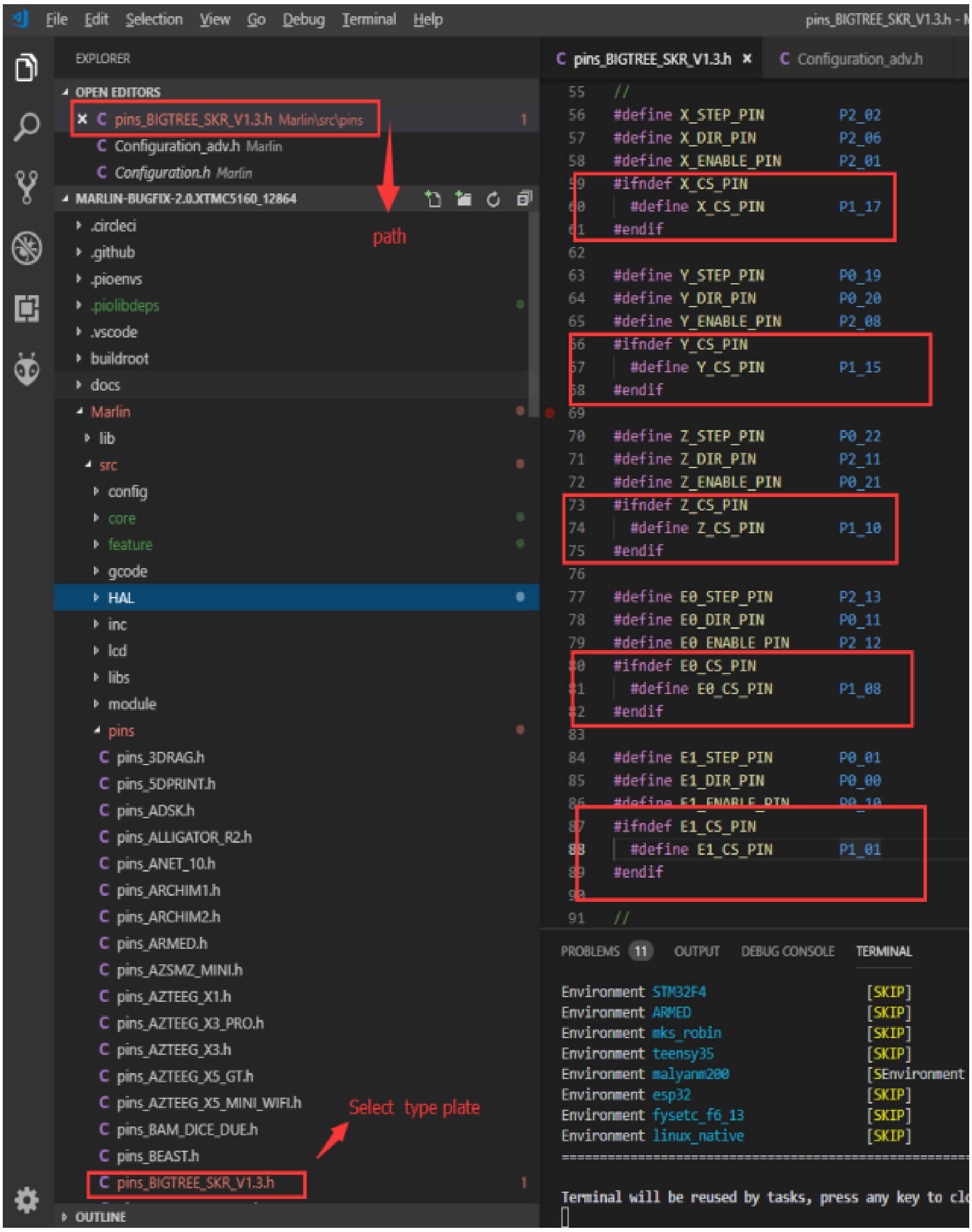
Find and open “Configuration.h” in Marlin 2.0 firmware File, and then find "# define MOTHERBOARD XXXXXX" and "XXXXX"Represents the type of board used. Confirm the motherboard you use.



```
 */
#define SERIAL_PORT_2 0

/**
 * This setting determines the communication speed of the printer.
 *
 * 250000 works in most cases, but you might try a lower speed if
 * you commonly experience drop-outs during host printing.
 * You may try up to 1000000 to speed up SD file transfer.
 *
 * :[2400, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000]
 */
#define BAUDRATE 115200

// Enable the Bluetooth serial interface on AT90USB devices
//#define BLUETOOTH

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
  #define MOTHERBOARD BOARD_BIGTREE_SKR_V1_3
#endif
```

Step 2:

Find the "pins_xxxxxx.h" file (xxxx represents the model of the board) in the Marlin \src \pins directory, and then find "X_CS_PIN", "Y_CS_PIN", "Z_CS_PIN", "EO_CS_PIN" under the file.Finally modify the following pin name to the pin you use.



path

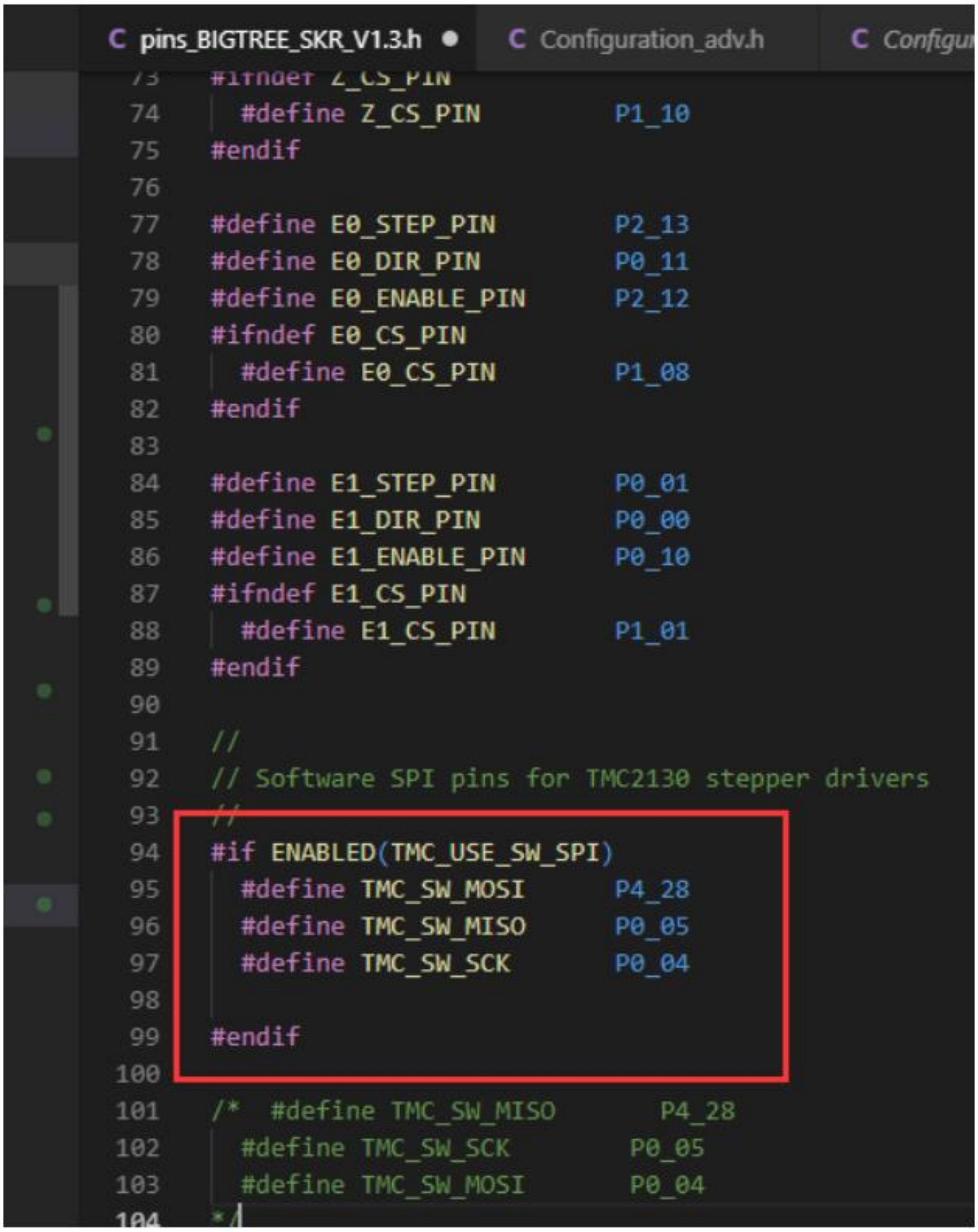
Select type plate

```
//
#define X_STEP_PIN         P2_02
#define X_DIR_PIN          P2_06
#define X_ENABLE_PIN       P2_01
#ifndef X_CS_PIN
  #define X_CS_PIN         P1_17
#endif

#define Y_STEP_PIN         P0_19
#define Y_DIR_PIN          P0_20
#define Y_ENABLE_PIN       P2_08
#ifndef Y_CS_PIN
  #define Y_CS_PIN         P1_15
#endif

#define Z_STEP_PIN         P0_22
#define Z_DIR_PIN          P2_11
#define Z_ENABLE_PIN       P0_21
#ifndef Z_CS_PIN
  #define Z_CS_PIN         P1_10
#endif

#define E0_STEP_PIN        P2_13
#define E0_DIR_PIN         P0_11
#define E0_ENABLE_PIN      P2_12
#ifndef E0_CS_PIN
  #define E0_CS_PIN        P1_08
#endif

#define E1_STEP_PIN        P0_01
#define E1_DIR_PIN         P0_00
#define E1_ENABLE_PIN      P0_10
#ifndef E1_CS_PIN
  #define E1_CS_PIN        P1_01
#endif

//
```

```
Environment STM32F4                        [SKIP]
Environment ARMED                          [SKIP]
Environment mks_robin                      [SKIP]
Environment teensy35                       [SKIP]
Environment malyanm200                     [SEnvironment
Environment esp32                          [SKIP]
Environment fysetc_f6_13                   [SKIP]
Environment linux_native                   [SKIP]
=======================================================

Terminal will be reused by tasks, press any key to clo
```

Step 3:

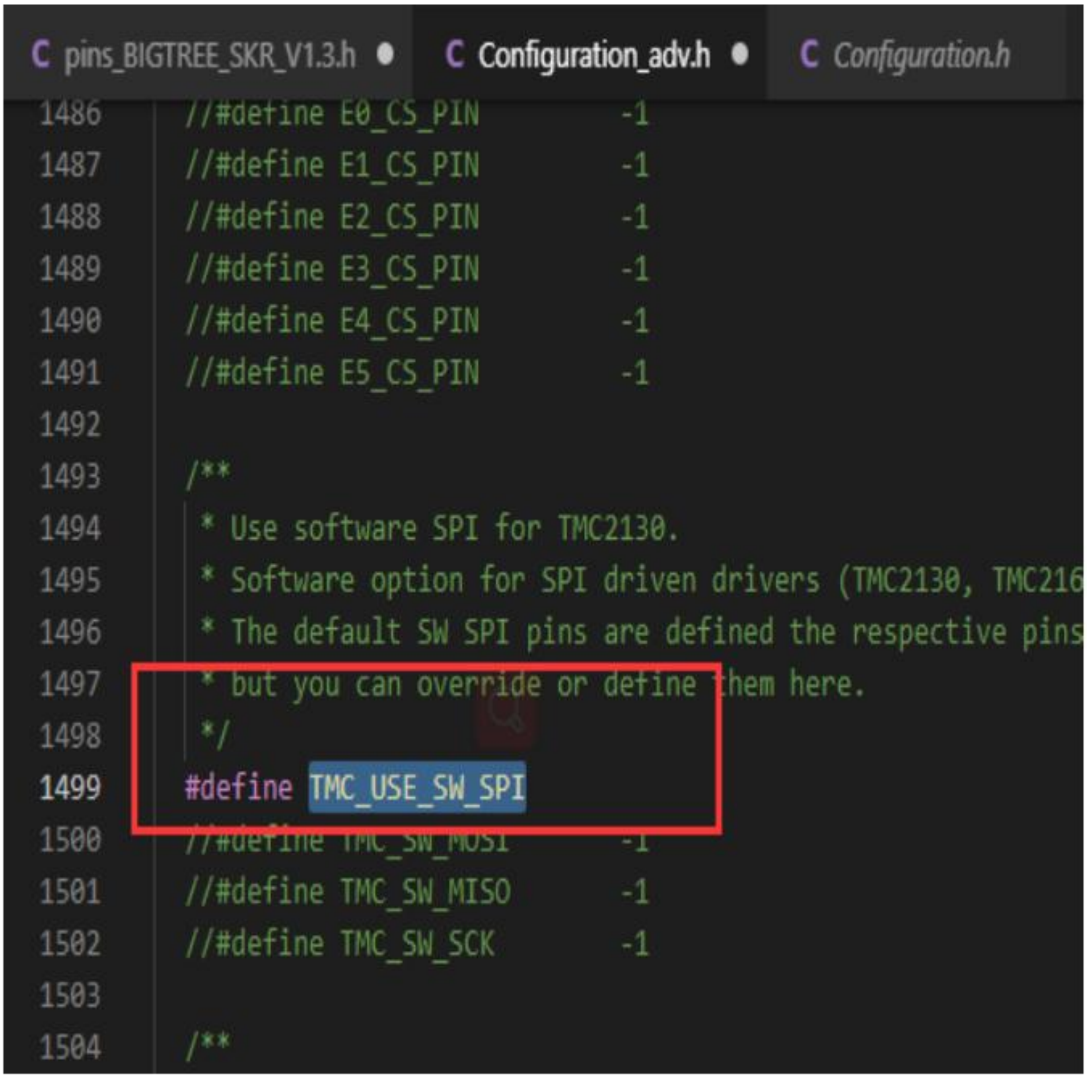
Find "#define TMC_SW_MOSI XXX", "#define TMC_SW_MISO XXX", "#define TMC_SW_SCK XXX" under the file in Step 2. Modify "XXX" to the pin you want to use.



```
#ifndef Z_CS_PIN
  #define Z_CS_PIN         P1_10
#endif

#define E0_STEP_PIN        P2_13
#define E0_DIR_PIN         P0_11
#define E0_ENABLE_PIN      P2_12
#ifndef E0_CS_PIN
  #define E0_CS_PIN        P1_08
#endif

#define E1_STEP_PIN        P0_01
#define E1_DIR_PIN         P0_00
#define E1_ENABLE_PIN      P0_10
#ifndef E1_CS_PIN
  #define E1_CS_PIN        P1_01
#endif

//
// Software SPI pins for TMC2130 stepper drivers
//
#if ENABLED(TMC_USE_SW_SPI)
  #define TMC_SW_MOSI      P4_28
  #define TMC_SW_MISO      P0_05
  #define TMC_SW_SCK       P0_04

#endif

/*  #define TMC_SW_MISO       P4_28
  #define TMC_SW_SCK        P0_05
  #define TMC_SW_MOSI       P0_04
*/
```

Step 4:

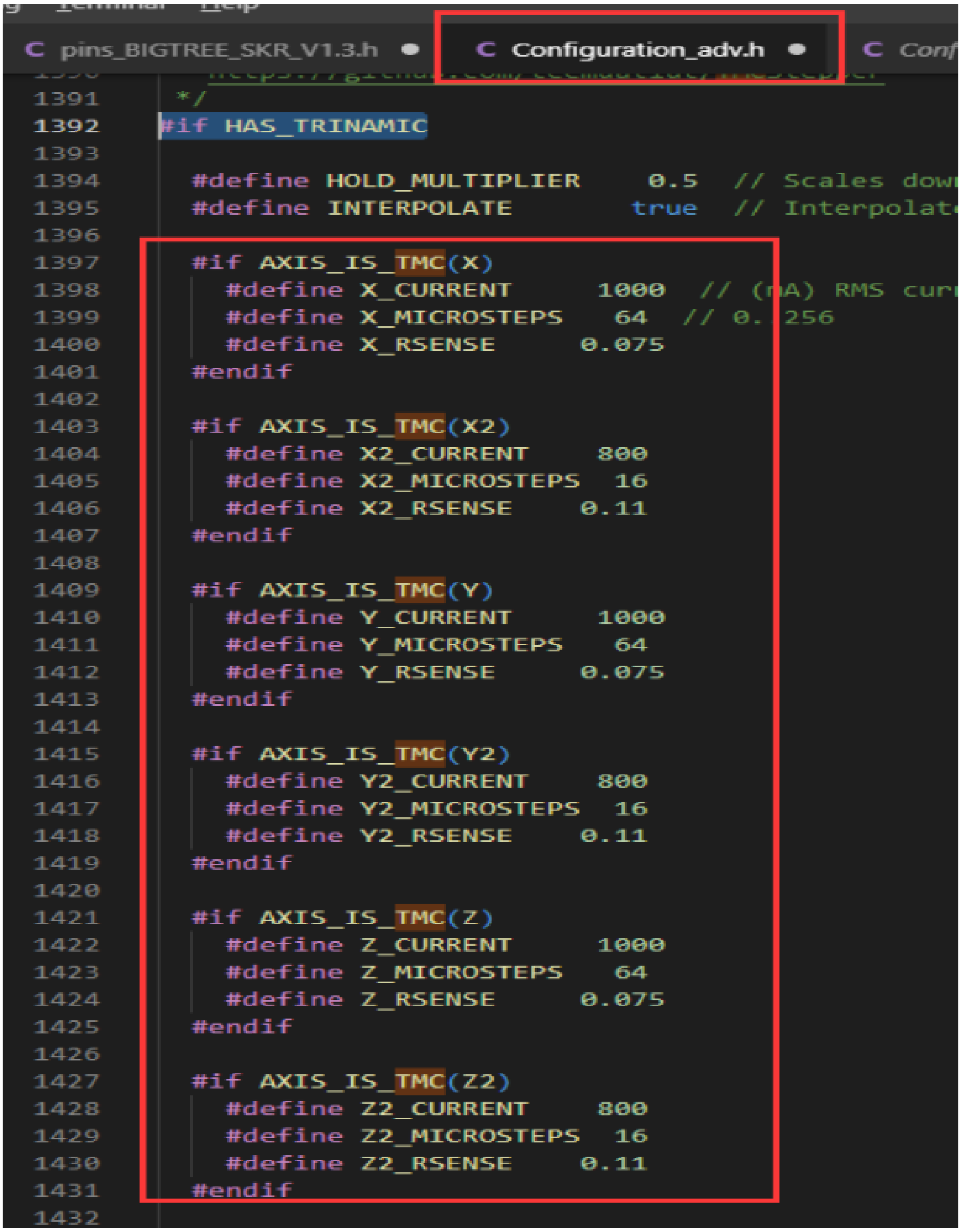
Find and open "Configuration_adv.h" and then find " define TMC_USE_SW_SPI" to remove the shield "//".



```
//#define E0_CS_PIN          -1
//#define E1_CS_PIN          -1
//#define E2_CS_PIN          -1
//#define E3_CS_PIN          -1
//#define E4_CS_PIN          -1
//#define E5_CS_PIN          -1

/**
 * Use software SPI for TMC2130.
 * Software option for SPI driven drivers (TMC2130, TMC216
 * The default SW SPI pins are defined the respective pins
 * but you can override or define them here.
 */
#define TMC_USE_SW_SPI
//#define TMC_SW_MOSI        -1
//#define TMC_SW_MISO        -1
//#define TMC_SW_SCK         -1

/**
```

Step 5:

Under the "Configuration_adv.h" file, find "# define X_CURRENT", "#define X_MICROSTEPS", "#define X_RSENSE". After modification of the parameters (the axes used need to be modified), the RSENE of the axes used should be changed to "0.075"

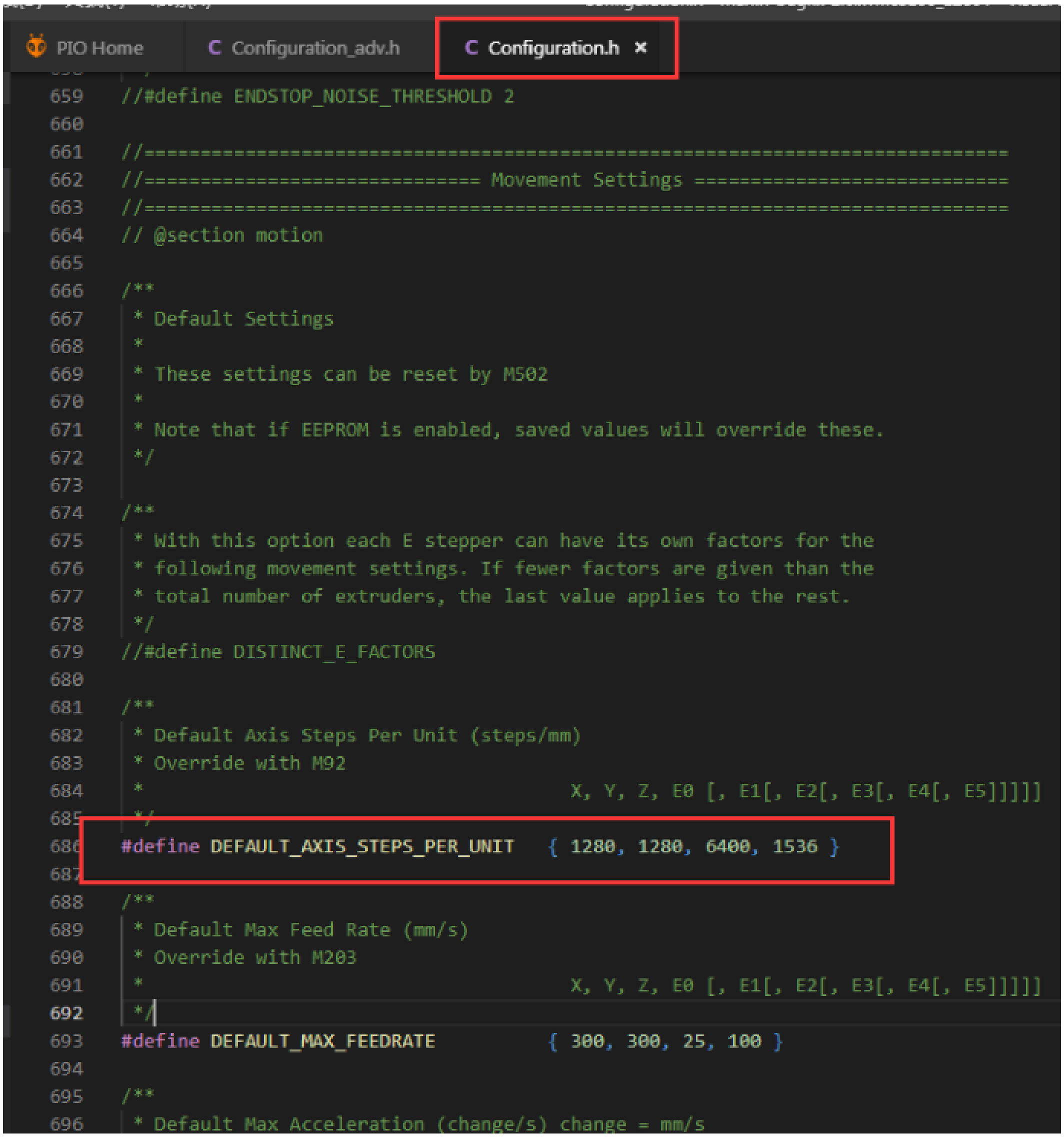


```
 */
#if HAS_TRINAMIC

  #define HOLD_MULTIPLIER    0.5  // Scales dow
  #define INTERPOLATE       true  // Interpolat

  #if AXIS_IS_TMC(X)
    #define X_CURRENT      1000  // (mA) RMS cur
    #define X_MICROSTEPS    64  // 0..256
    #define X_RSENSE      0.075
  #endif

  #if AXIS_IS_TMC(X2)
    #define X2_CURRENT     800
    #define X2_MICROSTEPS   16
    #define X2_RSENSE     0.11
  #endif

  #if AXIS_IS_TMC(Y)
    #define Y_CURRENT      1000
    #define Y_MICROSTEPS    64
    #define Y_RSENSE      0.075
  #endif

  #if AXIS_IS_TMC(Y2)
    #define Y2_CURRENT     800
    #define Y2_MICROSTEPS   16
    #define Y2_RSENSE     0.11
  #endif

  #if AXIS_IS_TMC(Z)
    #define Z_CURRENT      1000
    #define Z_MICROSTEPS    64
    #define Z_RSENSE      0.075
  #endif

  #if AXIS_IS_TMC(Z2)
    #define Z2_CURRENT     800
    #define Z2_MICROSTEPS   16
    #define Z2_RSENSE     0.11
  #endif

```

Step 6:

After the modification of Step 5 is completed, find and open "Configuration. h" and then find "# define" DEFAULT_AXIS_STEPS_PER_UNIT" modifies the following parameters to set the subdivision, which must correspond to the subdivision of step 5.

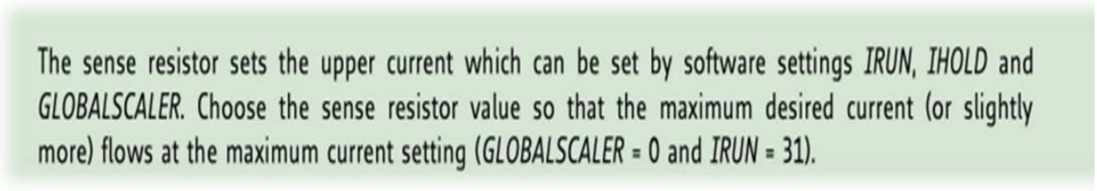
Subdivision calculation method, "80, 80, 400, 96" represents 16 subdivisions. If modified to 32 subdivisions, it will be "80* (32/16), 80* (32/16), 400* (32/16), 96* (32/16).



```
//#define ENDSTOP_NOISE_THRESHOLD 2

//===========================================================================
//============================= Movement Settings ============================
//===========================================================================
// @section motion

/**
 * Default Settings
 *
 * These settings can be reset by M502
 *
 * Note that if EEPROM is enabled, saved values will override these.
 */

/**
 * With this option each E stepper can have its own factors for the
 * following movement settings. If fewer factors are given than the
 * total number of extruders, the last value applies to the rest.
 */
//#define DISTINCT_E_FACTORS

/**
 * Default Axis Steps Per Unit (steps/mm)
 * Override with M92
 *                                      X, Y, Z, E0 [, E1[, E2[, E3[, E4[, E5]]]]]
 */
#define DEFAULT_AXIS_STEPS_PER_UNIT   { 1280, 1280, 6400, 1536 }

/**
 * Default Max Feed Rate (mm/s)
 * Override with M203
 *                                      X, Y, Z, E0 [, E1[, E2[, E3[, E4[, E5]]]]]
 */
#define DEFAULT_MAX_FEEDRATE          { 300, 300, 25, 100 }

/**
 * Default Max Acceleration (change/s) change = mm/s
```

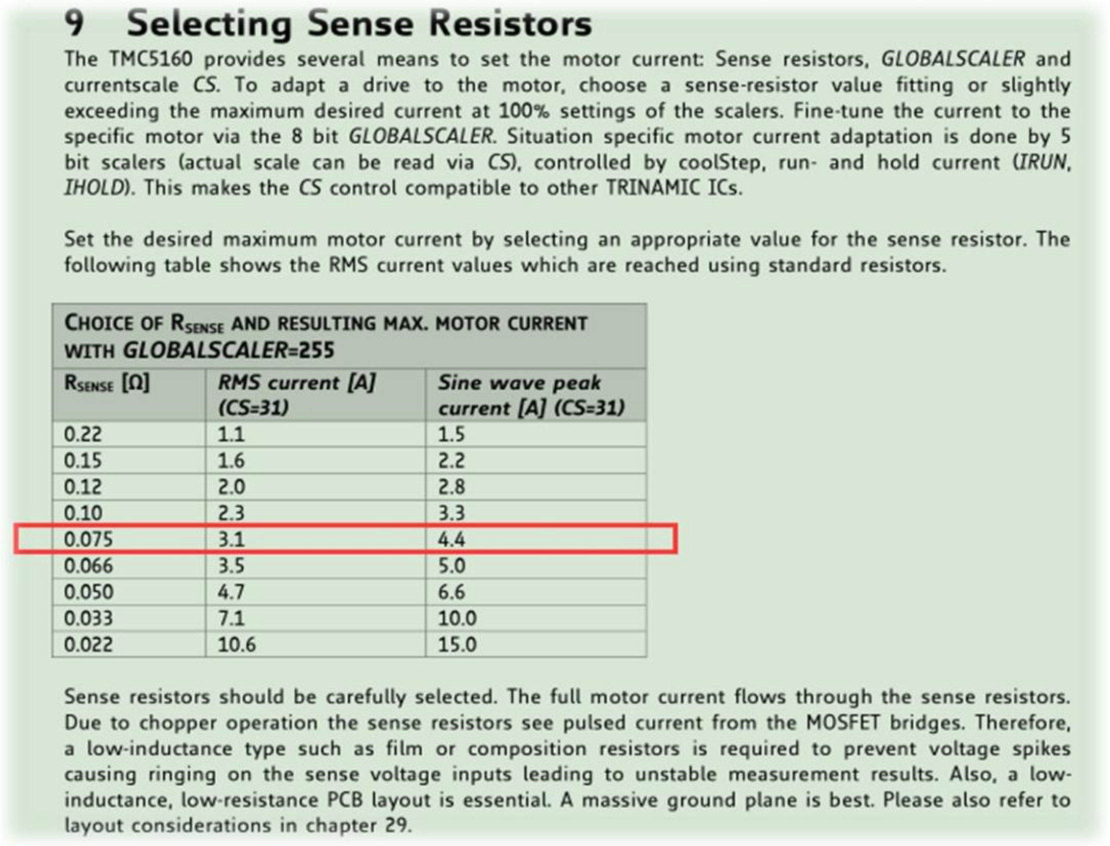
8. Driver Current Description

The range of driver current depends on the value of the sense resistor.



The sense resistor sets the upper current which can be set by software settings *IRUN*, *IHOLD* and *GLOBALSCALER*. Choose the sense resistor value so that the maximum desired current (or slightly more) flows at the maximum current setting (*GLOBALSCALER* = 0 and *IRUN* = 31).

The relationship between the effective value and maximum value of the driver current and the magnitude of the sense-resistor, please see the following picture:



9 Selecting Sense Resistors

The TMC5160 provides several means to set the motor current: Sense resistors, *GLOBALSCALER* and currentscale *CS*. To adapt a drive to the motor, choose a sense-resistor value fitting or slightly exceeding the maximum desired current at 100% settings of the scalers. Fine-tune the current to the specific motor via the 8 bit *GLOBALSCALER*. Situation specific motor current adaptation is done by 5 bit scalers (actual scale can be read via *CS*), controlled by coolStep, run- and hold current (*IRUN*, *IHOLD*). This makes the *CS* control compatible to other TRINAMIC ICs.

Set the desired maximum motor current by selecting an appropriate value for the sense resistor. The following table shows the RMS current values which are reached using standard resistors.

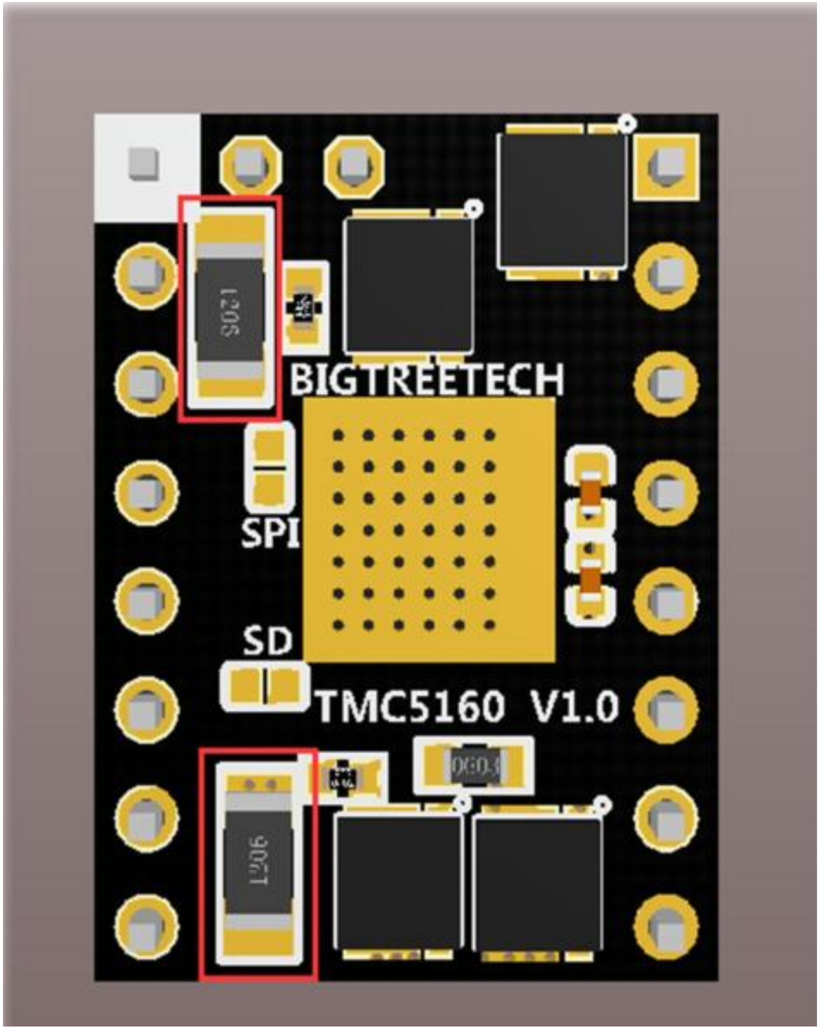
CHOICE OF R_{SENSE} AND RESULTING MAX. MOTOR CURRENT WITH *GLOBALSCALER*=255		
R_{SENSE} [Ω]	*RMS current [A] (CS=31)*	*Sine wave peak current [A] (CS=31)*
0.22	1.1	1.5
0.15	1.6	2.2
0.12	2.0	2.8
0.10	2.3	3.3
0.075	3.1	4.4
0.066	3.5	5.0
0.050	4.7	6.6
0.033	7.1	10.0
0.022	10.6	15.0

Sense resistors should be carefully selected. The full motor current flows through the sense resistors. Due to chopper operation the sense resistors see pulsed current from the MOSFET bridges. Therefore, a low-inductance type such as film or composition resistors is required to prevent voltage spikes causing ringing on the sense voltage inputs leading to unstable measurement results. Also, a low-inductance, low-resistance PCB layout is essential. A massive ground plane is best. Please also refer to layout considerations in chapter 29.

The sense resistors used in the TMC5160-V1.0 is 0.075R, so the effective value of the driver current of this driver is 3.1A, and the maximum current is 4.4A.

If you need to use a larger current, you need to replace the value of the sense resistor yourself (you need to prepare the components and soldering yourself). Replace the resistor not less than 0.05R (subject to module size limit).

Note: It is not recommended to replace the resistor. If it must be replaced,the driver will be damaged during the replacement process is at your own risk.



9. Precautions

1). Always disconnect the power supply before installing the driver to prevent the driver from burning.

2). Be sure to confirm the direction of the driver before installing the driver. Prevent the driver from burning due to reverse connection.

3). Please do not plug and unplug the driver module when power is on to avoid damage.

4). When installing the heat sink, please be careful not to contact the heat sink and the pin header to prevent short circuit.

5). The product is sensitive to static electricity, please handle it carefully when using, it is best to remove the package when using.

Data download address

https://github.com/bigtreetech/BIGTREETECH-TMC5160-V1.0